Complexity Results and Integer Programming Models for Hospitals / Residents Problem Variants

Iain McBride

Submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

School of Computing Science

College of Science and Engineering
University of Glasgow

September 2015

Abstract

The classical Hospitals / Residents problem (HR) is a many-to-one bipartite matching problem involving preferences, motivated by centralised matching schemes arising in entry level labour markets, the assignment of pupils to schools and higher education admissions schemes, among its many applications. The particular requirements of these matching schemes may lead to generalisations of HR that involve additional inputs or constraints on an acceptable solution. In this thesis we study such variants of HR from an algorithmic and integer programming viewpoint.

The Hospitals / Residents problem with Couples (HRC) is a variant of HR that is important in practical applications because it models the case where couples submit joint preference lists over pairs of (typically geographically close) hospitals. It is known that an instance of HRC need not admit a stable matching. We show that deciding whether an instance of HRC admits a stable matching is NP-complete even under some very severe restrictions on the lengths and the structure of the participants' preference lists. However, we show that under certain restrictions on the lengths of the agents' preference lists, it is possible to find a maximum cardinality stable matching or report that none exists in polynomial time.

Since an instance of HRC need not admit a stable matching, it is natural to seek the 'most stable' matching possible, i.e., a matching that admits the minimum number of blocking pairs. We use a gap-introducing reduction to establish an inapproximability bound for the problem of finding a matching in an instance of HRC that admits the minimum number of blocking pairs. Further, we show how this result might be generalised to prove that a number of minimisation problems based on matchings having NP-complete decision versions have the same inapproximability bound. We also show that this result holds for more general minimisation problems having NP-complete decisions versions that are not based on matching problems.

Further, we present a full description of the first Integer Programming (IP) model for finding a maximum cardinality stable matching or reporting that none exists in an arbitrary instance

of HRC. We present empirical results showing the average size of a maximum cardinality stable matching and the percentage of instances admitting stable matching taken over a number of randomly generated instances of HRC with varying properties. We also show how this model might be generalised to the variant of HRC in which ties are allowed in either the hospitals' or the residents' preference lists, the Hospitals / Residents problem with Couples and Ties (HRCT). We also describe and prove the correctness of the first IP model for finding a maximum cardinality 'most stable' matching in an arbitrary instance of HRC. We describe empirical results showing the average number of blocking pairs admitted by a most-stable matching as well as the average size of a maximum cardinality 'most stable' matching taken over a number of randomly generated instances of HRC with varying properties. Further, we examine the output when the IP model for HRCT is applied to real world instances arising from the process used to assign medical graduates to Foundation Programme places in Scotland in the years 2010-2012.

The Hungarian Higher Education Allocation Scheme places a number of additional constraints on the feasibility of an allocation and this gives rise to various generalisations of HR. We show how a number of these additional requirements may be modelled using IP techniques by use of an appropriate combination of IP constraints. We present IP models for HR with Stable Score Limits and Ties, HR with Paired Applications, Ties and Stable Score limits, HR with Common Quotas, Ties and Stable Score Limits and also HR with Lower Quotas, Ties and Stable Score limits that model these generalisations of HR.

The Teachers' Allocation Problem (TAP) is a variant of HR that models the allocation of trainee teachers to supervised teaching positions in Slovakia. In TAP teachers express preference lists over pairs of subjects at individual schools. It is known that deciding whether an optimal matching exists that assigns all of the trainee teachers is NP-complete for a number of restricted cases. We describe IP models for finding a maximum cardinality matching in an arbitrary TAP instance and for finding a maximum cardinality stable matching, or reporting that none exists, in a TAP instance where schools also have preferences. We show the results when applying the first model to the real data arising from the allocation of trainee teachers to schools carried out at P.J. Šafárik University in Košice in 2013.

Acknowledgements

I would like to thank my supervisor David Manlove for his patience, feedback and guidance during the course of my PhD. I would also like to thank my second supervisor Patrick Prosser for posing questions that always seemed to lead to improvements in the quality of my thesis.

My thanks also go to Rob Irving, Augustine Kwanashie and Baharak Rastegari for all of their help and input throughout the course of my PhD. I would also like to thank the members of the Algorithmicists and FATA groups in the School of Computing Science at the University of Glasgow for their suggestions and comments in response to various presentations of my work.

I would also like to thank the anonymous reviewers of submitted versions of parts of the work contained in this thesis for their valuable comments.

I would like to thank SICSA for funding my studentship and also COST Action IC1205 for funding my visits to Košice and Budapest during the course of my PhD.

I would like to thank Katarína Cechlárová for being such a generous and welcoming host during my visit to Košice and also for useful comments on the material in this thesis. I would also like to thank Péter Biró and Tamás Fleiner for being such wonderful hosts to my wife and I during our visit to Budapest.

I would like to thank my friend Steve for proof-reading parts of this thesis.

Declaration

This thesis is submitted in accordance with the rules for the degree of Doctor of Philosophy at the University of Glasgow. None of the material contained herein has been submitted for any other degree.

The proof of Theorem 3.3.1 in Section 3.3.1 is joint work with Péter Biró and David Manlove. The proof of Theorem 3.4.1 in Section 3.4 is joint work with David Manlove.

The material contained in Chapter 8 is joint work with Péter Biró.

The alternative measure function discussed in Section 10.2 was suggested by Kitty Meeks from the School of Mathematics and Statistics at the University of Glasgow.

All other work in this thesis is claimed as original.

Publications

The following publications have arisen from the work presented in this thesis.

1. I. McBride and D.F. Manlove. An Integer Programming Model for the Hospitals / Residents problem with Couples. In *Operations Research Proceedings 2013*, pages 293–299. Springer, 2014
(This paper relates to the work contained in Chapter 4.)

2. P. Biró, D.F. Manlove, and I. McBride. The Hospitals / Residents problem with Couples: Complexity and Integer Programming Models. In *Proceedings of SEA2014: the 8th Symposium on Experimental Algorithms*, volume 8504 of *Lecture Notes in Computer Science*, pages 10–21. Springer International Publishing, 2014
(This paper relates to the work contained in Chapter 4.)

3. P. Biró and I. McBride. Integer Programming Methods for Special College Admissions problems. In *Combinatorial Optimization and Applications*, volume 8881 of *Lecture Notes in Computer Science*, pages 429–443. Springer International Publishing, 2014
(This paper relates to the work contained in Chapter 8.)

4. K. Cechlárová, T. Fleiner, D.F. Manlove, I. McBride, and E. Potpinková. Modelling Practical Placement of Trainee Teachers to Schools. *Central European Journal of Operations Research*, 23(3):547–562, 2015
(This paper relates to the work contained in Chapter 9.)

To my wife Liz. I love you.

Table of Contents

1 Introduction and Thesis Outline **1**

1.1 Introduction 1

1.2 Thesis Outline 5

2 Literature Review and Preliminary Definitions **8**

2.1 The Stable Marriage problem 8

2.1.1 The Stable Marriage Problem with Incomplete Preference Lists . . 9

2.2 The Hospitals / Residents problem 9

2.3 The Hospitals / Residents problem with Couples 11

2.3.1 Fundamental definitions 11

2.3.2 The Hospitals / Residents problem with Paired Applications 16

2.4 'Most stable' matchings and optimisation problems 18

2.5 The Hospitals / Residents problem with Ties 20

2.6 The Hospitals / Residents problem with Lower / Common Quotas 21

2.6.1 The Hospitals / Residents problem with Lower Quotas 22

2.6.2 The Hospitals / Residents problem with Common Upper Quotas . . 23

2.7 The Hospitals / Residents problem with Stable Score Limits 25

2.7.1 Fundamental definitions 25

2.7.2 The Hospitals / Residents problem with Paired Applications and Score Limits 29

2.7.3 The Hospitals / Residents problem with Lower Quotas and Score Limits 30

2.7.4 The Hospitals / Residents problem with Common Quotas and Score Limits 32

2.8 The Teachers' Allocation Problem 34
2.8.1 The Teachers Allocation problem with Stability 36
2.8.2 The Teachers Allocation problem with Stability and Subject Specific Preference Lists 38
2.9 Exponential techniques applied to HR and its variants 39

3 Complexity Results for HRC **42**
3.1 Introduction 42
3.2 Complexity results for $(\infty, 1, \infty)$-HRC 43
3.3 Complexity results for $(2, 2)$-HRC 47
3.3.1 NP-completeness result for $(2, 2)$-HRC 47
3.3.2 Inapproximability of $(2, 2)$-MIN BP HRC 50
3.4 Complexity results for $(2, 3)$-HRC 52
3.5 Complexity results for $(3, 3)$-HRC DUAL MARKET 56
3.6 Complexity results for $(\infty, 1, \infty)$-HR PA 66
3.6.1 $(\infty, 1, \infty)$-HR PA is NP-complete 66
3.7 Efficiently solvable variants of HRC 68
3.7.1 Fixed assignments in HRC 68
3.7.2 $(2, 1, 2)$-HRC is efficiently solvable 70

4 Integer programming model for HRC variants under MM-stability **78**
4.1 Introduction 78
4.2 An IP formulation for HR 78
4.2.1 Variables in the IP model for HR 79
4.2.2 Constraints in the IP model for HR 79
4.2.3 Proof of correctness of the IP model for HR 80
4.3 An IP formulation for HRC 82
4.3.1 Variables in the IP model for HRC 83
4.3.2 Constraints in the IP model for HRC 84
4.3.3 Proof of correctness of constraints in the IP model for HRC 87
4.3.4 Complexity of the IP model for HRC 95

4.4 Creating the IP model from an example HRC instance 95
4.5 An integer programming formulation for HRCT 98
4.6 An integer programming formulation for MIN BP HRC 100
4.6.1 Additional variables in the IP model for MIN BP HRC 100
4.6.2 Replacement constraints in the IP model for MIN BP HRC 100
4.6.3 Objective functions in the IP model for MIN BP HRC 102
4.6.4 Proof of correctness the IP model for MIN BP HRC 103

5 An integer programming model for HRC under BIS-stability 105
5.1 Introduction 105
5.2 MM-stability and BIS-stability are not equivalent 105
5.3 A hospital cloning method for HRC under MM-stability 106
5.4 An IP formulation for HRC under BIS stability 112
5.4.1 Variables in the IP model for HRC under BIS stability 114
5.4.2 Constraints in the IP model for HRC under BIS stability 115

6 Empirical results from the IP models for HRC 121
6.1 Introduction 121
6.2 Overview of HRC experiments 122
6.2.1 Computational environment for HRC experiments 122
6.2.2 Correctness testing of the implemented model 123
6.2.3 Projection of preference lists to mimic the format of the SFAS instances 123
6.2.4 The popularity of hospitals 125
6.2.5 Pre-processing of instances before sending to the solver 126
6.3 Experimental results with randomly generated instances 127
6.3.1 HRC Experiment 1 127
6.3.2 HRC Experiment 2 131
6.3.3 HRC Experiment 3 136
6.3.4 HRC Experiment 4 141
6.4 HRCT IP models applied to instances arising from the SFAS application . . 146
6.4.1 Comparing solved instances 146
6.4.2 Results from 2010, 2011 and 2012 SFAS instances 147

7 Empirical results from the IP models for MIN BP HRC **149**

7.1 Introduction . . . 149

7.2 Overview of the MIN BP HRC experiments . . . 150

7.2.1 Computational environments for MIN BP HRC experiments . . . 150

7.2.2 Correctness testing of the implemented model . . . 150

7.2.3 Construction of random instances MIN BP HRC . . . 151

7.3 Experiments with randomly generated instances . . . 151

7.3.1 MIN BP HRC Experiment 1 . . . 151

7.3.2 MIN BP HRC Experiment 2 . . . 156

7.3.3 MIN BP HRC Experiment 3 . . . 160

7.3.4 MIN BP HRC Experiment 4 . . . 164

8 Integer programming models for HR with score limits **168**

8.1 Introduction . . . 168

8.2 A Type A IP model for HR SLT . . . 169

8.2.1 Variables in HR SLT A . . . 169

8.2.2 Constraints in HR SLT A . . . 169

8.2.3 Proof of correctness of the constraints in HR SLT A . . . 170

8.3 A Type A IP model for HR PA SLT . . . 172

8.3.1 Variables in HR PA SLT A . . . 172

8.3.2 Constraints in HR PA SLT A . . . 172

8.3.3 Proof of correctness of the constraints in HR PA SLT A . . . 174

8.4 A Type A IP model for HR CQ SLT . . . 175

8.4.1 Variables in HR CQ SLT A . . . 176

8.4.2 Constraints in HR CQ SLT A . . . 176

8.4.3 Proof of correctness of the constraints in HR CQ SLT A . . . 177

8.5 A Type B IP model for HR SLT . . . 179

8.5.1 Variables in HR SLT B . . . 180

8.5.2 Constraints in HR SLT B . . . 180

8.5.3 Proof of correctness of the constraints in HR SLT B . . . 182

8.6 A Type B IP model for HR LQ SLT . . . 184

8.6.1 Variables in HR LQ SLT B . . . 184
8.6.2 Constraints in HR LQ SLT B . . . 185
8.6.3 Proof of correctness of the constraints in HR LQ SLT B . . . 188

9 Complexity results and integer programming models for TAP variants 193

9.1 Introduction . . . 193
9.2 An integer programming formulation for TAP . . . 194
9.2.1 Variables in the IP model for TAP . . . 194
9.2.2 Constraints in the IP model for TAP . . . 194
9.2.3 Proof of correctness of the IP model for TAP . . . 195
9.3 The IP model for TAP applied to real data . . . 196
9.4 Complexity results derived from TAP . . . 198
9.5 An IP formulation for STABLE TAP . . . 199
9.5.1 Additional variables in the IP model for STABLE TAP . . . 199
9.5.2 Additional constraints in the IP model for STABLE TAP . . . 200
9.5.3 Proof of correctness of the IP model for STABLE TAP . . . 201
9.6 A reduction from STABLE TAP to HRC . . . 202

10 Inapproximability results for a class of minimisation problems. 204

10.1 Introduction . . . 204
10.2 An adjusted measure function for minimisation problems . . . 205
10.3 An inapproximability bound for a class of minimisation problems. . . . 206
10.4 Inapproximability bounds for problems in this thesis . . . 210
10.4.1 HRC variants . . . 210
10.4.2 TAP variants . . . 212
10.5 Inapproximability bounds for NP-complete stable matching problems . . . 214
10.5.1 Introduction . . . 214
10.5.2 HR LQ . . . 214
10.5.3 HR CQ . . . 215
10.5.4 HRT . . . 216
10.6 An inapproximability bound for k-COLOURING & SAT problems . . . 217

11 Conclusions and open problems 221

Bibliography 224

List of Tables

6.1 HRC Experiment 1 - Tabulation of output from HRC Experiment 1. 130
6.2 HRC Experiment 2 - Tabulation of output from HRC Experiment 2. 135
6.3 HRC Experiment 3 - Tabulation of output from HRC Experiment 3. 140
6.4 HRC Experiment 4 - Tabulation of output from HRC Experiment 4. 145
6.5 HRCT IP models applied to instances arising from the SFAS application. . . 148

7.1 MIN BP HRC Experiment 1 - Tabulation of output from MIN BP HRC Experiment 1. 155
7.2 MIN BP HRC Experiment 2 - Tabulation of output from MIN BP HRC Experiment 2. 159
7.3 MIN BP HRC Experiment 3 - Tabulation of output from MIN BP HRC Experiment 3. 163
7.4 MIN BP HRC Experiment 4 - Tabulation of output from MIN BP HRC Experiment 4. 167

9.1 Places available by subject for trainee teachers in Košice and surrounding areas 197

List of Figures

2.1 An instance of HRC that admits no stable matching due to Biró and Klijn. [14] 14

2.2 An instance of HRC that admits stable matchings of differing size due to Biró and Klijn. [14] 15

2.3 An instance of HR PA that admits no stable matching 17

2.4 An instance of HR PA that admits stable matchings of differing sizes 17

2.5 An instance of HRT that admits no resident-optimal stable matching. 21

2.6 An instance of HR LQ that admits two stable matchings of differing sizes [49]. 22

2.7 An instance of HR CQ that admits exactly two stable matchings, neither of which is optimal with respect to the residents. 24

2.8 An example instance showing that the assignment M induced from a set of H-stable score limits in an instance I of HR SLT need not be weakly stable matching in I', the instance of HRT induced from the residents' scores and preferences in I. 28

2.9 An instance of STABLE TAP that admits no stable matching. [17] 36

2.10 An instance of STABLE TAP that admits stable matchings of differing sizes. [49] 37

2.11 An instance of STABLE TAP that admits exactly two stable matchings, neither of which is optimal with respect to either set of agents. 38

3.1 Preference lists in I, the constructed instance of $(\infty, 1, \infty)$-HRC. 44

3.2 Preference lists in I, the constructed instance of $(2, 2)$-HRC. 48

3.3 A small instance of $(2, 2)$-HRC that admits no stable matching. 51

3.4 Preference lists in I, the constructed instance of $(2, 3)$-HRC. 54

3.5 Master list of preferences for resident couples in $(2, 3)$-HRC instance I. . . . 55

3.6 Master list of preferences for hospitals in $(2, 3)$-HRC instance I. 55

3.7 Preference lists in I, the constructed instance of $(3, 3)$-HRC. 58

3.8 Diagram of subset of agents in I, the constructed instance of $(3,3)$-HRC. . . . 59
3.9 Diagram of subset of agents in I, the constructed instance of $(3,3)$-HRC. . . . 59
3.10 Added agents in I', the constructed instance of $(3,3)$-HRC. 61
3.11 Diagram of added agents in I', the constructed instance of $(3,3)$-HRC. . . . 61
3.12 Master list of preferences for resident couples in $(3,3)$-COM HRC instance I. 65
3.13 Master list of preferences for single residents in $(3,3)$-COM HRC instance I. 65
3.14 Master list of preferences for hospitals in $(3,3)$-COM HRC instance I. 65
3.15 An instance of $(2,1,2)$-HRC containing an arbitrary number of couples and an arbitrary number of residents that has no unsatisfied fixed assignments. . 71
3.16 An exactly equivalent description of the instance shown in Figure 3.15 . . . 72

4.1 Example instance of HRC. 95
4.2 Variables created in J from the instance of HRC shown in Figure 4.1. 96
4.3 The assignment of values, $\mathbf{x}^u$, to the variables in J corresponding to the unstable matching M_u in the instance of HRC shown in Figure 4.1. 96
4.4 The assignment of values, $\mathbf{x}^s$, to the variables in J corresponding to the stable matching M_s in the instance of HRC shown in Figure 4.1. 97

5.1 An instance of HRC that admits a BIS-stable matching but admits no MM-stable matching. 106
5.2 An instance of HRC that admits an MM-stable matching but admits no BIS-stable matching. 106
5.3 An instance of HRC that shows that the cloning method does not work under BIS-stability. 112

6.1 HRC Experiment 1 - Number of stable matchings admitted by random instances. 128
6.2 HRC Experiment 1 - Mean size of a maximum cardinality stable matching. . 128
6.3 HRC Experiment 1 - Mean time to solve to optimality. 129
6.4 HRC Experiment 1 - Percentage of instances admitting a stable matching. . 129
6.5 HRC Experiment 2 - Number of stable matchings admitted by random instances. 132
6.6 HRC Experiment 2 - Mean size of a maximum cardinality stable matching. . 133
6.7 HRC Experiment 2 - Mean time to solve to optimality. 133

6.8 HRC Experiment 2 - Percentage of instances admitting a stable matching. . 134
6.9 HRC Experiment 3 - Number of stable matchings admitted by random instances. 137
6.10 HRC Experiment 3 - Mean size of a maximum cardinality stable matching. . 137
6.11 HRC Experiment 3 - Mean time to solve to optimality. 138
6.12 HRC Experiment 3 - Percentage of instances admitting a stable matching. . 138
6.13 HRC Experiment 4 - Number of stable matchings admitted by random instances. 142
6.14 HRC Experiment 4 - Mean size of a maximum cardinality stable matching. . 142
6.15 HRC Experiment 4 - Mean time to solve to optimality. 143
6.16 HRC Experiment 4 - Percentage of instances admitting a stable matching. . 143

7.1 MIN BP HRC Experiment 1 - Time taken to find a maximum cardinality 'most stable' matching. 152
7.2 MIN BP HRC Experiment 1 - Mean size of a maximum cardinality 'most stable' matching. 153
7.3 MIN BP HRC Experiment 1 - Mean and maximum number of blocking pairs in a 'most stable' matching. 154
7.4 MIN BP HRC Experiment 2 - Time taken to find a maximum cardinality 'most stable' matching. 156
7.5 MIN BP HRC Experiment 2 - Mean size of a maximum cardinality 'most stable' matching. 157
7.6 MIN BP HRC Experiment 2 - Mean and maximum number of blocking pairs in a 'most stable' matching. 158
7.7 MIN BP HRC Experiment 3 - Time taken to find a maximum cardinality 'most stable' matching. 160
7.8 MIN BP HRC Experiment 3 - Mean size of a maximum cardinality 'most stable' matching. 161
7.9 MIN BP HRC Experiment 3 - Mean and maximum number of blocking pairs in a 'most stable' matching. 162
7.10 MIN BP HRC Experiment 4 - Time taken to find a maximum cardinality 'most stable' matching. 164
7.11 MIN BP HRC Experiment 4 - Mean size of a maximum cardinality 'most stable' matching. 165

7.12 MIN BP HRC Experiment 4 - Mean and maximum number of blocking pairs in a 'most stable' matching. 166

Chapter 1

Introduction and Thesis Outline

1.1 Introduction

In this thesis we consider problems involving matching the members of a set of agents to one another where each agent has a preference order over its potential partners – the term *agent* intuitively means any participant in a matching process. In particular we consider matching problems in which agents have ordinal preferences; thus the agents' preferences have a notion of first, second, third choice etc.

In this setting the members of an agent's *preference list* are its *acceptable* partners in order of preference and any agent not in the preference list is an *unacceptable* partner. If two agents are involved in a pair in a matching, then they are *assigned* to each other in the matching or are *partners* in the matching. Commonly, matching problems have a notion of a *capacity* for an agent – an agent can only be assigned to a restricted number of acceptable partners. A *matching* is a set of pairs in which no agent is assigned to an unacceptable partner and no agent has a number of assignees that violates its capacity. In general we seek a matching that is *optimal* with respect to some predefined criteria.

Matching problems may be classified according to the quantity of disjoint subsets into which the set of agents may be subdivided and also whether the members of each of the disjoint subsets express preferences or not. Problems in which the agents comprise a single set and each agent expresses preferences over some subset of the other agents are termed *Non-bipartite matching problems with preferences*. The archetypal problem in this context is the *Stable Roommates* problem (SR) [31, 35, 40] where we seek to assign students to two person accommodation on the basis of their preferences over one another. However, this problem model also applies in the context of kidney exchange programmes where a patient with a willing but incompatible donor may obtain a transplant by swapping their donor with that of another patient in a similar position [51, 72, 73, 74].

Problems in which the set of agents comprises two disjoint subsets, but the agents in only one of the subsets express preferences over the members of the opposing subset are *bipartite matching problems with one sided preferences*. The archetypal problem in this context is the *House Allocation problem* (HA) [21, 34, 77]; HA is the underlying abstract allocation model in contexts involving the allocating of students to on-campus housing [19], allocating social housing in China [82] and assigning reviewers to papers [30].

In this thesis we consider specifically *bipartite matching problems with two sided preferences* – problems comprising two disjoint subsets of agents where each agent in each of the disjoint subsets has some preference ordering over members of the other subset. A number of matching programmes across the world have bipartite matching problems with two sided preferences as their underlying abstract problem model. In the medical sphere the National Resident Matching Program (NRMP) [92] was established in 1952, to match graduating medical residents to hospitals in the US. Analogous matching schemes exist in Canada [86] and Japan [91]. A similar process was used until recently to match medical graduates to Foundation Programme places in Scotland; the Scottish Foundation Allocation Scheme (SFAS) [36]. Moreover, a similar process is used in the context of Higher Education admission in Hungary [10, 89], Spain [65], Turkey [6] and Ireland [87, 90]. The reader is referred to [88] for details of matching practices in a number of practical contexts throughout Europe.

The size of these matching programmes can vary tremendously: the SFAS programme typically involved approximately 750 applicants and 50 hospitals annually; in the Hungarian Higher Education allocation process in 2011, 140,953 applicants participated [89]; in 2015 the NRMP process involved 52,880 residents and assigned 26,252 residents over 27,293 posts [63]; and in the case of the largest centralised allocation scheme currently known, the Chinese Higher Education matching scheme involved approximately ten million applicants in 2007 [83]. Clearly the efficiency with which an optimal matching might be obtained is of great practical interest.

In practice these *centralised matching schemes* operate by obtaining preference information from each of the agents involved and applying an algorithm to the preference data so obtained to compute a matching. A given instance of a matching problem may admit many matchings – we wish to select a matching from amongst the possible matchings that is fair, in some sense, to all of the agents involved. However, a number of definitions of fairness are possible.

A commonly-applied definition of fairness in bipartite matching problems with two sided preferences is the concept of *stability*. A matching is *stable* if no two agents prefer each other to their assignees in the matching and is *unstable* otherwise. Free-for-all markets involving a less structured process where individual agents compete for partners, such as the process used in the NRMP before 1952 in the US, have been shown to lead to problems in practice [48, 69, 70] due to their potentially unstable outcomes. The underlying abstract allocation problem that models the search for a stable matching in a centralised matching scheme such as the NRMP is the *Hospitals / Residents problem* (HR); the nomenclature of this problem derives from the NRMP context. Gale and Shapley [27] showed that a stable matching always exists in an instance of HR and provided an algorithm that outputs a stable matching in an arbitrary instance of HR in linear time.

Centralised matching schemes such as the NRMP and SFAS have had to evolve to accommodate couples who wish to be allocated to (geographically) compatible hospitals. The requirement to take into account the joint preferences of couples has been in place in the NRMP context since 1983 and since 2009 in the case of SFAS. In schemes where the agents may be involved in couples, the underlying matching problem can be modelled by the *Hospitals / Residents problem with Couples* (HRC). In contrast with HR, an instance of HRC need not admit a stable matching [67] and a number of possible definitions of stability exist [12, 31, 55]. Two key definitions of stability discussed in this thesis are *MM-stability* (described by Manlove and McDermid [55]) and *BIS-stability* (described by Biró, Irving and Schlotter [12]). (These stability concepts will be defined formally in Definitions 2.3.1 and 2.3.2 respectively in Section 2.3.)

Since the preference lists of the residents are often short in these schemes (e.g. in the SFAS context residents are asked to list eight hospitals in order of preference), we investigate the complexity of deciding whether an instance of HRC admits a stable matching when the agents' preference lists are of restricted length. We show that a polynomial-time algorithm for HRC is unlikely in several restricted cases where the length of the agents' preference lists are of bounded length. However, we present polynomial-time algorithms for finding a stable matching, or reporting that none exists, for two restricted cases of HRC in which the agents' preference lists are of very restricted length.

For problems that do not admit a polynomial-time algorithm, Integer Programming (IP) techniques can often yield exact solutions for problem instances of a useful size in a practically useful timescale. Indeed, IP is a commonly-applied technique for finding optimal solutions in the underlying abstract matching problem applicable in kidney exchange programmes [51, 74]. In this thesis we present and prove the correctness of the first IP models for finding a maximum cardinality stable matching in an arbitrary instance of HRC under two commonly-considered notions of stability applicable in the HRC context; namely MM-stability and BIS-stability [12, 55]. We further show how this model might be extended to the more general *Hospitals / Residents problem with Couples and Ties* (HRCT) context where the agents involved may be indifferent between their acceptable partners.

Finding a matching in an instance of HRC that is 'as stable as possible' [2] in a precise sense is the problem of finding a *'most stable'* matching. Clearly a stable matching, if one exists, is a 'most stable' matching. We show that it is hard to approximate a 'most stable' matching in an instance of HRC. We further show that this implies that the minimisation variants of a number of NP-complete decision problems involving stable matchings are inapproximable to within a given bound, unless P=NP. Moreover, we show that this implies that a class of minimisation problems related to SAT and k-colouring must also be similarly hard to approximate. Further, we present and prove the correctness of an IP model for MIN BP HRC, the problem of finding a maximum cardinality 'most stable' matching in an instance of HRC.

We provide an empirical analysis of the IP models for HRC and MIN BP HRC applied to random instances reflecting the properties of the instances arising from the SFAS application. We present data on the performance of the models as we vary the size of the instance; the percentage of residents in the instance involved in couples; the number of hospitals in the instance; and the length of the residents' preference lists. Further, we examine the output when the IP model for HRCT is applied to real world instances arising from the SFAS application in the years 2010-2012.

In the Higher Education admission schemes in Hungary [10, 89], Spain [65], Turkey [6], Chile [64] and Ireland [90, 87], the applicants express preferences over acceptable colleges. However, the colleges assign each acceptable applicant an integer value or *score*, typically a measure of academic aptitude derived from exam results. Higher scoring applicants are preferable to colleges. In these schemes the outcome is represented by a *set of score limits* containing an integer value or *score limit* for each college. An *assignment* is *induced* from a set of score limits by assigning each applicant to the first college on his preference list at which he meets the score limit. In matching problems involving scores a corresponding definition of fairness is applied and we seek a *set of stable score limits* [10]. A set of score limits is *stable* if no college may lower its score limit, all other score limits remaining unchanged, without becoming over-subscribed in the assignment induced from the reduced set of score limits. The underlying abstract matching problem model for these schemes is the *Hospitals / Resident problem with Stable Score Limits and Ties* (HR SLT). In this context ties may arise in the case that a number of applicants achieve the same score at a given hospital; thus the hospital is indifferent between these applicants.

In the Hungarian Higher Education Allocation scheme there are additional restrictions placed on the feasible matchings. In this thesis we consider two of these restrictions in detail. First, we consider the restriction in which a hospital must have a minimum number of assignees in a matching, specified by a *lower quota*. The problem of finding a stable matching in this context is termed the *Hospitals / Residents problem with Stable Score Limits, Ties and Lower Quotas* (HR LQ SLT). Then we consider the restriction in which a coalition of hospitals may share a *common upper quota* – this is a bound on the total number of assignees to which the members of the coalition as a whole may be assigned in a matching. This problem is referred to as the *Hospitals / Resident problem with Stable Score Limits, Ties and Common Upper Quotas* (HR CQ SLT). In this thesis we present and prove the correctness of the first IP models for finding sets of stable score limits in HR LQ SLT and HR CQ SLT instances reflecting the properties of the instances arising in the Hungarian Higher Education Allocation scheme.

Another bipartite matching problem of interest in this thesis is motivated by the allocation of trainee teachers to training places in Slovakia. Trainee teachers studying at P.J. Šafárik University in Košice, Slovakia study two separate subjects selected from amongst a number of subjects on offer at the university e.g. Mathematics, Physics, History and Geography. As part of their training each trainee teacher is required to take part in supervised teaching of classes at real schools under the supervision of experienced and suitably-qualified teachers. Further, the trainee must be able to teach both subjects under appropriate supervision at the same school. In this thesis we describe an IP model for the underlying abstract matching problem model, the *Teachers Allocation Problem* (TAP) [18] and present data obtained from the application of the model for TAP applied to real data from the allocation process for trainee teachers at P.J. Šafárik University for the Spring 2013/14 allocation.

1.2 Thesis Outline

In Chapter 2 we review the literature on stable matching problems with a particular focus on the algorithmic aspects of those problem variants involved in this thesis.

In Chapter 3 we present a range of complexity results for restrictions of HRC imposing upper bounds on the length of the preference lists of agents and/or assuming structural properties of these preference lists. For example we show that the problem of deciding whether a stable matching exists is NP-complete even if each couple finds acceptable exactly one hospital pair. NP-completeness also holds even if each couple and hospital has a preference list of length at most two and there are no single residents. On the other hand we give polynomial time algorithms for the cases: (i) when the length of each hospital's preference list is at most one; and (ii) when the length of each couple's joint preference list is at most one, and the length of the preference list of each hospital and individual resident is at most two. As far as structural restrictions are concerned we show that the decision problem is NP-complete even if the preference lists of each resident, couple and hospital is derived from a master list of individual hospitals, pairs of hospitals and single residents respectively. Another structural restriction where NP-completeness holds relates to the case where the agents in the instance form a dual market, in a specified sense. Moreover, these results for structural restrictions hold even if the preference lists are of bounded length.

In Chapter 4 we present the first IP model for finding a maximum cardinality stable matching in an arbitrary instance of HRC under MM-stability. First, we present and prove the correctness of an IP model for finding a maximum cardinality stable matching in an instance of HR and then show how this model can be extended to the HRC context under MM-stability.

We then demonstrate by means of a concrete example how an instance of the IP model for HRC is constructed from an instance of HRC. We further show how the HRC model might be extended to HRCT, the variant of HRC in which an agent might be indifferent between agents in its preference list. Finally we present and prove the correctness of an IP model for finding a 'most stable' matching in an instance of MIN BP HRC, the minimisation variant of HRC.

In Chapter 5 we demonstrate how BIS-stability and MM-stability differ by means of a pair of example instances. Further, we present a cloning methodology for HRC that can be used to construct an instance of one-to-one HRC from an instance of many-to-one HRC such that the MM-stable matchings in the many-to-one instance correspond to the MM-stable matchings in the one-to-one instance. We provide a small counter example that proves that this cloning method is not applicable under BIS-stability. Finally, we present the first IP model for finding a maximum cardinality stable matching in an arbitrary instance of HRC under BIS-stability.

In Chapter 6 we present data from an empirical evaluation of an implementation of the IP models described in Chapters 4 and 5 for finding a maximum cardinality stable matching in an arbitrary instance of HRC or reporting that no stable matching exists. We show how the performance of the model and the properties of a maximum cardinality stable matching vary as we modify a range of parameters in the constructed instances. Further, we provide data obtained from the application of the IP model for finding a maximum cardinality stable matching in an instance of HRCT to real world instances arising from the SFAS application for the years 2010, 2011 and 2012.

In Chapter 7 we present data from an empirical evaluation of an implementation of the IP model for MIN BP HRC under MM-stability described in Chapter 4, and an implementation of the corresponding MIN BP HRC model under BIS-stability derived from the HRC model presented in Chapter 5. The models in these experiments each find a maximum cardinality 'most stable' matching in an arbitrary instance of MIN BP HRC under the corresponding stability definition. We show how the performance of the model and the properties of a 'most stable' matching vary as we modify a range of parameters in the constructed instances.

In Chapter 8 we describe the first IP models for HR SLT variants with upper and lower quotas and also the HR SLT variant in which residents may express preferences over unordered pairs of hospitals. We present and prove the correctness of three IP models for finding a minimal set of stable score limits in instances of an HR SLT variant – a minimal set of stable score limits has the property that the sum over the score limits in the set is minimal taken over all sets of stable score limits admitted by the instance. (By construction these models can only output a *minimal set of stable score limits* in an instance of the corresponding problem and are designated Type A models.) The three models for which we present Type A models are: HR SLT; the *Hospitals / Residents problem with Stable Score Limits, Ties and Paired Applications*; and the *Hospitals / Residents problem with Stable Score Limits, Ties* *and Common Upper Quotas*.

Further, we present and prove the correctness of two models for finding a set of stable score limits in an instance of an HR SLT variant where the optimal sets of score limits obtained from the model may have properties other than minimality. By amending the structure of a Type A model we can apply an alternative objective function that allows us to choose a measure of optimality other than minimality. These models are designated Type B models. We present two Type B IP models for finding a set of stable score limits $\hat{t}$ in an arbitrary instance of HR SLT such that the assignment induced from $\hat{t}$ is of maximum cardinality taken over all of the matchings induced from the sets of stable score limits in the instance. The two problems for which we present Type B models are: HR SLT; and *Hospitals / Residents problem with Ties, Lower Quotas and Stable Score Limits* (HR LQ SLT).

In Chapter 9 we describe and prove the correctness of IP models for the NP-complete problems, TAP and STABLE TAP. We show empirical data from the application of the model for TAP to the process of allocating Trainee teachers studying at P.J. Šafárik University in Košice, Slovakia for the Spring 2013/14 allocation. Further, we demonstrate how the IP model for the TAP problem may be adapted to the STABLE TAP context, where the applicants and schools have a preference ordering over their acceptable partners. Further, we show how complexity results from the TAP context allow us to prove the NP-completeness of the problem of deciding whether an arbitrary graph involving paired vertices admits a complete matching.

In Chapter 10 we present a framework for classifying certain minimisation problems with NP-complete decision variants and use this to prove that such minimisation problems are inapproximable to within a given bound, unless P=NP. We first consider the measure function with respect to which approximation is defined in stable matching problems. For minimisation problems π such as the minimisation variants of stable matching problems, the measure of an optimal solution may be zero – in this special case the performance guarantee of an approximation algorithm for π as described in Section 2.4 is not well defined. We describe an adjusted measure function that naturally extends the previous measure function for minimisation problems and moreover leads to a well-defined notion of a performance guarantee in the special case of minimisation problems having an optimal solution with a measure of zero. By defining the approximation ratio when solving minimisation problems with respect to this new adjusted measure function we present a generalisation of the proof of the inapproximability of $(2, 2)$-MIN BP HRC described in Section 3.3.2 to show that a general class of minimisation problems having an NP-complete decision variant are inapproximable to within a given bound, unless P=NP.

Chapter 2

Literature Review and Preliminary Definitions

2.1 The Stable Marriage problem

An instance of the Stable Marriage problem (SM) [27, 31, 75] involves two set of agents; a set $\{m_1, \ldots, m_n\}$ containing *men* and a set $\{w_1, \ldots, w_n\}$ containing *women*. Each agent has a strict linear order over all agents of the opposite gender, a *preference list*. A one-to-one *matching* between men and women is sought, which is a set of man-woman pairs such that each man and woman appear in exactly one pair. A matching is *stable* if no man and woman prefer one another to their partner in the matching. Given an instance I of SM, a man m_i $(1 \leq i \leq n)$ and a woman w_j $(1 \leq j \leq n)$, if M is a stable matching in I and $(m_i, w_j) \in M$, then w_j is a *stable partner* of m_i in I and vice versa.

Gale and Shapley [27] described a linear-time algorithm (GS) for finding a stable matching in an instance of SM; their algorithm can be understood as a sequence of proposals from one set of agents to another. The GS algorithm applied with the men proposing outputs the *man-optimal stable matching*. In this matching each man is assigned the best partner he could achieve in any stable matching, his best stable partner. By having the women play the role of proposers in the algorithm, exactly analogous results for women are produced, resulting in the *woman-optimal stable matching* in which each woman is assigned her best stable partner. It may be the case in some instances that the man-optimal stable matching and the woman-optimal stable matching are one and the same matching; such an instance admits exactly one stable matching. However, in general stable marriage instances may admit many stable matchings and the number of stable matchings can grow exponentially with the size of the instance [39].

Gale and Shapley proved that their algorithm proceeds in at most $n^2 - 2n + 2$ proposal steps

[27]. Knuth established in his book on stable marriage problems that the time complexity of the GS algorithm is $O(n^2)$ and asked if a more efficient algorithm was possible [46]. Ng and Hirschberg proved that $\Omega(n^2)$ is a lower bound on the complexity of an algorithm for finding a stable matching in an instance of SM [58].

McVitie and Wilson [56] described a recursive algorithm for finding a stable matching in an arbitrary instance I of SM and by extension presented an $O(n^3|S|)$ algorithm for producing all stable matchings in I, where S is the set of stable matchings in I. A more efficient $O(n^2 + n|S|)$ algorithm exists [31] that will output the set of all stable matchings admitted by an arbitrary SM instance.

2.1.1 The Stable Marriage Problem with Incomplete Preference Lists

A natural extension of SM considers the case where agents may find some members of the opposing set unacceptable. In this case each agent expresses a preference list over some subset of the members of the opposite sex, its *acceptable* partners. Any member of the opposite sex who is not acceptable is considered *unacceptable*. This extension of SM is the *Stable Marriage Problem with Incomplete Lists* (SMI). Given an instance I of SMI and a matching M in I, we denote m_i's assigned partner in M by $M(m_i)$; if m_i is unassigned in M, then $M(m_i) = \emptyset$. Similarly, we denote w_j's assigned partner in M by $M(w_j)$. Following the definition used in [27], a matching M in an instance of SMI is *stable* if it admits no *blocking pair*.

Definition 2.1.1. *A* blocking pair *in an instance of* SMI *consists of a mutually acceptable man-woman pair* (m_i, w_j) *such that both of the following hold:*

(i) either m_i *is unassigned in* M*, or* m_i *prefers* w_j *to* $M(m_i)$ and*;*

(ii) either w_j *is unassigned in* M*, or* w_j *prefers* m_i *to* $M(w_j)$*.*

In any instance of SMI all of the stable matchings assign exactly the same number of agents [28]. Further, Gale and Sotomayor showed that the men and women in an instance of SMI are each partitioned into two sets, those who are assigned in all stable matchings and those who are assigned in no stable matching [28].

2.2 The Hospitals / Residents problem

The *Hospitals / Residents problem* (HR) is a many-to-one generalisation of SMI. An instance of HR involves two sets of agents – a set $R = \{r_1, \ldots, r_{n_1}\}$ containing *residents* and a set

$H = \{h_1, \ldots, h_{n_2}\}$ containing *hospitals*. Each resident $r_i \in R$ expresses a linear preference over some subset of the hospitals in H, his *preference list*. The hospitals on r_i's preference list are *acceptable* to r_i, all other hospitals being *unacceptable* to r_i. Each hospital $h_j \in H$ expresses a linear preference over those residents who find it acceptable, these residents are *acceptable* to h_j, all other residents are *unacceptable*. Further, each hospital $h_j \in H$ has a positive integral *capacity*, c_j, the maximum number of residents to which it may be assigned.

Clearly, the preferences expressed in this fashion are reciprocal: if a resident $r_i \in R$ is acceptable to a hospital $h_j \in H$, then h_j is also acceptable to r_i, and vice versa. A many-to-one *matching* between residents and hospitals is sought. A matching M is a set of acceptable resident-hospital pairs such that each resident appears in at most one pair and each hospital h_j appears in at most c_j pairs. If $(r_i, h_j) \in M$, then r_i is said to be *assigned* to h_j in M, if r_i does not appear in a pair in M, then r_i is *unassigned*. Similarly, if $(r_i, h_j) \in M$, then h_j is said to be assigned r_i in M. A hospital assigned fewer residents than its capacity in M is *undersubscribed* in M, moreover a hospital having no assignees in M is unassigned in M. We denote by $M(h_j)$ the set containing h_j's assignees in M and we denote by $M(r_i)$ the assignee of r_i, if r_i is assigned in M.

Following the definition used in [27], a matching M is *stable* if it admits no *blocking pair*.

Definition 2.2.1. *A* blocking pair *consists of a mutually acceptable resident-hospital pair* (r_i, h_j) *such that both of the following hold:*

(i) either r_i is unassigned, or r_i prefers h_j to $M(r_i)$ and*;*

(ii) either h_j is undersubscribed in M, or h_j prefers r_i to at least one member of $M(h_j)$.

Were such a blocking pair to exist, they could form a private arrangement outside of the matching, undermining the integrity of the matching [67].

It is known that every instance of HR admits at least one stable matching and such a matching may be found in time linear in the size of the instance [27]. Also, for an arbitrary HR instance I, any resident assigned in one stable matching in I is assigned in all stable matchings in I. Moreover any hospital that is undersubscribed in a stable matching in I is assigned exactly the same set of residents in every stable matching in I [28, 68, 67]. In the *resident-optimal stable matching* each resident is assigned the best partner he could achieve in any stable matching. Analogously, the *hospital-optimal stable matching* has each hospital assigned the best partners it could receive in any stable matching.

2.3 The Hospitals / Residents problem with Couples

2.3.1 Fundamental definitions

Centralised matching schemes have had to evolve to accommodate couples who wish to be allocated to (geographically) compatible hospitals. The requirement to take into account the joint preferences of couples has been in place in the NRMP context since 1983 and since 2009 in the case of SFAS. In schemes in which the agents may be involved in couples, the underlying allocation problem can be modelled by the so-called *Hospitals / Residents problem with Couples* (HRC).

As in the case of HR, an instance of HRC involves a set $H = \{h_1, \ldots, h_{n_2}\}$ containing *hospitals* and a set $R = \{r_1, \ldots, r_{n_1}\}$ containing *residents*. The residents in R are partitioned into two sets, S and S'. The set S contains *single* residents and the set S' contains those residents involved in *couples*. There is a set $C = \{(r_i, r_j) : r_i \in S', r_j \in S'\}$ of *couples* such that each resident in S' belongs to exactly one pair in C.

Each single resident $r_i \in S$ expresses a linear preference order over his acceptable hospitals. Each pair of residents $(r_i, r_j) \in C$ expresses a joint linear preference order over a subset A of $H \times H$ where $(h_p, h_q) \in A$ represents the joint assignment of r_i to h_p and r_j to h_q. The hospital pairs in A represent those joint assignments that are *acceptable* to (r_i, r_j), all other joint assignments being *unacceptable* to (r_i, r_j).

Each hospital $h_j \in H$ expresses a linear preference order over those residents who find h_j acceptable, either as a single resident or as part of a couple. As in the HR case, each hospital $h_j \in H$ has a positive integral *capacity*, c_j.

A many-to-one *matching* between residents and hospitals is sought, which is defined as for HR with the additional restriction that each couple (r_i, r_j) is either unassigned, meaning that both r_i and r_j are unassigned, or assigned to some pair (h_k, h_l) that (r_i, r_j) find acceptable. As in the case of HR, we seek a *stable* matching, which guarantees that no resident and hospital, and no couple and pair of hospitals, have an incentive to deviate from their assignments and become assigned to each other. We denote by $M(h_j)$ the set containing h_j's assignees in M (where $M(h_j) = \emptyset$ if h_j has no assignees) and we denote by $M(r_i)$ the assignee of r_i, if r_i is assigned in M (where $M(r_i) = \emptyset$ if r_i is unassigned in M).

Roth [67] considered stability in the HRC context although he did not define the concept explicitly. Whilst Gusfield and Irving [31] defined stability in HRC, their definition neglected to deal with the case that both members of a couple may wish to be assigned to the same hospital. Manlove and McDermid [55] extended their definition to deal with this possibility. Henceforth, we refer to Manlove and McDermid's stability definition as *MM-stability*. We now define this concept formally.

Definition 2.3.1. *A matching M is MM-stable if none of the following holds:*

1. *The matching is blocked by a hospital h_j and a single resident r_i, as in the classical* HR *problem as defined in Definition 2.2.1.*
2. *The matching is blocked by a couple (r_i, r_j) and a hospital h_k such that* either
   (a) *(r_i, r_j) prefers $(h_k, M(r_j))$ to $(M(r_i), M(r_j))$, and h_k is either undersubscribed in M or prefers r_i to some member of $M(h_k)\backslash\{r_j\}$* or
   (b) *(r_i, r_j) prefers $(M(r_i), h_k)$ to $(M(r_i), M(r_j))$, and h_k is either undersubscribed in M or prefers r_j to some member of $M(h_k)\backslash\{r_i\}$*
3. *The matching is blocked by a couple (r_i, r_j) and (not necessarily distinct) hospitals $h_k \neq M(r_i)$, $h_l \neq M(r_j)$; that is, (r_i, r_j) prefers the joint assignment (h_k, h_l) to $(M(r_i), M(r_j))$, and* either
   (a) *$h_k \neq h_l$, and h_k (respectively h_l) is either undersubscribed in M or prefers r_i (respectively r_j) to at least one of its assigned residents in M;* or
   (b) *$h_k = h_l$, and h_k has at least two free posts in M, i.e., $c_k - |M(h_k)| \geq 2$;* or
   (c) *$h_k = h_l$, and h_k has one free post in M, i.e., $c_k - |M(h_k)| = 1$, and h_k prefers at least one of r_i, r_j to some member of $M(h_k)$;* or
   (d) *$h_k = h_l$, h_k is full in M, h_k prefers r_i to some $r_s \in M(h_k)$, and h_k prefers r_j to some $r_t \in M(h_k)\backslash\{r_s\}$.*

More recently Drummond et al. [22] described a notion of stability that is very closely related to MM-stability. In fact, for a given instance I of HRC, the set of stable matchings admitted by I under Drummond et al.'s definition of stability is a subset of the set of stable matchings admitted under MM-stability. A further stability definition due to Biró, Irving and Schlotter [12] (henceforth *BIS-stability*) ensures that if a single resident $r_i \in R$ is not assigned to a hospital $h_j \in H$, then all h_j's assignees in M are strictly preferable to r_i. Moreover, if a couple (r_i, r_j) $(r_i \in R, r_j \in R)$ is not assigned to a hospital pair (h_j, h_j) $(h_j \in H)$, then all h_j's assignees in M are strictly preferable to the worse of r_i and r_j according to h_j. We now define BIS-stability formally as follows.

Definition 2.3.2. *A matching M is BIS-stable if none of the following holds:*

1. *The matching is blocked by a hospital h_j and a single resident r_i, as in the classical* HR *problem as defined in Definition 2.2.1.*
2. *The matching is blocked by a hospital h_k and a resident r_i who is coupled, say with r_j; such that* either

(a) (r_i, r_j) *prefers* $(h_k, M(r_j))$ *to* $(M(r_i), M(r_j))$ *and* either

(i) $h_k \neq M(r_j)$ *and* h_k *is either undersubscribed in* M *or prefers* r_i *to some member of* $M(h_k)$ or

(ii) $h_k = M(r_j)$ *and* h_k *is either undersubscribed in* M *or prefers both* r_i *and* r_j *to some member of* $M(h_k) \setminus \{r_j\}$

(b) (r_i, r_j) *prefers* $(M(r_i), h_k)$ *to* $(M(r_i), M(r_j))$ *and* either *(i) or (ii) as above adapted to symmetric case*

3. *The matching is blocked by a couple* (r_i, r_j) *and (not necessarily distinct) hospitals* $h_k \neq M(r_i)$ *and* $h_l \neq M(r_j)$*; that is,* (r_i, r_j) *prefers the joint assignment* (h_k, h_l) *to* $(M(r_i), M(r_j))$*, and* either

(a) $h_k \neq h_l$*, and* h_k *(respectively* h_l*) is either undersubscribed in* M *or prefers* r_i *(respectively* r_j*) to at least one of its assignees in* M*;* or

(b) $h_k = h_l$*, and* h_k *has at least two free posts in* M or

(c) $h_k = h_l$*, and* h_k *has one free post in* M *and both* r_i *and* r_j *are preferred by* h_k *to some member of* $M(h_k)$ and

(d) $h_k = h_l$*,* h_k *is full in* M *and* either

(i) h_k *prefers each of* r_i *and* r_j *to some* $r_p \in M(h_k)$ *who is a member of a couple with some* $r_q \in M(h_k)$*,*

(ii) *the least preferred resident among* r_i *and* r_j *(according to* h_k*) is preferred by* h_k *to two members of* $M(h_k)$*.*

It is notable that, for an arbitrary instance I of HRC in which hospitals may have a capacity of greater than one, an MM-stable matching need not be BIS-stable and vice versa. The instances described in Section 5.2 demonstrate this. In the restriction of HRC in which each hospital has capacity one, BIS-stability and MM-stability are both equivalent to the stability definition given by Gusfield and Irving [31] since no couple (r_i, r_j) may express a preference for a hospital pair (h_k, h_k).

The *Hospitals / Residents Problem with Couples and Ties* (HRCT) is a generalisation of HRC in which hospitals (respectively single residents or couples) may find some subsets of their acceptable residents (respectively hospitals or hospital pairs) equally preferable. Residents (respectively hospitals or hospital pairs) that are found equally preferable by a hospital (respectively resident or couple) are *tied* with each other in the preference list of that hospital (respectively resident or couple). The stability concepts given by Definitions 2.3.1 and 2.3.2 remain unchanged in the HRCT context.

In contrast with HR, an instance of HRC need not admit a stable matching [67]. Moreover, an instance of HRC may admit stable matchings of differing sizes [4]. It is known that the problem of deciding whether a stable matching exists in an instance of HRC is NP-complete, even in the restricted case in which there are no single residents and each hospital has capacity one [58, 66]. Nguyen and Vohra [60] showed that it is always possible to find a stable matching in an instance of HRC if the capacities of the hospitals may be adjusted by at most three.

Residents' Preferences

$$(r_1, r_2) : \quad (h_1, h_2)$$
$$r_3 : \quad h_1 \;\; h_2$$

Hospitals' Preferences

$$h_1 : \quad r_1 \;\; r_3$$
$$h_2 : \quad r_3 \;\; r_2$$

Figure 2.1: An instance of HRC that admits no stable matching due to Biró and Klijn. [14]

The instance shown in Figure 2.1 demonstrates that an instance of HRC need not admit a stable matching. Let I be the instance of HRC shown in Figure 2.1 where each hospital has capacity one. Clearly, I admits exactly three non-empty matchings, namely

$$M_1 = \{(r_1, h_1), (r_2, h_2)\}$$

$$M_2 = \{(r_3, h_1)\}$$

$$M_3 = \{(r_3, h_2)\}.$$

None of these matchings are stable. Resident r_3 blocks M_1 in I with h_2, couple (r_1, r_2) blocks M_2 in I with (h_1, h_2) and resident r_3 blocks M_3 in I with h_1. Thus, I admits no stable matching.

The instance shown in Figure 2.2 demonstrates that an instance of HRC can admit stable matchings of differing sizes. Let I be the instance of HRC shown in Figure 2.2 where each hospital has capacity one. Thus, I admits exactly two stable matchings, namely

$$M_1 = \{(r_1, h_1), (r_4, h_2), (r_2, h_3), (r_3, h_4)\}$$

$$M_2 = \{(r_2, h_1), (r_3, h_2)\}$$

and these matchings are not of the same size.

Residents' Preferences

(r_1, r_4) : (h_1, h_2)
(r_2, r_3) : (h_1, h_2) (h_3, h_4)

Hospitals' Preferences

h_1 : r_1 r_2
h_2 : r_3 r_4
h_3 : r_2
h_4 : r_3

Figure 2.2: An instance of HRC that admits stable matchings of differing size due to Biró and Klijn. [14]

In many practical applications of HRC the residents' preference lists are short. Let (α, β, γ)-HRC denote the variant of HRC in which each single resident's preference list contains at most α hospitals, each couple's preference list contains at most β pairs of hospitals and each hospital's preference list contains at most γ residents. We will abbreviate (α, α, γ)-HRC to (α, γ)-HRC. Manlove and McDermid [55] showed that $(3, 6)$-HRC is NP-complete.

A further restriction of HRC is HRC DUAL MARKET, defined as follows. Given an instance I of HRC, let the set containing the first members of each couple in I be $R_1 \subseteq R$, and the set containing the second members of each co'uple in I be $R_2 \subseteq R$. Further, let the set of acceptable partners of the residents in R_1 in I be $H_1 \subseteq H$ and the set of acceptable partners of the residents in R_2 in I be $H_2 \subseteq H$. We define I to be an instance of HRC DUAL MARKET consisting of the two disjoint markets $R_1 \cup H_1$ and $R_2 \cup H_2$ if in I, $H_1 \cap H_2 = \emptyset$ and no single resident has acceptable partners in both H_1 and H_2. The problem of deciding whether an instance of HRC DUAL MARKET admits a stable matching is also known to be NP-complete [59] even if the instance contains no single residents and each hospital has capacity one.

Since the existence of an efficient algorithm for finding a stable matching, or reporting that none exists, in an instance of HRC is unlikely, in practical applications such as SFAS and NRMP, stable matchings are found by applying heuristics [12, 71]. However, neither the SFAS heuristic, nor the NRMP heuristic guarantee to terminate and output a stable matching, even in instances where a stable matching does exist. Hence, a method which guarantees to find a maximum cardinality stable matching in an arbitrary instance of HRC, where one exists, might be of considerable interest. For further results on HRC the reader is referred to [14] and [48].

2.3.2 The Hospitals / Residents problem with Paired Applications

In the Hungarian Higher Education admission process applicants can apply for pairs of programmes in the case of teachers studies, e.g. when they want to become a teacher in both maths and physics. In this setting with paired applications, if an applicant is not admitted to a pair of hospitals, or to any better hospitals (or pair of hospitals) in his preference list, then either of the hospitals involved in the pair must have filled its quota with better applicants. We now define this problem formally as the *Hospitals / Residents problem with Paired Applications* (HR PA) as follows. An instance I of HR PA involves a set $R = \{r_1, r_2, \ldots r_{n_1}\}$ containing residents, a set $H^S = \{h_1, h_2, \ldots h_{n_2}\}$ containing individual hospitals and a set H^P containing acceptable unordered pairs of hospitals $\{h_{j_1}, h_{j_2}\}$ $(1 \leq j_1 \leq n_2, 1 \leq j_2 \leq n_2, h_{j_1} \in H^S, h_{j_2} \in H^S, h_{j_1} \neq h_{j_2})$. Further let $H^* = H^S \cup H^P$ be the set of all possible options over which a resident might express preferences.

Each resident $r_i \in R$ has a strictly ordered preference list of length $l(r_i)$ over the members of H^*. We refer to an element of r_i's preference list as an *application*. We say that $h_j \in pref(r_i, p)$ if the application at position p on r_i's preference list involves h_j, either as an application to the single hospital h_j or as an application to some pair $\{h_j, h_k\} \in H^P$ for some k $(1 \leq k \leq n_2, k \neq j)$. Further, each hospital $h_j \in H$ has capacity c_j, the maximum number of residents that h_j may be assigned.

We define stability in the HR PA context as follows:

Definition 2.3.3. *A matching M in an instance of* HR PA *is stable if there exists no resident $r_i \in R$ and application $pref(r_i, p)$ $(1 \leq p \leq l(r_i))$ such that*

(i) *r_i is unassigned or prefers $pref(r_i, p)$ to $M(r_i)$* and

(ii) *For each $h_j \in pref(r_i, p)$ either h_j is undersubscribed or prefers r_i to some member of $M(h_j)$.*

We now demonstrate that an instance of HR PA need not admit a stable matching and further that an instance of HR PA can admit stable matchings of differing sizes.

The instance shown in Figure 2.3 demonstrates that an instance of HR PA need not admit a stable matching. Let I be the instance of HR PA shown in Figure 2.3 where each hospital has capacity one. Thus I admits exactly three non-empty matchings, namely

$$M_1 = \{(r_1, h_1), (r_1, h_2)\}$$

$$M_2 = \{(r_2, h_1)\}$$

Residents' Preferences	
r_1 :	$\{h_1, h_2\}$
r_2 :	h_1 h_2

Hospitals' Preferences	
h_1 :	r_1 r_2
h_2 :	r_2 r_1

Figure 2.3: An instance of HR PA that admits no stable matching

$$M_3 = \{(r_2, h_2)\}.$$

Clearly, none of these matchings is stable. Resident r_2 blocks M_1 in I with h_2, couple (r_1, r_2) blocks M_2 in I with (h_1, h_2) and resident r_2 blocks M_3 in I with h_1. Thus, I admits no stable matching.

Residents' Preferences	
r_1 :	$\{h_1, h_2\}$
r_2 :	$\{h_1, h_2\}$ $\{h_3, h_4\}$

Hospitals' Preferences	
h_1 :	r_1 r_2
h_2 :	r_2 r_1
h_3 :	r_2
h_4 :	r_2

Figure 2.4: An instance of HR PA that admits stable matchings of differing sizes

The instance shown in Figure 2.4 demonstrates that an instance of HR PA can admit stable matchings of differing sizes. Let I be the instance of HR PA shown in Figure 2.3 where each hospital has capacity one. Thus, I admits exactly two stable matchings, namely

$$M_1 = \{(r_1, h_1), (r_1, h_2), (r_2, h_3), (r_2, h_4)\}$$

$$M_2 = \{(r_2, h_1), (r_2, h_2)\}$$

and these matchings are clearly not of the same size.

2.4 'Most stable' matchings and optimisation problems

Given an instance I of HRC that does not admit a stable matching, a natural question is to ask whether there is some other matching that might be the best alternative amongst the matchings admitted by I. Roth [69, 70] suggested that instability in the outcome of an allocation process gives participants greater incentive to circumvent formal procedures – it follows that we might seek to minimise the amount of instability in any alternative matching selected. We suggest that it is natural to seek a matching that admits the minimum number of blocking pairs taken over all of the matchings admitted by the instance. Eriksson and Häggström [25] have suggested that the number of blocking pairs admitted by a matching is a meaningful way of measuring the degree of instability of a matching. Abraham et al. [2] described matchings with the minimum number of blocking pairs in unsolvable instances of SR as being 'as stable as possible'. We define a *'most stable'* matching as follows: a matching M in I is a *'most stable'* matching in I if it admits the minimum number of blocking pairs taken over all of the matchings admitted by I. Clearly a stable matching in I, if one exists, is a 'most stable' matching in I.

Determining whether I admits a matching with zero blocking pairs is clearly a decision problem – the answer is either yes or no. However the search for a 'most stable' matching in I is an optimisation problem since we seek a matching where the minimum number of blocking pairs is some non-negative integer. We now introduce and formally define the notation used in the discussions of optimisation problems that follow in this work. We begin with the definition of an optimisation problem.

Definition 2.4.1. *An optimisation problem π is a tuple ($\mathcal{I}$, SOL, m, $GOAL$) where:*

- *$\mathcal{I}$ is a set of instances;*
- *SOL is a function that maps a given instance $I \in \mathcal{I}$ to a set of feasible solutions $SOL(I)$;*
- *m is a measure function that associates a non-negative integer $m(I, S)$ with a feasible solution $S \in SOL(I)$ for a given instance $I \in \mathcal{I}$;*
- *$GOAL$ is the objective of the optimisation problem which is either to maximise or to minimise.*

Given an instance $I \in \mathcal{I}$, the optimal measure for I, *denoted by $opt(I)$ is defined as follows:*

$$opt(I) = GOAL\{m(I, S) : S \in SOL(I)\}.$$

An optimal solution *for I is a feasible solution $S^* \in SOL(I)$ such that $m(I, S^*) = opt(I)$. If $GOAL = \min$, π is called a minimisation problem, otherwise π is referred to as a maximisation problem.*

The class NPO denotes the class of NP optimisation problems whose decision version is in the class NP. Let π be an NPO problem and further let I be an instance of π. An *approximation algorithm A* for π guarantees to return a feasible solution $A(I)$ for any instance I of π. If π is a maximisation problem the *performance guarantee* of A with respect to I is defined as follows:

$$R_A(I) = \frac{opt(I)}{m(I, A(I))}. \tag{2.1}$$

Similarly, if π is a minimisation problem the performance guarantee of A with respect to I is defined as follows:

$$R_A(I) = \frac{m(I, A(I))}{opt(I)}. \tag{2.2}$$

An algorithm A is a *c-approximation algorithm* for π and has performance guarantee c for some constant c, if $R_A(I) \leq c$ for all instances $I \in \mathcal{I}$. If π admits a c-approximation algorithm, then we say that π is *c-approximable*. If it is possible to prove that no such algorithm can exist, then π is *inapproximable* to within a factor of c.

Having introduced the required notation we now formally define MIN BP HRC, the problem of finding a 'most stable' matching in an instance of HRC.

MIN BP HRC

Instance: An instance I of HRC;
Feasible solutions: All the matchings admitted by I;
Measure: The number of blocking pairs admitted by a matching in I;
Goal: min;
Optimisation version: Minimise the number of blocking pairs taken over all of the matchings admitted by I;
Decision version: Is there a matching in I that admits no blocking pairs?.

Let (α, β, γ)-MIN BP HRC denote the variant of MIN BP HRC in which each single resident's preference list contains at most α hospitals, each couple's preference list contains at most β pairs of hospitals and each hospital's preference list contains at most γ residents. We will abbreviate (α, β, γ)-MIN BP HRC to (α, γ)-MIN BP HRC. Further, we shall denote by MIN BP HRC DUAL MARKET the minimisation variant of HRC DUAL MARKET.

2.5 The Hospitals / Residents problem with Ties

The Hospitals / Residents problem with Ties (HRT) is the variant of HR in which the preference lists of the agents may contain *ties*. If a group of residents are involved in a tie in a hospital's preference list, then the hospital is indifferent between the residents involved in the tie. Similarly if a group of hospitals are involved in a tie in a resident's preference list, then the resident is indifferent between all of the hospitals involved in the tie.

In this context a number of stability definitions exist. The definition which attracts most interest in the literature is *weak stability*, defined as follows.

Definition 2.5.1. *A matching is* weakly stable *if it admits no* blocking pair. *Following the definition used in [48], a blocking pair in an instance of* HRT *is a resident-hospital pair* (r_i, h_j) *such that both of the following conditions hold:*

(i) either r_i *is unassigned, or* r_i *strictly prefers* h_j *to* $M(r_i)$ and*;*

(ii) either h_j *is undersubscribed in the matching, or* h_j *strictly prefers* r_i *to at least one member of* $M(h_j)$

Every instance of HRT admits a weakly stable matching – a weakly stable matching can be obtained by breaking each of the ties in an arbitrary fashion to obtain an instance of HR and subsequently applying the Gale Shapley algorithm to the resulting HR instance. However, the order in which the ties are broken will lead to differing instances of HR and it is known that the instances created in this manner may lead to weakly stable matchings of differing sizes [50]. Manlove et al. [50] showed that the problem of finding a maximum cardinality weakly stable matching in an instance of HRT, MAX HRT, is NP-hard and the same is true for the problem of finding a minimum cardinality weakly stable matching. Further, they showed that arbitrarily breaking the ties in this fashion gives an approximation algorithm with a performance guarantee of two.

Halldórsson et al. [32] showed that it is NP-hard to approximate MAX HRT to within δ for some $\delta > 1$ even if each hospital has capacity one, the residents' preference lists are of length at most seven, the hospitals' preference lists are of length at most four and the preference lists of the hospitals and residents are derived from a master list of residents and hospitals respectively. Irving et al. [42] showed that it remains NP-hard to approximate MAX HRT to within a constant factor even if each hospital has capacity one, the residents' preference lists are of length at most three and the hospitals' preference lists are of length at most four.

Halldórsson et al. proved that if the preference lists of the residents are of unbounded length, the residents' preference lists are strictly ordered and the hospitals' preference lists are strictly ordered or contain a tie of length two, then it is NP-hard to approximate MAX HRT to within $21/19 - \varepsilon$ for any $\varepsilon > 0$, unless P=NP. Further, Yanagisawa [81] showed that it is NP-hard to approximate MAX HRT to within $33/29$ in the case where there are ties on either the hospitals' or the residents' preference lists, but the ties are of length at most two. A number of approximation algorithms exist for finding a maximum cardinality stable matching in an instance of MAX HRT, see for example [45, 54]. The best currently described algorithm achieving an approximation guarantee of $3/2$ [45].

The instance of HRT shown in Figure 2.5 due to Roth [67] admits no resident-optimal weakly stable matching. In an instance of HRT a group of residents (respectively hospitals) surrounded by $[\ldots]$ in the preference list of a hospital (respectively resident) indicates a group of residents (respectively hospitals) tied in the preference list of the hospital (respectively resident).

Residents

$r_1: \quad h_1 \; h_2$
$r_2: \quad h_1 \; h_2$

Hospitals

$h_1 : 1 : \quad [\, r_1 \quad r_2 \,]$
$h_2 : 1 : \quad r_1 \quad r_2$

Figure 2.5: An instance of HRT that admits no resident-optimal stable matching.

Let I be the instance of HRT shown in Figure 2.5. The instance I admits exactly two stable matchings, namely

$$M_1 = \{(r_1, h_1), (r_2, h_2)\}$$

$$M_2 = \{(r_1, h_2), (r_2, h_1)\}$$

Clearly, neither M_1 nor M_2 is an optimal matching with respect to the residents.

2.6 The Hospitals / Residents problem with Lower / Common Quotas

In Sections 2.6.1 and 2.6.2 we present two additional models for HR variants that have additional restrictions on the number of residents which a hospital may be assigned. First we formally define the Hospitals / Residents problem with Lower Quotas (HR LQ) in Section 2.6.1. Then we formally define the Hospitals / Residents problem with Common Upper Quotas (HR CQ) in Section 2.6.2.

2.6.1 The Hospitals / Residents problem with Lower Quotas

An instance I of HR LQ extends an instance of HR by the following additional restriction. As well as a positive integral capacity c_j^+, the maximum number of residents to which it may be assigned in any stable matching, each hospital $h_j \in H$ has a positive integral lower quota c_j^- $(0 \leq c_j^- \leq c_j^+)$, the minimum number of residents to which it may be assigned in any stable matching. A hospital with at least $\min\{1, c_j^-\}$ or greater assignees is *open*. Otherwise, the hospital is *closed* and can accept no assignees.

A matching is *stable* if it admits no *blocking pair* and also admits no *blocking coalition*.

Definition 2.6.1. *Following the definition used in [11], a matching M in an instance of HR LQ is stable if both of the following conditions hold:*

(i) (Blocking Pair) *There exists no acceptable resident-hospital pair (r_i, h_j) such that r_i is unassigned or prefers h_j to $M(r_i)$* and *h_j is an open hospital which is either undersubscribed in the matching, or prefers r_i to at least one member of $M(h_j)$.*

(ii) (Blocking Coalition) *There exists no closed hospital h_j and set of at least c_j^- residents and no more than c_j^+ residents such that each of the residents is either unassigned or prefers h_j to their assigned partner.*

Biró et al. [11] showed that the problem of deciding whether an instance of HR LQ admits a stable matching is NP-complete even in the case that each hospital has upper and lower quota equal to three, but the complexity remains open if each lower quota is at most two or if each upper quota (and hence each lower quota) is at most two. The instance shown in Figure 2.6 due to Manlove [49] demonstrates that an instance of HR LQ may admit stable matchings of differing sizes.

Residents' Preferences

$r_1 :\ h_2\ h_1$
$r_2 :\ h_1$
$r_3 :\ h_1\ h_2$

Hospitals' Preferences

$h_1 : 3 : 3 :\ r_1\ r_2\ r_3$
$h_2 : 2 : 2 :\ r_1\ r_3$

Figure 2.6: An instance of HR LQ that admits two stable matchings of differing sizes [49].

Let I be the instance of HR LQ shown in Figure 2.6. The instance I admits exactly two stable matchings, namely $M_1 = \{(r_1, h_1), (r_2, h_1), (r_3, h_1)\}$ and $M_2 = \{(r_1, h_2), (r_3, h_2)\}$. We demonstrate the stability of M_1 and M_2 as follows. First consider M_1. Since $(r_2, h_1) \in M_1$ and $(r_3, h_1) \in M_1$, then clearly h_2 cannot meet its lower quota of two since r_1 is the single remaining resident in I that prefers h_2 to his assigned partner in M_1. Hence, M_1 must be a stable matching in I. Now consider M_2. Since $(r_1, h_2) \in M_2$ then clearly h_1 cannot meet its lower quota of three since there are only two other residents in I that prefer h_1 to their assigned partners in M_2. Hence, M_2 is a stable matching in I.

2.6.2 The Hospitals / Residents problem with Common Upper Quotas

We next consider the variant of HR in which coalitions of hospitals may share common upper quotas. We define this variant as the Hospitals / Residents problem with Common Upper Quotas (HR CQ). In HR CQ we generalise the concept of a hospital's capacity as follows. A coalition of hospitals may share a common upper quota, such that the total number of residents admitted to hospitals in this group may not exceed this quota. Let $H^* = \{H_1, H_2, \ldots H_{n_3}\}$ where each $H_k \in H^*$ is a subset of H representing a set of hospitals sharing a common upper quota. For each coalition $H_k \in H^*$ let u_k be the common upper quota for the coalition (in this model the capacity of each individual hospital is represented by ensuring that $\{h_j\} \in H^*$).

In our HR CQ problem model we assume that each $H_k \in H^*$ has a preference list which is a *master list* of preferences for each $h_j \in H_k$. Thus the preferences in I satisfy the following properties where $P(H_k)$(respectively $P(h_j)$) represents the ordered list of residents on H_k's preference list (respectively h_j's preference list):

(i) $P(H_k) = \bigcup_{h_j \in H_k} P(h_j)$

(ii) $P(h_j)$ is constructed from $P(H_k)$ by removing those residents from $P(h_k)$ who are not in $P(h_j)$

(iii) For any two coalitions $H_k \in H^*$ and $H_l \in H^*$, let $H' = H_k \cap H_l$ and for any pair of residents r_s and r_t such that $r_s \in P(H')$ and $r_t \in P(H')$, r_s precedes r_t in $P(H_k)$ if and only if r_s precedes r_t in $P(H_l)$.

Definition 2.6.2. *A matching in an instance of* HR CQ *is* stable *if it admits no* blocking pair. *Following the definition used in [11], a blocking pair consists of a mutually acceptable resident-hospital pair* (r, h) *such that both of the following conditions hold:*

(i) either r *is unassigned, or* r *prefers* h *to* $M(r)$ and;

(ii) for each $H_k \in H^*$ *such that* $h \in H_k$*, either* H_k *is undersubscribed or prefers* r *to at least one of its assigned residents.*

Biró et al. [11] showed that an instance of HR CQ need not admit a stable matching and they further showed that the stable matchings admitted by an instance of HR CQ need not be of the same size. Further, Biró et al. [11] also showed that the problem of deciding whether an instance of HR CQ admits a stable matching is NP-complete even if every coalition has upper quota one, no coalition contains more than two hospitals and each hospital appears in at most three coalitions (including the coalition consisting of only that hospital).

We say that the coalitions of hospitals in an instance of HR CQ are *nested* if for any two coalitions $H_k \in H^*$ and $H_l \in H^*$, if $H_k \bigcap H_l \neq \emptyset$ then either $H_k \subseteq H_l$ or $H_l \subseteq H_k$. Biró et al. [11] showed that a stable matching always exists in an instance of HR CQ in which the coalitions of hospitals are nested and further described a polynomial time algorithm which finds a stable matching in an instance HR CQ in which the coalitions of hospitals are nested.

Residents' Preferences	
r_1 :	h_1 h_7
r_2 :	h_5 h_2
r_3 :	h_3 h_6
r_4 :	h_4 h_7

Individual Hospitals' Preferences	
h_1 : 1 :	r_1
h_2 : 1 :	r_2
h_3 : 1 :	r_3
h_4 : 1 :	r_2
h_5 : 1 :	r_2
h_6 : 1 :	r_3
h_7 : 1 :	r_1 r_4

Coalitions' Master Lists	
$\{h_1, h_2\}$: 1 :	r_2 r_1
$\{h_2, h_3\}$: 1 :	r_2 r_3
$\{h_4, h_5\}$: 1 :	r_2 r_4
$\{h_5, h_6\}$: 1 :	r_3 r_2

Figure 2.7: An instance of HR CQ that admits exactly two stable matchings, neither of which is optimal with respect to the residents.

Let I be the instance of HR CQ shown in Figure 2.7. The instance I admits exactly two stable matchings, namely $M_1 = \{(r_1, h_1), (r_2, h_5), (r_3, h_3), (r_4, h_7)\}$ and $M_2 = \{(r_1, h_7), (r_2, h_2), (r_3, h_6), (r_4, h_4)\}$. We demonstrate the stability of M_1 and M_2 as follows. First we consider the matching M_1. Assume M_1 is blocked in I by (r_4, h_4). Now we have that the coalition $\{h_4, h_5\}$ is fully subscribed and moreover is assigned to its first preference r_2, and hence (r_4, h_4) cannot block M_1 in I, a contradiction. Since all other residents are assigned to their first preference in M_1, M_1 must be stable in I.

Now consider the matching M_2. Assume, M_2 is blocked in I by (r_1, h_1). Now we have that the coalition $\{h_1, h_2\}$ is fully subscribed and moreover is assigned to its first preference r_2, and hence (r_1, h_1) cannot block M_2 in I, a contradiction. Assume, M_2 is blocked in I by (r_2, h_5). Now we have that the coalition $\{h_5, h_6\}$ is fully subscribed and moreover is assigned to its first preference r_3, and hence (r_2, h_5) cannot block M_2 in I, a contradiction. Assume, M_2 is blocked in I by (r_3, h_3). Now we have that the coalition $\{h_2, h_3\}$ is fully subscribed and moreover is assigned to its first preference r_2, and hence (r_3, h_3) cannot block M_2 in I, a contradiction. Since no other residents may improve their allocation in M_2 it follows that M_2 is stable in I. Clearly, neither M_1 nor M_2 is an optimal matching with respect to the residents.

2.7 The Hospitals / Residents problem with Stable Score Limits

2.7.1 Fundamental definitions

A number of centralised matching schemes take into account *scores* that are assigned to applicants by their acceptable positions. This is the case, for example, in the context of Higher Education admission in Hungary [10, 89], Spain [65], Turkey [6] and Ireland [90, 87]. The outcome of such schemes may be represented by a set of *score limits* for the programmes involved, where the score limit of a programme represents the lowest score that would allow an applicant to be assigned to that programme. In this section we describe the underlying abstract allocation problem for such schemes involving score limits, defined as the *Hospitals / Residents problem with Stable Score Limits* (HR SL) [13] (we remark that Biró and Kiselgof [13] refer to this problem as the College Admission problem with Stable Score Limits).

An instance of HR SL consists of two sets of agents: a set $R = \{r_1, r_2, \ldots, r_{n_1}\}$ containing *residents*, and a set $H = \{h_1, h_2, \ldots, h_{n_2}\}$ containing *hospitals*. Each resident $r_i \in R$ expresses a linear preference over some subset of the hospitals, his *acceptable* hospitals; all other hospitals being *unacceptable*. The *acceptable* resident partners for each hospital $h_j \in H$ are those residents who find h_j acceptable, all other residents being *unacceptable* to h_j. Let $s_{i,j}$ be a non-negative integer representing the score of r_i at h_j and further let $\bar{s}_j$ represent the maximum possible score a resident might achieve at h_j. We say that h_j *prefers* r_{i_1} to r_{i_2} if $s_{i_1,j} > s_{i_2,j}$. If $s_{i_1,j} = s_{i_2,j}$ then r_{i_1} and r_{i_2} are in a *tie* in h_j's list. Further, every hospital has a positive integral *capacity*, c_j, the maximum number of residents that may be assigned to h_j.

Given an instance of HR SL the objective is to compute for each hospital an integer value representing the score limit at that hospital. Let $\hat{t} = \{t_1, \ldots, t_{n_2}\}$ be a set of score limits in

I, where t_j is the score limit at h_j $(1 \leq j \leq n_2)$. The range of score limits for a hospital $h_j \in H$ ranges from zero to $\bar{s}_j + 1$, since the score limit can be higher than the maximal score (when all students are rejected, e.g. because more students have maximal score than the upper quota). An *assignment* M is *induced* in I from $\hat{t}$ and the residents' preferences as follows. For a given resident r_i, let h_j be the first hospital on r_i's preference list such that $s_{i,j} \geq t_j$. Add (r_i, h_j) to M.

In most real applications any ties between residents at a hospital are broken in some manner. In the context of higher education admission in Turkey [6] any ties are broken according to the applicants' dates of birth, whilst in Spain [65] a very fine grained scoring system is used so ties are highly unlikely in practice. In Ireland ties are broken by assigning applicants a random number and breaking ties by ordering tied applicants according to these random numbers. We will discuss the Hungarian context in more detail below. We now define stability in the context where there are no ties amongst the acceptable residents for any of the hospitals. A set of score limits, $\hat{t}$ is *feasible* if no hospital exceeds its capacity in the assignment induced from $\hat{t}$.

Definition 2.7.1. *A set of score limits, $\hat{t}$ in an instance of* HR SL *is* stable *if both of the following hold:*

(i) $\hat{t}$ is feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}$ then $\hat{t}'$ is infeasible.

We induce an instance I' of HR from the residents' scores and preferences in an instance of HR SL as follows. For each h_j in I' create a preference list in which h_j's acceptable residents are strictly ordered by decreasing score. Balinski and Sönmez [6] showed that in the absence of ties the stable matchings in I' are in one-to-one correspondence with the sets of stable score limits in I.

Let $\hat{t}^1 = \{t^1_1, \ldots, t^1_{n_2}\}$ and $\hat{t}^2 = \{t^2_1, \ldots, t^2_{n_2}\}$ be two sets of score limits in I. We say that $\hat{t}^1$ is *better than* $\hat{t}^2$ for the residents in I if $t^1_j \leq t^2_j$ for all j $(1 \leq j \leq n_2)$ and $t^1_j < t^2_j$ for at least one j $(1 \leq j \leq n_2)$. The residents would consider $\hat{t}^1$ better than $\hat{t}^2$ since every resident would be admitted to the same hospital or a better hospital under $\hat{t}^1$ than under $\hat{t}^2$.

In the Hungarian higher education admission process, ties between residents are never broken. Either all residents tied at a hospital are admitted to that hospital or none of them are; this is known as an *equal treatment* policy. We denote this variant of HR SL in which residents may be tied at a hospital by HR SLT for the avoidance of ambiguity. We now define stability in the more general HR SLT setting.

We first describe a feasibility concept that is similar to the notion of feasibility described in the HR SL context. Let $\hat{t}^H$ be a set of score limits in I and let M be the assignment in I induced from $\hat{t}^H$. Then $\hat{t}^H$ is *H-feasible* if no hospital exceeds its capacity in M.

Definition 2.7.2. *A set of score limits $\hat{t}^H$ in an instance of* HR SLT *is* H-stable *if both of the following hold:*

(i) $\hat{t}^H$ is H-feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}^H$ then $\hat{t}'$ is H-infeasible

Correspondingly, we define *L-feasibility* and *L-stability* as follows. Let $\hat{t}^L = \{t_1, \ldots, t_{n_2}\}$ be a set of score limits in I and let M be the assignment in I induced from $\hat{t}^L$. Given any $h_j \in H$, let $\hat{t}^{L,j} = \{t_1^j, \ldots, t_{n_2}^j\}$ be the set of score limits obtained from $\hat{t}^L$ by setting $t_k^j = t_k$ for all k $(1 \leq k \leq n_2, k \neq j)$ and $t_j^j = t_j + 1$. Let M^j be the assignment in I induced from $\hat{t}^{L,j}$. Then $\hat{t}^L$ is *L-feasible* if, for each $h_j \in H$, either $|M(h_j)| \leq c_j$ or $|M^j(h_j)| < c_j$.

Definition 2.7.3. *A set of score limits $\hat{t}^L$ in an instance of* HR SLT *is* L-stable *if both of the following hold:*

(i) $\hat{t}^L$ is L-feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}^L$ then $\hat{t}'$ is L-infeasible.

We induce an instance I' of HRT from the residents' scores and preferences in an instance I of HR SLT as follows. In I' create a preference list for each h_j such that h_j's acceptable residents are (not necessarily strictly) ordered by non-increasing score.

Biró and Kiselgof [13] showed that in any HR SLT instance I there exists a set of H-stable score limits that are the best possible set of H-stable score limits for the residents in I with respect to the "better than" relation. These are called the *resident optimal* H-stable score limits in I, denoted by $\hat{t}_0^H$. Similarly, they showed that there exists a set of score limits that are the worst possible set of H-stable score limits for the residents in I with respect to the "better than" relation. These are called the *resident pessimal* H-stable score limits in I denoted by $\hat{t}_z^H$. Biró and Kiselgof [13] showed that a resident optimal and resident pessimal set of L-stable score limits for the residents in I, denoted $\hat{t}_0^L$ and $\hat{t}_z^L$ respectively also exist. Moreover, $\hat{t}_0^H$ and $\hat{t}_0^L$ are upper and lower bounds respectively for the resident optimal stable score limits in I under any random tie breaking mechanism in I. Similarly, $\hat{t}_z^H$ and $\hat{t}_z^L$ are upper and lower bounds respectively for the resident pessimal stable score limits in I under any random tie breaking mechanism in I.

Residents' Preferences
r_1 : h_1
r_2 : h_1

Hospitals' Preferences
$h_1 : 1 : \quad [r_1\ r_2]$

Figure 2.8: An example instance showing that the assignment M induced from a set of H-stable score limits in an instance I of HR SLT need not be weakly stable matching in I', the instance of HRT induced from the residents' scores and preferences in I.

Intuitively, in the assignment induced from a set of H-stable score limits no hospital can be oversubscribed. However, in the assignment induced from a set of L-stable score limits a hospital may be over-subscribed as a result of accepting a group of residents tied at the hospital. This can only be the case if the hospital would undersubscribed if the residents involved in the tie were all refused entry. In the Chilean Higher Education admission scheme [64] L-stable score limits are used and in the Hungarian Higher education admission scheme H-stable score limits are used.

Let $\hat{t}^H$ be a set of H-stable score limits in an instance I of HR SLT and let M be the assignment in I induced from $\hat{t}^H$. Let I' be the instance of HRT induced from the residents' scores and preferences in I. It need not be the case under H-stability that M is a weakly stable matching in I'. Moreover, under L-stability it need not be the case that M is a feasible matching in I'. We demonstrate by the following example these observations.

In I let $s_{1,1} = 1$ and $s_{2,1} = 1$ and thus both r_1 and r_2 achieve a score of 1 at h_1. Further, let h_1 have a capacity of one and a score limit of two. Clearly this is a set of stable score limits in I since reducing the score limit of h_1 by one would create a new set of score limits, t' and h_1 would be over-subscribed in the induced assignment from t'. Now, the assignment M induced in I from the residents' scores and preference lists and the hospital score limits has no residents assigned to h_1 and $M = \emptyset$.

Figure 2.8 shows the instance I' of HRT induced from the residents' score and preferences in I. Clearly, M is not a weakly stable matching in I' since h_1 is undersubscribed and both r_1 and r_2 are unassigned and would prefer h_1.

Now we consider the same instance under the L-stability definition. Let h_1 have a score limit of one. Clearly this is an L-stable set of score limits in I. The assignment in I induced under L-stability from the residents' score and preferences and the score limits of the hospitals in I would be $M = \{(r_1, h_1), (r_2, h_1)\}$ since both r_1 and r_2 achieve the score limit at h_1. However, now we have that h_1 is oversubscribed in M and M is not a feasible matching in

I'.

2.7.2 The Hospitals / Residents problem with Paired Applications and Score Limits

In the Hungarian Higher Education admission process applicants can apply to pairs of programmes in the case of teachers studies, e.g. when they want to become a teacher in both maths and physics. In this setting with paired applications if a resident is not admitted to a pair of hospitals, or to a better hospital (or pair of hospitals) in his preference list then the resident must not achieve the score-limit at one or more of the hospitals in the pair. In this section we formally define HR with Paired Applications, Ties and Stable Score limits(HR PA SLT). This problem is similar to HRCT. However, unlike in an instance of HRCT, where each resident is either single or a member of a couple but not both, in HR PA SLT, a resident is always single, but may express preferences over both single hospitals and pairs of hospitals in the same preference list.

An instance I of HR PA SLT consists of a set containing residents $R = \{r_1, r_2, \ldots, r_{n_1}\}$, a set containing single hospitals $H^S = \{h_1, h_2, \ldots, h_{n_2}\}$ and a set, H^P, containing acceptable unordered pairs of hospitals $\{h_j, h_k\}$ $(1 \leq j \leq n_2, 1 \leq k \leq n_2, h_j, h_k \in H^S, h_j \neq h_k)$. Further let $H^* = H^S \cup H^P$ be the set of all possible options over which a resident might express his preferences; the *applications*. Each resident r_i, has a (not necessarily strictly ordered) preference list of length $l(r_i)$ consisting of preferences over some subset of the applications in H^*. Let $s_{i,j}$ be a non-negative integer representing the score of r_i at h_j and further let $\bar{s}_j$ represent the maximum possible score a resident might achieve at hospital h_j. We say that h_j *prefers* r_{i_1} to r_{i_2} if $s_{i_1,j} > s_{i_2,j}$. If $s_{i_1,j} = s_{i_2,j}$ then r_{i_1} and r_{i_2} are in a *tie* at h_j. Further, each hospital $h_j \in H$ has capacity $c_j \geq 1$, the maximum number of residents that h_j may be assigned.

We say that $h_j \in \mathit{pref}(r_i, p)$ if the application at position p on r_i's preference list involves h_j, either as an application to the single hospital h_j or as an application to some pair $\{h_j, h_k\} \in H^P$. Let $\hat{t}$ be a set of score limits in I. An *assignment* M is *induced* in I from $\hat{t}$ as follows. For a given resident r_i, let p represent the first position on r_i's preference list where r_i achieves the score limit at each $h_j \in \mathit{pref}(r_i, p)$. Add (r_i, h_j) to M for each $h_j \in \mathit{pref}(r_i, p)$.

Let $\hat{t} = \{t_1, \ldots, t_{n_2}\}$ be a set of score limits in I, where t_j is the score limit at h_j $(1 \leq j \leq n_2)$. The range of score limits for a hospital $h_j \in H$ ranges from zero to $\bar{s}_j + 1$, since the score limit can be higher than the maximal score (when all students are rejected, e.g. because more students have maximal score than the upper quota). Given an instance of HR PA SLT the objective is to compute for each h_j an integer value t_j representing the score limit at h_j

such that $\hat{t}$ represents a *set of stable score limits*.

We first define stability in HR PA SL, the restriction of HR PA SLT in which the hospital preference lists are strictly ordered i.e. no two residents achieve the same score at any hospital. A set of score limits $\hat{t}$ is *feasible* if no hospital exceeds its capacity in the assignment induced from $\hat{t}$.

Definition 2.7.4. *A set of score limits $\hat{t}$ in an instance of* HR PA SL *is* stable *if both of the following hold:*

(i) $\hat{t}$ *is feasible* and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}$ then $\hat{t}'$ is infeasible.

Now, this definition of stability may be extended to the corresponding definitions of *H-feasibility* and *H-stability* and also *L-feasibility* and *L-stability* in HR PA SLT as defined for HR SL in the Section 2.7 for the HR SLT context.

2.7.3 The Hospitals / Residents problem with Lower Quotas and Score Limits

An instance of the *Hospitals / Residents problem with Ties, Lower Quotas and Stable Score Limits* (HR LQ SLT) consists of two sets of agents: a set $R = \{r_1, r_2, \ldots, r_{n_1}\}$ containing *residents*, and a set $H = \{h_1, h_2, \ldots, h_{n_2}\}$ containing *hospitals*. Each resident $r_i \in R$ expresses a linear preference over some subset of the hospitals, his *acceptable* hospitals; all other hospitals being *unacceptable* to r_i. The *acceptable* resident partners for each hospital $h_j \in H$ are those residents who find h_j acceptable; all other residents being *unacceptable* to h_j. Let $s_{i,j}$ be a non-negative integer representing the score of r_i at h_j and further let $\bar{s}_j$ represent the maximum possible score a resident might achieve at h_j. We say that h_j *prefers* r_{i_1} to r_{i_2} if $s_{i_1,j} > s_{i_2,j}$. If $s_{i_1,j} = s_{i_2,j}$ then h_j is indifferent between r_{i_1} and r_{i_2} and thus r_{i_1} and r_{i_2} are in a *tie* at h_j. Each $h_j \in H$ has a positive integral *upper quota* c_j^+, (equivalent to its capacity in the HR context) the maximum number of assignees it may receive in an assignment. Further, each h_j has an integral *lower quota* c_j^- $(0 \leq c_j^- \leq c_j^+)$ representing the minimum number of assignees h_j may receive in an assignment. A hospital with at least $\min\{1, c_j^-\}$ or greater assignees is *open*. Otherwise, the hospital is *closed* and has no assignees.

Let $\hat{t} = \{t_1, \ldots, t_{n_2}\}$ be a set of score limits in I, where t_j is the score limit at h_j $(1 \leq j \leq n_2)$. The range of score limits for a hospital $h_j \in H$ ranges from zero to $\bar{s}_j + 1$, since the score limit can be higher than the maximal score (when all students are rejected, e.g. because more students have maximal score than the upper quota). An *assignment* M is *induced* in I from $\hat{t}$ and the residents' preferences as follows. For a given resident $r_i \in R$, let h_j be the first hospital on r_i's preference list such that $s_{i,j} \geq t_j$. Add (r_i, h_j) to M. Given an instance of HR LQ SLT the objective is to compute for each h_j an integer value t_j representing the score limit at h_j such that $\hat{t}$ represents a *set of stable score limits*.

We first define stability in HR LQ SL, the restriction of HR LQ SLT where the hospital preference lists are strictly ordered, i.e. no two residents are assigned the same score at any hospital. A set of score limits $\hat{t}$ is *feasible* if no hospital exceeds its capacity in the assignment induced from $\hat{t}$. A set of score limits, $\hat{t}$ is *feasible* if for each $h_j \in H$, $|M(h_j)| \in \{0\} \bigcup \{c_j^-, c_j^- + 1, \ldots, c_j^+\}$ where M is the assignment induced from $\hat{t}$.

Definition 2.7.5. *A set of score limits, $\hat{t}$ in an instance of* HR LQ SL *is* stable *if both of the following hold:*

(i) $\hat{t}$ is feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}$ then $\hat{t}'$ is infeasible.

We now describe *H-feasibility* and *H-stability* in the HR LQ SLT context. Let $\hat{t}^H$ be a set of score limits in I and let M be the assignment induced in I from $\hat{t}^H$. Then $\hat{t}^H$ is *H-feasible* if for each $h_j \in H$, $|M(h_j)| \in \{0\} \bigcup \{c_j^-, c_j^- + 1, \ldots, c_j^+\}$.

Definition 2.7.6. *A set of score limits, $\hat{t}^H$ in an instance of* HR LQ SLT *is* H-stable *if both of the following hold:*

(i) $\hat{t}^H$ is H-feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}^H$ then $\hat{t}'$ is H-infeasible.

Correspondingly, we define *L-feasibility* and *L-stability* in the HR LQ SLT context as follows. Let $\hat{t}^L = \{t_1, \ldots, t_{n_2}\}$ be a set of score limits in I and let M be the assignment in I induced from $\hat{t}^L$. Given any $h_j \in H$, let $\hat{t}^{L,j} = \{t_1^j, \ldots, t_{n_2}^j\}$ be the set of score limits obtained from $\hat{t}^L$ by setting $t_k^j = t_k$ for all k $(1 \leq k \leq n_2, k \neq j)$ and $t_j^j = t_j + 1$. Let M^j be the assignment in I induced from $\hat{t}^{L,j}$. Then $\hat{t}^L$ is *L-feasible* if, for each $h_j \in H$, either $c_j^- \leq |M(h_j)| \leq c_j^+$ or $c_j^- \leq |M^j(h_j)| > c_j^+$.

Definition 2.7.7. *A set of score limits, $\hat{t}^L$ in an instance of* HR LQ SLT *is* L-stable *if both of the following hold:*

(i) $\hat{t}^L$ *is L-feasible* and*;*

(ii) if $\hat{t}'$ *is a set of score limits obtained by reducing exactly one of the positive score limits in* $\hat{t}^L$ *then* $\hat{t}'$ *is L-infeasible.*

2.7.4 The Hospitals / Residents problem with Common Quotas and Score Limits

An instance of the *Hospitals / Residents problem with Common Quotas and Stable Score Limits* (HR CQ SLT) consists of two sets of agents: a set $R = \{r_1, r_2, \ldots, r_{n_1}\}$ containing *residents*, and a set $H = \{h_1, h_2, \ldots, h_{n_2}\}$ containing *hospitals*. Each resident $r_i \in R$ expresses a linear preference over some subset of the hospitals, his *acceptable* hospitals; all other hospitals being unacceptable to r_i.

As in the model HR CQ described in Section 2.6.2, coalitions of hospitals may share common upper quotas, meaning that the total number of residents admitted to the hospitals in each such coalition may not exceed the upper quota of the coalition. Let $H^* = \{H_1, H_2, \ldots H_{n_3}\}$ where each $H_k \in H^*$ is a subset of H representing a set of hospitals sharing a common upper quota. For each coalition $H_k \in H^*$ let u_k be the common upper quota for the coalition (in this model the capacity of each individual hospital $h_j \in H$ is represented by ensuring that $\{h_j\} \in H^*$).

Let $s_{i,j}$ be a non-negative integer representing the score of r_i at h_j and further let $\bar{s}_j$ represent the maximum possible score a resident might achieve at h_j. We say that h_j *prefers* r_{i_1} to r_{i_2} if $s_{i_1,j} > s_{i_2,j}$. If $s_{i_1,j} = s_{i_2,j}$ then r_{i_1} and r_{i_2} are in a *tie* in h_j's list.

In the more general HR CQ problem model described in Section 2.6.2 we place restrictions on the preferences of the agents and coalitions of agents in an instance. We formally stated in Section 2.6.2 that for any two coalitions $H_k \in H^*$ and $H_l \in H^*$, and for any pair of residents r_s and r_t such that $\{r_s, r_t\} \subseteq P(H')$ where $H' = H_k \bigcap H_l$, r_s precedes r_t in $P(H_k)$ if and only if r_s precedes r_t in $P(H_l)$.

However, in the HR CQ SLT context we insist that for each resident r_i who finds a given hospital h_j acceptable, where h_j appears in coalition H_k, r_i achieves the same score at each hospital in H_k. This allows us to define the notation describing r_i's score at the coalition H_k, denoted by $S_{i,k}$. Let $\bar{S}_k = \max\{\bar{s}_j : h_j \in H_k\}$ represent the maximum possible score a resident might achieve at H_k. For each coalition $H_k \in H^*$ if $h_{j_1} \in H_k$ and $h_{j_2} \in H_k$ then for any r_i acceptable to both h_{j_1} and h_{j_2}, $s_{i,j_1} = s_{i,j_2} = S_{i,k}$.

Let $\hat{t} = \{t_1, \ldots, t_{n_3}\}$ be a set of score limits in I, where t_k is the score limit at H_k $(1 \leq k \leq n_3)$. The score limit for a coalition $H_k \in H^*$ ranges from 0 to $\bar{S}_k + 1$, since the score limit can be higher than the maximal score (when all students are rejected, e.g. because more students have maximal score than the common upper quota). An *assignment* M is *induced* in I from $\hat{t}$ and the residents' preferences as follows. For a given resident r_i, let h_j be the first hospital on r_i's preference list such that $S_{i,k} \geq t_k$ for all $H_k \in H^*$ such that $h_j \in H_k$. Add (r_i, h_j) to M.

We first define stability in the HR CQ SL context where there are no ties amongst the acceptable residents for any of the coalitions. A set $\hat{t}$ of score limits is *feasible* if no coalition exceeds its capacity in the induced assignment of $\hat{t}$.

Definition 2.7.8. *A set of score limits, $\hat{t}$ in an instance of* HR CQ SL *is* stable *if both of the following hold:*

(i) $\hat{t}$ is feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}$ then $\hat{t}'$ is infeasible.

We now define stability in the HR CQ SLT context where two residents may achieve the same score at a given coalition. We first define H-feasibility and H-stability. Let $\hat{t}^H$ be a set of score limits in I and let M be the assignment in I induced from $\hat{t}^H$. Then $\hat{t}^H$ is *H-feasible* if no coalition exceeds its capacity in M.

Definition 2.7.9. *A set of score limits, $\hat{t}^H$ in an instance of* HR CQ SLT *is* H-stable *if both of the following hold:*

(i) $\hat{t}^H$ is H-feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}^H$ then $\hat{t}'$ is H-infeasible

Correspondingly, we define *L-feasibility* and *L-stability* in the HR CQ SLT context as follows. Let $\hat{t}^L = \{t_1, \ldots, t_{n_3}\}$ be a set of score limits in I and let M be the assignment in I induced from $\hat{t}^L$. Given any $H_k \in H^*$, let $\hat{t}^{L,k} = \{t_1^k, \ldots, t_{n_3}^k\}$ be the set of score limits obtained from $\hat{t}^L$ by setting $t_p^k = t_p$ for all p $(1 \leq p \leq n_3, p \neq k)$ and $t_k^k = t_k + 1$. Let M^k be the assignment in I induced from $\hat{t}^{L,k}$. Then $\hat{t}^L$ is *L-feasible* if, for each $H_k \in H^*$, either $|M(H_k)| \leq u_k$ or $|M^k(H_k)| > u_k$.

Definition 2.7.10. *A set of score limits, $\hat{t}^L$ in an instance of* HR CQ SLT *is* L-stable *if both of the following hold:*

(i) $\hat{t}^L$ is L-feasible and*;*

(ii) if $\hat{t}'$ is a set of score limits obtained by reducing exactly one of the positive score limits in $\hat{t}^L$ then $\hat{t}'$ is L-infeasible.

In the models in Chapter 8, we focus on H-stability, which is the stability concept applied in the Hungarian Higher Education Admissions scheme [10, 89]. Hence, given an instance of HR CQ SLT the objective is to compute for each $H_k \in H^*$ an integer value t_k representing the score limit at H_k such that $\hat{t}$ represents a set of *stable score limits*.

2.8 The Teachers' Allocation Problem

Trainee teachers studying at P.J. Šafárik University in Košice, Slovakia study two separate subjects selected from amongst the subjects on offer at the University for example Mathematics, Physics, History and Geography. As part of their training each trainee teacher is required to take part in supervised teaching of classes at real schools under the supervision of experienced and suitably-qualified teachers.

A requirement of this process is that each trainee teacher is able to take part in supervised teaching of both of their chosen subjects at the same school during the same term. Moreover, in order to be supervised at a particular school each trainee must have a suitably qualified supervising teacher for both of their subjects of choice at the school. Thus if a trainee teacher studying Maths and Chemistry wishes to carry out their supervised teaching assignment at a school, then that school must have a suitably-qualified supervising teacher in both Maths and Chemistry available. The underlying abstract allocation problem in this context is defined as the Teachers Allocation Problem (TAP) by Cechlárová et al. [18].

An instance I of TAP involves a set $A = \{a_1, a_2 \dots a_{n_1}\}$ containing *applicants*, a set $S = \{s_1, s_2 \dots\ s_{n_2}\}$ containing *schools* and a set $D = \{d_1, d_2 \dots\ d_{n_3}\}$ containing *subjects* (where, for example, d_1 might be maths, d_2 chemistry, etc). Each applicant $a_i \in A$ finds a subset of the schools *acceptable*, all other schools being *unacceptable* to a_i. Further, each school $s_j \in S$ finds *acceptable* those applicants that find s_j acceptable, all other applicants being *unacceptable* to s_j.

Each applicant $a_i \in A$ wishes to study a pair of distinct subjects denoted by $\{d^1(a), d^2(a)\}$ where $d^1(a) \in D$ and $d^2(a) \in D$. Each school $s_j \in S$ has a capacity for each subject $d \in D$. The vector of capacities is denoted by $c(s) = (c_1(s), \dots, c_{n_3}(s))$ where the entry of $c(s)$ corresponding to subject $d_k \in D$, denoted by $c_k(s)$, is the *partial capacity* of school s with respect to subject d_k. Here, $c_k(s_j)$ is the maximum number of applicants whose specialisation involves d_k that s_j may be assigned.

A *matching* M in I is a set of mutually acceptable applicant-school pairs where each applicant $a_i \in A$ is assigned to at most one school in M and no school $s_j \in S$ is assigned more than $c_k(s_j)$ applicants with respect to any subject $d_k \in D$. We say $M(a_i) = s_j$ if applicant a_i is assigned to school s_j and thus $(a_i, s_j) \in M$. The set of applicants assigned to a school s_j will be denoted by $M(s_j) = \{a_i \in A : (a_i, s_j) \in M\}$. We shall also denote by $M_{d_k}(s_j)$ the set of applicants assigned to $s_j \in S$ whose specialisation includes subject $d_k \in D$ and by $M_{d_p,d_q}(s_j)$ $(1 \leq p \leq n_3, 1 \leq q \leq n_3)$ the set of applicants assigned to s_j whose specialisation is exactly the pair $\{d_p, d_q\}$. We say that a school $s_j \in S$ is *full* with respect to subject d_k if $|M_{d_k}(s_j)|= c_k(s)$ $(1 \leq k \leq n_3)$ and $s_j \in S$ is *undersubscribed* with respect to subject d_k if $|M_{d_k}(s)|< c_k(s)$.

Given an instance I of TAP, let MAX TAP denote the problem of finding the largest integer k such that I admits a matching of size at least k. Further, we denote by FULL TAP the problem of deciding whether I admits an applicant complete matching. We denote by MAX TAP D the problem of deciding, given an integer k whether I admits a matching of size at least k. Cechlárová et al. [18] have shown that MAX TAP is polynomially solvable under the following non-simultaneous restrictions:

(i) The total number of subjects is two.

(ii)) The total number of subjects is three and the partial capacity of a school with respect to each subject is at most one.

We denote by (α, β)-TAP the restriction of TAP in which each applicant may list at most α acceptable schools and where the partial capacity at each school with respect to any subject is at most β. Moreover, we denote by (α, β)-MAX TAP the restriction of MAX TAP in which each applicant may list at most α acceptable schools and where the partial capacity at each school with respect to any subject is at most β. Further, given an instance I of TAP in which each applicant may list at most α acceptable schools and where the partial capacity at each school with respect to any subject is at most β we denote by (α, β)-FULL TAP the problem of deciding whether I admits an applicant complete matching. Cechlárová et al. [18] showed that $(2, 1)$-FULL TAP is polynomially solvable; however they also proved that (2,1)-MAX TAP is NP-complete.

Further, Cechlárová et al. showed that FULL TAP is NP-complete even under the following non-simultaneous restrictions:

(i) Each applicant lists at most three acceptable schools, the total number of subjects is three and no partial capacity at any school is greater than two.

(ii) Each applicant lists at most three acceptable schools, the total number of subjects is four and no partial capacity at any school is greater than one.

Further, Cechlárová et al. [18] demonstrated that MAX TAP is NP-complete even when each school is acceptable to each applicant and no partial capacity is greater than two.

2.8.1 The Teachers Allocation problem with Stability

An instance I of the Teachers Allocation problem with Stability (STABLE TAP) extends an instance of TAP as follows. Each applicant $a_i \in A$ has a *preference list* which is a linear preference over some subset of the schools, his *acceptable* schools; all other schools being *unacceptable* to a_i. Further, each school $s_j \in S$ expresses a linear preference over those applicants who find s_j acceptable, s_j's *preference list*.

A matching is *stable* if it admits no *blocking pair*. Following the definition used in [18], a blocking pair consists of a mutually acceptable applicant-school pair (a, s) defined as follows:

Definition 2.8.1. *An acceptable pair $(a_i, s_j) \notin M$ where $\{d^1(a_i), d^2(a_i)\} = \{d_p, d_q\}$ blocks M if a_i is not assigned in M or prefers s_j to $M(a_i)$ and at least one of the following conditions hold:*

(i) *s_j is undersubscribed with respect to both d_p and d_q,*

(ii) *s_j is undersubscribed in d_p (respectively d_q) and prefers a_i to at least one applicant in $M_{d_q}(s_j)$ (respectively $M_{d_p}(s_j)$),*

(iii) *s_j prefers a_i to one applicant in $M_{d_p,d_q}(s_j)$,*

(iv) *s_j prefers a_i to two different applicants a_x, a_y such that $a_x \in M_{d_p}(s)$ and $a_y \in M_{d_q}(s)$.*

Cechlárová et al. [17] showed that an instance of STABLE TAP need not admit a stable matching by the example instance shown in Figure 2.9 in which s_1 has partial capacity of one with respect to subjects one and two and partial capacity of two with respect to subject three whereas s_2 has partial capacity of one with respect to each of subjects one, two and three.

Applicants' Preferences

$a_1 : 1, 3 : \quad s_2 \;\; s_1$
$a_2 : 2, 3 : \quad s_1 \;\; s_2$
$a_3 : 1, 3 : \quad s_1$

Schools' Preferences

$s_1 : \quad a_1 \;\; a_3 \;\; a_2$
$s_2 : \quad a_2 \;\; a_1$

Figure 2.9: An instance of STABLE TAP that admits no stable matching. [17]

Let I be the instance of STABLE TAP shown in Figure 2.9. If $M(a_1) = s_2$, then either $M(s_1) = a_2$ or $M(s_1) = a_3$. If $M(s_1) = a_2$, then M is blocked by (a_3, s_1), otherwise $M(s_1) = a_3$ and M is blocked by (a_2, s_2). If $M(a_1) = s_1$, then $a_2 \in M(s_1)$ (and thus $M(s_1) = \{a_1, a_2\}$) in which case M is blocked by (a_1, s_2). Hence I admits no stable matching.

We demonstrate by the instance shown in Figure 2.10 that an instance of STABLE TAP may admit stable matchings of differing sizes.

Applicants' Preferences

$a_1 : 1, 2 : \quad s_2 \;\; s_1$
$a_2 : 1, 3 : \quad s_1 \;\; s_2$
$a_3 : 3, 4 : \quad s_1$

Schools' Preferences

$s_1 : \quad a_1 \;\; a_2 \;\; a_3$
$s_2 : \quad a_2 \;\; a_1$

Figure 2.10: An instance of STABLE TAP that admits stable matchings of differing sizes. [49]

Let I be the instance of STABLE TAP shown in Figure 2.10 where s_1 and s_2 are schools with partial capacities of one with respect to each of four subjects. Let these four subjects be d_1, d_2, d_3 and d_4. The notation $a_i : x, y : \ldots$ denotes the applicant a_i $(1 \leq i \leq 3)$ who expresses a preference for subjects p_x and p_y $(1 \leq y \leq 4, 1 \leq x \leq 4, x \neq y)$. I admits exactly two stable matchings - $M_1 = \{(a_1, s_2), (a_2, s_1)\}$ and $M_2 = \{(a_1, s_1), (a_2, s_2), (a_3, s_1)\}$. Clearly these matchings are not the same size.

Moreover, we demonstrate by the instance shown in Figure 2.11 that an instance of STABLE TAP need not admit a matching that is optimal with respect to either set of agents. Let I be the instance of STABLE TAP shown in Figure 2.11 where s_1 and s_2 are schools with partial capacities of one with respect to each of four subjects. Let these four subjects be d_1, d_2, d_3 and d_4 and as before the notation $a_i : x, y : \ldots$ denotes the applicant a_i $(1 \leq i \leq 4)$ who expresses a preference for subjects p_x and p_y $(1 \leq y \leq 4, 1 \leq x \leq 4, x \neq y)$. Let M be a stable matching in I. If a_1 is unassigned in M, then (a_1, s_1) blocks M. Further, if a_4 is unassigned in M, then (a_4, s_2) blocks M. Now suppose that a_2 is unassigned. If a_1 and a_4 are both assigned to s_1, then (a_2, s_2) blocks M. Alternatively, if a_1 and a_4 are both assigned to s_2, then (a_2, s_1) blocks M. Now suppose that both $(a_1, s_1) \in M$ and $(a_4, s_2) \in M$, then (a_1, s_2) blocks M. Alternatively, suppose $(a_1, s_2) \in M$ and $(a_4, s_1) \in M$, then (a_2, s_1) blocks M. By a similar argument a_3 cannot be unassigned in M.

Thus, all a_i $(1 \leq i \leq 4)$ must be assigned in any stable matching in I. There are exactly

Applicants' Preferences	
$a_1 : 1, 4 :$	$s_2 \; s_1$
$a_2 : 1, 3 :$	$s_1 \; s_2$
$a_3 : 2, 4 :$	$s_2 \; s_1$
$a_4 : 2, 3 :$	$s_1 \; s_2$

Schools' Preferences	
$s_1 :$	$a_1 \; a_2 \; a_3 \; a_4$
$s_2 :$	$a_4 \; a_2 \; a_3 \; a_1$

Figure 2.11: An instance of STABLE TAP that admits exactly two stable matchings, neither of which is optimal with respect to either set of agents.

two matchings in I in which all the applicants are assigned, namely $M_1 = \{(a_1, s_2), (a_2, s_1), (a_3, s_1), (a_4, s_2)\}$ and $M_2 = \{(a_1, s_1), (a_2, s_2), (a_3, s_2), (a_4, s_1)\}$. Clearly, M_1 and M_2 are the only two stable matchings admitted by I and neither M_1 nor M_2 is an optimal matching with respect to either set of agents.

Further, Cechlárová et al. [17] showed that deciding whether an instance of STABLE TAP admits a stable matching is NP-complete even when there are at most three subjects, each partial capacity of a school is at most two and the preference list of each applicant is of length at most three.

We denote the problem in which the preference lists of the schools are derived from a master list of applicants as STABLE TAP AM. Similarly, we denote the problem in which the preference lists of the applicants are derived from a master list of schools as STABLE TAP SM. Cechlárová et al [17] showed that each of STABLE TAP AM and STABLE TAP SM admits a unique stable matching which may be found in polynomial time by the application of a serial dictatorship mechanism. [1]

2.8.2 The Teachers Allocation problem with Stability and Subject Specific Preference Lists

Cechlárová et al. [17] considered a variant of STABLE TAP in which schools may rank applicants differently for different subjects. For example a school may rank an applicant higher as a mathematician than as a chemist. We denote this variant of STABLE TAP as *The Teachers*

[1]Serial dictatorship [1] is a family of algorithms for two-sided matching problems with one-sided preferences that involves ordering the applicants in some way and then, with respect to this ordering, each applicant in turn is considered and is given their most preferred post with sufficient available capacity.

Allocation problem with Stability and Subject Specific Preference Lists (STABLE TAP SS).

In STABLE TAP SS, following the definition used in [17], we define stability as follows. A matching is *stable* if it admits no *blocking pair*, a blocking pair consists of a mutually acceptable applicant-school pair (a_i, s_j) defined as follows:

Definition 2.8.2. *An acceptable pair* $(a_i, s_j) \notin M$ *where* $\{d^1(a_i), d^2 a_i\} = \{d_p, d_q\}$ *blocks* M *if* a_i *is not assigned in* M *or prefers* s_j *to* $M(a_i)$ *and at least one of the following conditions hold:*

(i) s_j *is undersubscribed with respect to both* d_p *and* d_q*,*

(ii) s_j *is undersubscribed in* d_p *(respectively* d_q*) and prefers* a_i *to at least one applicant in* $M_{d_q}(s_j)$ *(respectively* $M_{d_p}(s_j)$*)*

(iii) s_j *prefers* a_i *with respect to both subjects* d_p *and* d_q *to one applicant in* $M_{d_p,d_q}(s_j)$*,*

(iv) s_j *prefers* a_i *to two different applicants* a_{i_1}, a_{i_2} *such that* $a_{i_1} \in M_{d_p}(s_j)$ *and* $a_{i_2} \in M_{d_q}(s_j)$*.*

Cechlárová et al. [17] showed that the problem of deciding whether an instance of STABLE TAP SS admits a stable matching is NP-complete even if there are at most three subjects, each partial capacity is at most one and the preference lists of a school are derived from subject-specific master lists of applicants and the preference lists of the applicants are derived from a single master list of schools.

2.9 Exponential techniques applied to HR and its variants

Linear Programming (LP) formulations for HR and other matching problems have attracted a great deal of interest in the literature. A LP formulation is represented in canonical form as:

$$\begin{aligned} \text{minimise} \quad & c^T x \\ \text{subject to} \quad & Ax \leq b \end{aligned}$$

where x is a vector of variables, c and b represent vectors of coefficients and A is a constraint matrix of coefficients. The expression to be maximised or minimised is the *objective function*. Equivalently, an LP model can be viewed as comprising a set of variables, a set of linear inequalities in terms of those variables and an objective function. LP approaches to matching problems involve creating a set of variables representing the possible assigned partners of the agents in the instance and constructing a set of linear inequalities in terms of these variables such that there is some useful correspondence between the set of stable matchings in the matching problem instance and the feasible solutions to the set of inequalities.

Vande Vate [78] described an LP formulation for SM, the one-to-one variant of HR in which all hospitals have a capacity of one, the numbers of residents and hospitals are the same, and each of the residents finds every hospital acceptable. Vande Vate showed that the extreme points of the polyhedron described by the system of linear inequalities in his LP model represent the set of stable matchings in the instance of SM from which they are derived. Rothblum [76] generalised this model to the HR context for arbitrary instances. Baïou and Balinski [5] formulated an LP model for HR that Fleiner [26] further generalised to the many-to-many version of HR, a variant in which both hospitals and residents may have capacities exceeding one.

In more general LP models the solutions to the LP model need not assign exclusively integer values to the variables in the model. Such a solution would not represent a well-defined matching. This problem may be avoided by applying IP techniques. In IP the domain values of the variables are constrained such that only solutions in which the variables take integer values are considered feasible solutions [79, 80]. It is known that a solution to an LP can be found in a time polynomial in the size of the problem [44]. In contrast to the LP case, finding a solution to an IP model is known to be NP-hard [29, 43]. Fortunately powerful IP solvers can often be used to find solutions to instances that are of a practically useful size allowing IP techniques to be applied to real allocation problems.

Podhradskỳ [62] empirically investigated the performance of approximation algorithms for MAX SMTI (the 1-1 restriction of MAX HRT, the NP-hard problem of finding a maximum cardinality stable matching given an instance of HRT) and compared them against one another and against an IP formulation for MAX SMTI. Kwanashie and Manlove [47] described an IP model for MAX HRT and applied their model to find maximum cardinality stable matchings in real instances derived from the SFAS process in which no couples were present.

The problem of boolean satisfiability (SAT) is defined as follows: given a Boolean formula B in CNF over a set of variables V, decide whether there is an assignment of truth values to the variables in V such that B is satisfied. Cook showed that this problem is NP-complete [20]. Very recently Drummond et al. [22] demonstrated that HRC instances where each hospital has capacity one may be encoded as instances of SAT under a stability concept that may be regarded as a stronger version of MM-stability. Drummond et al. [22] also compared the performance of SAT encodings of HRC with the performance of an IP encoding of HRC. The authors found that SAT solvers were an effective method of solving HRC problems, but found that their IP models did not scale as well as the SAT encodings as the size of the problem increased.

A constraint satisfaction problem (CSP) consists of a set of variables, a set of domains for those variables and a set of constraints restricting the values that the variables may simultaneously take. A solution to a CSP is an assignment of values to the variables from their domains such that every constraint is satisfied. Manlove et al. [52] and Eirinakis et al. [23] applied CSP techniques to HR while O'Malley [61] described a CSP formulation for HRT. Subsequently, Eirinakis et al. [24] gave a generalised CSP formulation for many-to-many HR. The reader is referred to Ref. [48, Sections 2.4 & 2.5] for more information about previous work involving the application of IP and CSP techniques to allocation problems such as HR.

Chapter 3

Complexity Results for HRC

3.1 Introduction

In this chapter we present new complexity results for HRC variants, showing that deciding whether an instance of HRC admits a stable matching is NP-complete even under very severe restrictions on the length of the agents' preference lists. We begin in Section 3.2 by proving NP-completeness for the problem of deciding whether a stable matching exists in a highly restricted instance of HRC in which the length of each couple's joint preference list is exactly one and each hospital has capacity one. In Section 3.3.1 we prove that deciding whether a stable matching exists in an instance of HRC is NP-complete even if the preference list of each couple and hospital are of length at most two, there are no single residents and each hospital has capacity one.

In Section 3.3.2 we consider the complexity of MIN BP HRC, the minimisation variant of HRC, in which we seek a 'most stable' matching in an instance of HRC. It follows from Theorem 3.3.1 in Section 3.3.1 that this problem is NP-hard – by combining instances of $(2, 2)$-HRC as constructed in the proof of Theorem 3.3.1, we arrive at a gap-introducing reduction that establishes an inapproximability result for MIN BP HRC under the same restrictions as in the proof of Theorem 3.3.1 in Section 3.3.1.

In Section 3.4 we show that, given an instance of HRC, the problem of deciding whether the instance admits a stable matching is NP-complete even if the length of the preference list of each couple is at most two, the length of the preference list of each hospital is at most three, there are no single residents and each hospital has capacity one. Moreover, we show that the problem is still NP-complete even if the preference lists of all of the single residents, couples and hospitals are derived from a master list of hospitals, hospital pairs and residents respectively. In Section 3.5 we show that, the problem of deciding whether the instance admits a stable matching is NP-complete even if the length of the preference list of each couple, single resident and hospital is at most three and each hospital has capacity one. Again, we show that the result holds under the further restriction that the preference lists of all of the single residents, couples and hospitals are derived from a master list of hospitals, hospital pairs and residents respectively and the agents involved comprise a dual market.

In Section 3.6.1 we show, by reduction from $(\infty, 1, \infty)$-HRC, that, given an instance of HR PA, the problem of deciding whether the instance admits a stable matching is NP-complete even in highly restricted instances of HR PA in which each couple's joint preference lists is of length one and each hospital has capacity one.

Finally in Section 3.7, we prove that in two highly restricted variants of HRC we can find a maximum cardinality stable matching or report that no stable matching exists in polynomial-time in two cases: (i) when the length of each hospital's preference list is at most one; and (ii) when the length of each couple's joint preference list is at most one, and the length of the preference list of each hospital and individual resident is at most two.

3.2 Complexity results for $(\infty, 1, \infty)$-HRC

We now establish that the problem of deciding whether an instance of $(\infty, 1, \infty)$-HRC admits a stable matching is NP-complete.

Theorem 3.2.1. *Given an instance of $(\infty, 1, \infty)$-HRC, the problem of deciding whether there exists a stable matching is NP-complete. The result holds even if each hospital has capacity one.*

Proof. The proof of this result uses a polynomial-time reduction from a restricted version of the vertex cover problem. More specifically, let VC3 denote the problem of deciding, given a cubic graph G and an integer K, whether G contains a vertex cover of size at most K.

Deciding whether an instance of $(\infty, 1, \infty)$-HRC admits a stable matching is clearly in NP, as a given assignment may be verified to be a stable matching in polynomial time. To show NP-hardness we now present a polynomial time reduction from an instance of VC3 to an instance of $(\infty, 1, \infty)$-HRC. Let $\langle G, K\rangle$ be an instance of VC3, where $G = (V, E)$, such that $V = \{v_1, \dots, v_n\}$ and $E = \{e_1, \dots, e_m\}$. For each i $(1 \le i \le n)$, suppose that v_i is incident to edges e_{j_1}, e_{j_2} and e_{j_3} in G, where without loss of generality $j_1 < j_2 < j_3$. Define $e_{i,s} = e_{j_s}$ $(1 \le s \le 3)$. Similarly, for each j $(1 \le j \le m)$, suppose that $e_j = \{v_{i_1}, v_{i_2}\}$, where without loss of generality $i_1 < i_2$. Define $v_{j,r} = v_{i_r}$ $(1 \le r \le 2)$.

We form an instance I of $(\infty, 1, \infty)$-HRC from $\langle G, K\rangle$ as follows. The set of residents in I is $A \cup B \cup F \cup R \cup X \cup Y$ where $A = \{a_t : 1 \le t \le K\}$, $B = \{b_t : 1 \le t \le n - K\}$, $F = \cup_{t=1}^{K} F_t$, where $F_t = \{f_t^s : 1 \le s \le 6\}$, $R = \cup_{j=1}^{m} R_j$, where $R_j = \{r_j^s : 1 \le s \le 4\}$, $X = \{x_i : 1 \le i \le n\}$ and $Y = \cup_{t=1}^{n-K} Y_t$, where $Y_t = \{y_t^s : 1 \le s \le 6\}$.

Residents' Preferences

(r_j^1, r_j^2) :	(h_j^1, h_j^2)	$(1 \le j \le m)$
(r_j^3, r_j^4) :	(h_j^1, h_j^2)	$(1 \le j \le m)$
(f_t^1, f_t^2) :	(g_t^1, g_t^2)	$(1 \le t \le K)$
(f_t^3, f_t^4) :	(g_t^2, g_t^3)	$(1 \le t \le K)$
(f_t^5, f_t^6) :	(g_t^3, g_t^1)	$(1 \le t \le K)$
(y_t^1, y_t^2) :	(z_t^1, z_t^2)	$(1 \le t \le n-K)$
(y_t^3, y_t^4) :	(z_t^2, z_t^3)	$(1 \le t \le n-K)$
(y_t^5, y_t^6) :	(z_t^3, z_t^1)	$(1 \le t \le n-K)$
a_t :	$p_t \; g_t^1$	$(1 \le t \le K)$
b_t :	$q_t \; z_t^1$	$(1 \le t \le n-K)$
x_i :	$p_1 \; p_2 \; \dots \; p_K \; h^1(x_i) \; h^2(x_i) \; h^3(x_i) \; q_1 \; q_2 \; \dots \; q_{n-K}$	$(1 \le i \le n)$

Hospitals' Preferences

g_t^1 :	$a_t \; f_t^1 \; f_t^6$	$(1 \le t \le K)$
g_t^2 :	$f_t^3 \; f_t^2$	$(1 \le t \le K)$
g_t^3 :	$f_t^5 \; f_t^4$	$(1 \le t \le K)$
h_j^1 :	$r_j^1 \; x(h_j^1) \; r_j^3$	$(1 \le j \le m)$
h_j^2 :	$r_j^4 \; x(h_j^2) \; r_j^2$	$(1 \le j \le m)$
p_t :	$x_1 \; x_2 \; \dots \; x_n \; a_t$	$(1 \le t \le K)$
q_t :	$x_1 \; x_2 \; \dots \; x_n \; b_t$	$(1 \le t \le n-K)$
z_t^1 :	$b_t \; y_t^1 \; y_t^6$	$(1 \le t \le n-K)$
z_t^2 :	$y_t^3 \; y_t^2$	$(1 \le t \le n-K)$
z_t^3 :	$y_t^5 \; y_t^4$	$(1 \le t \le n-K)$

Figure 3.1: Preference lists in I, the constructed instance of $(\infty, 1, \infty)$-HRC.

The set of hospitals in I is $G \cup H \cup P \cup Q \cup Z$, where $G = \cup_{t=1}^{K} G_t$, where $G_t = \{g_t^r : 1 \leq r \leq 3\}$ $(1 \leq t \leq K)$, $H = \cup_{j=1}^{m} H_j$, where $H_j = \{h_j^s : 1 \leq s \leq 2\}$, $P = \{p_t : 1 \leq t \leq K\}$, $Q = \{q_t : 1 \leq t \leq n - K\}$ and $Z = \cup_{t=1}^{n-K} Z_t$, where $Z_t = \{z_t^r : 1 \leq r \leq 3\}$ and each hospital has capacity one. The preference lists of the resident couples, single residents and hospitals in I are shown in Figure 3.1.

In the preference list of a resident x_i $(1 \leq i \leq n)$ the symbol $h^s(x_i)$ $(1 \leq s \leq 3)$ denotes the hospital h_j^r $(1 \leq r \leq 2)$ such that $e_j = e_{i,s}$ and $v_i = v_{j,r}$. Similarly, in the preference list of a hospital h_j^r $(1 \leq j \leq m, 1 \leq r \leq 2)$ the symbol $x(h_j^r)$ denotes the resident x_i such that $v_i = v_{j,r}$.

We claim that G contains a vertex cover of size at most K if and only if I admits a stable matching. Let C be a vertex cover in G such that $|C| \leq K$. Without loss of generality we may assume that $|C| = K$ for if otherwise a sufficient number of vertices can be added to C without violating the vertex cover condition.

We show how to define a matching M in I from C as follows. Let $C = \{v_{r_1}, v_{r_2}, \ldots, v_{r_K}\}$ where without loss of generality $r_1 < r_2 < \ldots < r_K$. Further let $V \setminus C = \{v_{s_1}, v_{s_2}, \ldots, v_{s_{n-K}}\}$ where without loss of generality $s_1 < s_2 < \ldots < s_{n-k}$. For each vertex $v_{r_i} \in C$ add the pairs $\{(x_{r_i}, p_i), (a_i, g_i^1), (f_i^3, g_i^2), (f_i^4, g_i^3)\}$ for $1 \leq i \leq K$ to M. For each vertex $v_{s_i} \in V \setminus C$ add $\{(x_{s_i}, q_i), (b_i, z_i^1), (y_i^3, z_i^2), (y_i^4, z_i^3)\}$ for $1 \leq i \leq n - K$ to M. For each edge $e_j \in E$ at least one of $v_{j,1}$ or $v_{j,2}$ must be in C. If $v_{j,1} \in C$ add the pairs $\{(r_j^3, h_j^1), (r_j^4, h_j^2)\}$ to M. Otherwise $v_{j,2} \in C$ so add the pairs $\{(r_j^1, h_j^1), (r_j^2, h_j^2)\}$ to M.

It remains to show that M is a stable matching in I. Firstly, we show that no hospital $h_j^r \in H$ $(1 \leq j \leq m, 1 \leq r \leq 2)$ can form part of a blocking pair of M. Assume a hospital $h_j^r \in H$ is part of a blocking pair of M for some j $(1 \leq j \leq m)$ and r $(1 \leq r \leq 2)$. Now, since C is a vertex cover in G, an arbitrary edge $e_j \in E$ must be covered by either $v_{j,1}$ or $v_{j,2}$ or both. Assume firstly that $v_{j,1} \in C$. Then by construction $(x_{r_t}, p_t) \in M$ and $\{(r_j^3, h_j^1), (r_j^4, h_j^2)\} \subseteq M$ where $v_{j,1} = v_{r_t}$. Assume $(x(h_j^1), h_j^1)$ blocks M for some j $(1 \leq j \leq m)$. Since $v_{j,1} \in C$ and thus $M(x(h_j^1)) \in P$, $x(h_j^1)$ prefers $M(x(h_j^1))$ to h_j^1, a contradiction. Now assume that $((r_j^1, r_j^2), (h_j^1, h_j^2))$ blocks M. However, now h_j^2 prefers $M(h_j^2) = r_j^4$ to r_j^2, a contradiction.

Now assume $v_{j,1} \notin C$. Then $v_{j,2} \in C$ and by construction $(x_{r_{t'}}, p_{t'}) \in M$ and $\{(r_j^1, h_j^1), (r_j^2, h_j^2)\} \subseteq M$ where $v_{j,2} = v_{r_{t'}}$. Assume $(x(h_j^2), h_j^2)$ blocks M for some j $(1 \leq j \leq m)$. Since $v_{j,2} \in C$ and thus $M(x(h_j^2)) \in P$, $x(h_j^2)$ prefers $M(x(h_j^2))$ to h_j^2, a contradiction. Now assume $((r_j^3, r_j^4), (h_j^1, h_j^2))$ blocks M. However, h_j^1 prefers $M(h_j^1) = r_j^1$ to r_j^3, a contradiction. Thus, no $h_j^r \in H$ $(1 \leq j \leq m, 1 \leq r \leq 2)$ can form part of a blocking pair of M.

We now show that no hospital in $P \cup Q$ can be involved in a blocking pair of M. By construction $M(p_t) \in X$ for all t $(1 \leq t \leq K)$. Assume some pair (x_{k_1}, p_{l_1}) blocks M. Let $M(x_{k_1}) = p_{l_2}$ and $M(p_{l_1}) = x_{k_2}$. Since (x_{k_1}, p_{l_1}) blocks M then $l_1 < 1_2$ and $k_1 < k_2$ in contradiction to the construction of M. A similar argument shows that no hospital in Q may be involved in a blocking pair of M, and thus it follows that no hospital in $P \cup Q$ may be involved in a blocking pair of M.

We now show that no hospital in $G \cup Z$ can be involved in a blocking pair of M. Assume a hospital $g_t^s \in H$ is part of a blocking pair of M for some t $(1 \leq t \leq K)$ and s $(1 \leq s \leq 3)$. Now, since g_t^1 and g_t^2 are both assigned their first preference they cannot form part of a blocking pair for M. Now, assume g_t^3 prefers f_t^5 to $M(g_t^3) = f_t^4$. However, f_t^5 is a member of the couple (f_t^5, f_t^6) that expresses a joint preference for the pair (g_t^3, g_t^1) and g_t^1 prefers $M(g_t^1) = a_t$ to f_t^6, a contradiction. Thus, no hospital $g_t^s \in H$ $(1 \leq t \leq K, 1 \leq s \leq 3)$ can form part of a blocking pair of M. A similar argument may be used to show that no $z_t^s \in H$ $(1 \leq t \leq n - K, 1 \leq s \leq 3)$ can form part of a blocking pair of M and it follows that no hospital in $G \cup Z$ can be involved in a blocking pair of M. Thus no hospital in I may be part of a blocking pair of M and hence M must be a stable matching in I.

Conversely, let M be a stable matching in I. We first show that the stability of M implies that $M(x_i) \in P \cup Q$ for all i $(1 \leq i \leq n)$. Observe that if $(a_t, g_t^1) \notin M$ for t $(1 \leq t \leq K)$, then no stable assignment is possible amongst the agents in $F_t \cup G_t$ as will be established by Lemma 3.7.4 in Section 3.7.2. However, if $\{(a_t, g_t^1), (f_t^3, g_t^2), (f_t^4, g_t^3)\} \subseteq M$, then no blocking pair exists in $F_t \cup G_t$. It follows that if $(a_t, g_t^1) \in M$, then (a_t, p_t) blocks M unless $M(p_t) \in X$. A similar argument shows that $M(q_t) \in X$ for all t $(1 \leq t \leq n - K)$. Now, since $|X| = n$ and $|P \cup Q| = n$, clearly all $x \in X$ must be assigned a member of $P \cup Q$ and moreover, $M(x_i) \notin H$ in any stable matching in I.

Now we prove that the stability of M implies that h_j^1 and h_j^2 are fully subscribed in M for all j $(1 \leq j \leq m)$. Clearly since the only preferences expressed for h_j^1 and h_j^2 are as part of the hospital pair (h_j^1, h_j^2), either both h_j^1 and h_j^2 are fully subscribed in M or they both have no assignees. Let j $(1 \leq j \leq m)$ be given. Assume that both h_j^1 and h_j^2 are undersubscribed in M. Since $M(x(h_j^r)) \neq h_j^r$ for all j $(1 \leq j \leq m)$ and r $(1 \leq r \leq 2)$, it follows that $((r_j^1, r_j^2), (h_j^1, h_j^2))$ blocks M, a contradiction. Thus either $\{(r_j^1, h_j^1), (r_j^2, h_j^2)\} \subseteq M$ or $\{(r_j^3, h_j^1), (r_j^4, h_j^2)\} \subseteq M$ in any stable matching in I. If $\{(r_j^1, h_j^1), (r_j^2, h_j^2)\} \subseteq M$, then $((r_j^3, r_j^4), (h_j^1, h_j^2))$ does not block M. Similarly, if $\{(r_j^3, h_j^1), (r_j^4, h_j^2)\} \subseteq M$, then $((r_j^1, r_j^2), (h_j^1, h_j^2))$ does not block M. Thus we have that h_j^1 and h_j^2 are fully subscribed in M for all j $(1 \leq j \leq m)$. Moreover, we have that all hospitals must be fully subscribed in any stable matching M in I.

Define a set of vertices C in G as follows. For each i $(1 \leq i \leq n)$ if $M(x_i) \in P$, add v_i to C. Since M is a stable matching and $|P| = K$, this process selects exactly K of the n vertices in V and thus $|C| = K$. We now show that C represents a vertex cover in G. Consider an arbitrary edge $e_j \in E$. Assume that both $v_{j,1} \notin C$ and $v_{j,2} \notin C$ and hence that C is not a vertex cover in G. Then $M(x_{j,1}) \in Q$ and $M(x_{j,2}) \in Q$. As M is stable and thus hospital complete, either $\{(r_j^1, h_j^1), (r_j^2, h_j^2)\} \subset M$ or $\{(r_j^3, h_j^1), (r_j^4, h_j^2)\} \subset M$. If $\{(r_j^1, h_j^1), (r_j^2, h_j^2)\} \subset M$, then $(x_{j,2}, h_j^2)$ blocks M, a contradiction. If $\{(r_j^3, h_j^1), (r_j^4, h_j^2)\} \subset M$, then $(x_{j,1}, h_j^1)$ blocks M, a contradiction. Hence C represents a vertex cover in G of size K and the theorem is proven. □

3.3 Complexity results for $(2, 2)$-HRC

3.3.1 NP-completeness result for $(2, 2)$-HRC

We now establish that the problem of deciding whether an instance of $(2, 2)$-HRC admits a stable matching is NP-complete.

Theorem 3.3.1. *Given an instance of $(2, 2)$-HRC, the problem of deciding whether the instance admits a stable matching is NP-complete. The result holds even if there are no single residents and each hospital has capacity one.*

Proof. The proof of this result uses a reduction from a restricted version of SAT. More specifically, let (2,2)-E3-SAT denote the problem of deciding, given a Boolean formula B in CNF over a set of variables V, whether B is satisfiable, where B has the following properties: (i) each clause contains exactly three literals and (ii) for each $v_i \in V$, each of literals v_i and $\bar{v}_i$ appears exactly twice in B. Berman et al. [9] showed that (2,2)-E3-SAT is NP-complete.

The problem $(2, 2)$-HRC is clearly in NP, as a given assignment may be verified to be a stable matching in polynomial time. To show NP-hardness we now present a polynomial time reduction from an instance of (2,2)-E3-SAT to an instance of $(2, 2)$-HRC. Let B be an instance of (2,2)-E3-SAT. Let $V = \{v_1, v_2, \ldots, v_n\}$ and $C = \{c_1, c_2, \ldots, c_m\}$ be the set of variables and clauses respectively in B. Then for each $v_i \in V$, each of literals v_i and $\bar{v}_i$ appears exactly twice in B. Also $|c_j|= 3$ for each $c_j \in C$. Hence $m = \frac{4n}{3}$. We form an instance I of $(2, 2)$-HRC from an instance of (2,2)-E3-SAT as follows.

The set of residents in I is $A \cup B \cup X \cup Y$ where $A = \cup_{i=1}^n A_i$, $A_i = \{a_i^r : 1 \leq r \leq 2\}$ $(1 \leq i \leq n)$, $B = \cup_{i=1}^n B_i$, $B_i = \{b_i^r : 1 \leq r \leq 2\}$ $(1 \leq i \leq n)$, $X = \cup_{j=1}^m X_j$, $X_j = \{x_j^s : 1 \leq s \leq 3\}$ $(1 \leq j \leq m)$ and $Y = \cup_{j=1}^m Y_j$, $Y_j = \{y_j^s : 1 \leq s \leq 3\}$ $(1 \leq j \leq m)$. There are no single residents in I and the pairing of the residents into couples is as shown in Figure 3.2.

The set of hospitals in I is $H \cup T$, where $H = \cup_{i=1}^n H_i$, $H_i = \{h_i^r : 1 \leq r \leq 6\}$ $(1 \leq i \leq n)$ and $T = \cup_{j=1}^m T_j$, $T_j = \{t_j^r : 1 \leq r \leq 6\}$ $(1 \leq j \leq m)$ and each hospital has capacity one. The preference lists of the resident couples and hospitals in I are shown in Figure 3.2.

In the joint preference list of a couple (x_j^s, y_j^s) $(1 \leq j \leq m, 1 \leq s \leq 3)$ the symbol $h(x_j^s)$ is defined as follows. If the r^{th} occurrence $(1 \leq r \leq 2)$ of literal v_i occurs at position s of c_j

Residents' Preferences			
(a_i^1, b_i^1) :	(h_i^1, h_i^3)	(h_i^2, h_i^4)	$(1 \le i \le n)$
(a_i^2, b_i^2) :	(h_i^2, h_i^5)	(h_i^1, h_i^6)	$(1 \le i \le n)$
(x_j^1, y_j^1) :	$(h(x_j^1), t_j^4)$	(t_j^1, t_j^3)	$(1 \le j \le m)$
(x_j^2, y_j^2) :	$(h(x_j^2), t_j^5)$	(t_j^2, t_j^1)	$(1 \le j \le m)$
(x_j^3, y_j^3) :	$(h(x_j^3), t_j^6)$	(t_j^3, t_j^2)	$(1 \le j \le m)$

Hospitals' Preferences			
h_i^1 :	a_i^2	a_i^1	$(1 \le i \le n)$
h_i^2 :	a_i^1	a_i^2	$(1 \le i \le n)$
h_i^3 :	b_i^1	$x(h_i^3)$	$(1 \le i \le n)$
h_i^4 :	b_i^1	$x(h_i^4)$	$(1 \le i \le n)$
h_i^5 :	b_i^2	$x(h_i^5)$	$(1 \le i \le n)$
h_i^6 :	b_i^2	$x(h_i^6)$	$(1 \le i \le n)$
t_j^1 :	x_j^1	y_j^2	$(1 \le j \le m)$
t_j^2 :	x_j^2	y_j^3	$(1 \le j \le m)$
t_j^3 :	x_j^3	y_j^1	$(1 \le j \le m)$
t_j^4 :	y_j^1		$(1 \le j \le m)$
t_j^5 :	y_j^2		$(1 \le j \le m)$
t_j^6 :	y_j^3		$(1 \le j \le m)$

Figure 3.2: Preference lists in I, the constructed instance of $(2,2)$-HRC.

then $h(x_j^s) = h_i^{2r+1}$. If the r^{th} occurrence $(1 \le r \le 2)$ of literal $\bar{v}_i$ occurs at position s of c_j then $h(x_j^s) = h_i^{2r+2}$.

In the preference list of a hospital h_i^{2r+1} $(1 \le r \le 2)$, the symbol $x(h_i^{2r+1})$ denotes the resident x_j^s such that the r^{th} occurrence of literal v_i occurs at position s of clause c_j. Similarly in the preference list of a hospital h_i^{2r+2} $(1 \le r \le 2)$, the symbol $x(h_i^{2r+2})$ denotes the resident x_j^s such that the r^{th} occurrence of literal $\bar{v}_i$ occurs at position s of clause c_j.

For each i $(1 \le i \le n)$, let

$$T_i = \{(a_i^1, h_i^2), (a_i^2, h_i^1), (b_i^1, h_i^4), (b_i^2, h_i^6), (x(h_i^3), h_i^3), (x(h_i^5), h_i^5)\}$$

and

$$F_i = \{(a_i^1, h_i^1), (a_i^2, h_i^2), (b_i^1, h_i^3), (b_i^2, h_i^5), (x(h_i^4), h_i^4), (x(h_i^6), h_i^6)\}.$$

This completes the reduction.

We claim that B is satisfiable if and only if I admits a stable matching. First, let f be a satisfying truth assignment of B. From f we construct a matching M in I as follows. For each variable $v_i \in V$, if v_i is true under f, add the pairs in T_i to M, otherwise add the pairs in F_i to M. Let j $(1 \leq j \leq m)$ be given. Then c_j contains at least one literal that is true under f. Suppose c_j contains exactly one literal that is true under f. Let s be the position of c_j containing a true literal. In this case add the pairs $\{(x_j^{s+1}, t_j^{s+1}), (y_j^{s+1}, t_j^s)\}$ (where addition is taken modulo three) to M. Now suppose c_j contains exactly two literals that are true under f. Let s be the position of c_j containing a false literal, and add the pairs $\{(x_j^s, t_j^s), (y_j^s, t_j^{s+2})\}$ (where addition is taken modulo three) to M. If c_j contains three literals that are true under f no additional pairs need be added.

It remains to prove that M is stable in I. We prove this is the case by considering the resident pairs in turn and showing that no resident pair may be involved in a blocking pair in M and hence M must be stable. Firstly, no resident pair (a_i^1, b_i^1) or (a_i^2, b_i^2) may be involved in a blocking pair of M, as no matching in which (a_i^1, b_i^1) is assigned to (h_i^2, h_i^4) is blocked by (a_i^1, b_i^1) with (h_i^1, h_i^3), and similarly no matching in which (a_i^2, b_i^2) is assigned to (h_i^1, h_i^6) is blocked by (a_i^2, b_i^2) with (h_i^2, h_i^5).

Secondly, no resident pair (x_j^s, y_j^s) $(1 \leq s \leq 3)$ may block M with $(h(x_j^s), t_j^{s+3})$ (where addition is taken modulo three). To prove this observe that all h_i^r are assigned in M and hence if some h_i^r is not assigned to its corresponding $x(h_i^r)$, then h_i^r must be assigned to the member of B_i in first place on its preference list. Thus (x_j^s, y_j^s) may not block M with $(h(x_j^s), t_j^{s+3})$.

Finally, no resident pair (x_j^s, y_j^s) $(1 \leq s \leq 3)$ may block M with (t_j^s, t_j^{s+2}) (where addition is taken modulo three). Clearly (x_j^s, y_j^s) could only block M with (t_j^s, t_j^{s+2}) if (x_j^s, y_j^s) is unassigned in M. From the construction, this may only be the case if c_j contains exactly one literal that is true under f. In this case, (x_j^{s+2}, y_j^{s+2}) is assigned to (t_j^{s+2}, t_j^{s+4}) (where addition is taken modulo three) and thus (x_j^s, y_j^s) $(1 \leq s \leq 3)$ does not block M with (t_j^s, t_j^{s+2}), since t_j^{s+2} prefers x_j^{s+2} to y_j^s. Hence M is a stable matching in I.

Conversely, suppose that M is a stable matching in I. We form a truth assignment f in B from M as follows. For any i $(1 \leq i \leq n)$, if (a_i^1, b_i^1) is unassigned then M is blocked by (a_i^1, b_i^1) with (h_i^2, h_i^4). Similarly, if (a_i^2, b_i^2) is unassigned then M is blocked by (a_i^2, b_i^2) with (h_i^1, h_i^6). Hence either $\{(a_i^1, h_i^2), (b_i^1, h_i^4), (a_i^2, h_i^1), (b_i^2, h_i^6)\} \subseteq M$ or $\{(a_i^1, h_i^1), (b_i^1, h_i^3), (a_i^2, h_i^2), (b_i^2, h_i^5)\} \subseteq M$.

Now, let c_j be a clause in C $(1 \leq j \leq m)$. Suppose, $(x_j^s, h(x_j^s)) \notin M$ for all s $(1 \leq s \leq 3)$.

Clearly, at most one couple (x_j^s, y_j^s) may be assigned to the hospital pair in second place on its preference list. Since no (x_j^s, y_j^s) is assigned to the pair in first place in its preference list one of the remaining two unassigned (x_j^s, y_j^s)'s must block with the hospital pair in second place on its preference list, a contradiction. Thus $\{(x_j^s, h(x_j^s)), (y_j^s, t_j^{s+3}))\} \subseteq M$ for some s $(1 \leq s \leq 3)$ by the stability of M

Hence, for each j $(1 \leq j \leq m)$, let s $(1 \leq s \leq 3)$ be given such that (x_j^s, y_j^s) is assigned to $(h(x_j^s), t_j^{s+3})$. Let $h_i^r = h(x_j^s)$. If $r \in \{3, 5\}$ then we set $f(v_i) = T$. Thus, variable v_i is true under f and hence clause c_j is true under f since the literal v_i occurs in c_j. Otherwise, $r \in \{4, 6\}$ and we set $f(v_i) = F$. Thus, variable v_i is false under f and hence clause c_j is true under f since the literal $\bar{v}_i$ occurs in c_j.

This assignment of truth values is well-defined, for if $(h_i^r, t_j^{s+3}) \in M$ for $r \in \{3, 5\}$, then $\{(b_i^1, h_i^4), (b_i^2, h_i^6)\} \subseteq M$, so neither h_i^4 nor h_i^6 is assigned a member of X in M. Similarly if $(h_i^r, t_j^{s+3}) \in M$ for $r \in \{4, 6\}$, then $\{(b_i^1, h_i^3), (b_i^2, h_i^5)\} \subseteq M$, so neither h_i^3 nor h_i^5 is assigned a member of X in M. Hence f is a satisfying truth assignment of B. Thus, we have that that B is satisfiable if and only if I admits a stable matching and the result is proven. □

3.3.2 Inapproximability of (2, 2)-MIN BP HRC

In Theorem 3.3.1 we showed that the problem of deciding whether an instance of (2, 2)-HRC admits a stable matching is NP-complete. In Theorem 3.3.2 we prove that the minimisation problem of finding a 'most stable' matching in an instance of (2, 2)-HRC, denoted by (2, 2)-MIN BP HRC, is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP.

Before beginning the statement of the theorem we first define an instance S of HRC with the property that any non-empty matching in S admits exactly one blocking pair. This instance is used as an element in the proof of Theorem 3.3.2 to ensure that the large instance of HRC constructed in the proof can admit no stable matching.

Let S be an instance of (2, 2)-HRC as shown in Figure 3.3. In S the residents are a, b and c, the hospitals are z_1 and z_2 and each hospital has capacity one. The instance S admits three non-empty matchings, namely

$$M_1 = \{(a, z_1), (b, z_2)\}$$

$$M_2 = \{(c, z_1)\}$$

$$M_3 = \{(c, z_2)\}$$

Clearly, none of the matchings are stable. Further each of the non-empty matchings in S is blocked by exactly one blocking pair. Resident c blocks M_1 in A with z_2, couple (a, b)

Residents' Preferences

$(a, b): \quad (z_1, z_2)$
$c: \quad z_1 \;\; z_2$

Hospitals' Preferences

$z_1: \quad a \;\; c$
$z_2: \quad c \;\; b$

Figure 3.3: A small instance of $(2,2)$-HRC that admits no stable matching.

blocks M_2 in S with (z_1, z_2) and resident c blocks M_3 in S with z_1. Thus, S admits no stable matching and taken over all of the non-empty matchings admitted by S, the only number of blocking pairs possible (and thus the minimum) is one. We now present a proof that $(2,2)$-MIN BP HRC is not approximable to within a tight bound, unless P=NP.

Theorem 3.3.2. *The problem $(2,2)$-MIN BP HRC is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. The result holds even if there are no single residents and each hospital has capacity one.*

Proof. Let B be an instance of (2,2)-E3-SAT and let I be the corresponding instance of $(2,2)$-HRC as constructed in Theorem 3.3.1. We now show how to modify I in order to obtain an extended instance I'' of $(2,2)$-HRC as follows.

Assume $\varepsilon > 0$. Choose $c = \lceil 2/\varepsilon \rceil$ and $k = n^c$. Now, let $I_1, I_2, \ldots, I_k$ be k disjoint copies of the instance I. Let I' be the $(2,2)$-HRC instance formed by taking the union of the subinstances $I_1, I_2, \ldots, I_k$. Let I'' be the instance constructed by taking the union of I' with the instance S of $(2,2)$-HRC shown in Figure 3.3.

If B admits a satisfying truth assignment, then by Theorem 3.3.1, I admits a stable matching and clearly each copy of I must also admit a stable matching. Thus I' must admit a stable matching. Moreover, since any non-empty matching admitted by S admits exactly one blocking pair, a matching exists in I'' that admits exactly one blocking pair.

If B admits no satisfying truth assignment, then by Theorem 3.3.1, I admits no stable matching. Now, any matching admitted by I'' must be blocked by at least $k+1$ blocking pairs – we demonstrate this as follows. Clearly, since I admits no stable matching, any matching in I must admit at least one blocking pair. Thus any matching in each I_r $(1 \leq r \leq k)$ admits one or more blocking pairs. Now, since the only non-empty matchings admitted by S admit a single blocking pair, any matching admitted by I'' must have at least $k+1$ blocking pairs.

The number of residents in I'' is $n_1 = 4nk + 6mk + 3$. From the construction of I in Theorem

3.3.1 we know that $4n = 3m$ and thus $n_1 \leq 12nk + 3$. We lose no generality by assuming that $n \geq 3$. Thus $n_1 \leq 13nk = 13n^{c+1}$.

Moreover,

$$13^{-c/(c+1)} n_1^{c/(c+1)} \leq k. \tag{3.1}$$

Now we know that $n_1 \geq k = n^c$. We lose no generality by assuming that $n \geq 13$ and hence $n_1 \geq 13^c$. It follows that

$$n_1^{-1/(c+1)} \leq 13^{-c/(c+1)}. \tag{3.2}$$

Thus it follows from Inequality 3.1 and 3.2 that

$$n_1^{(c-1)/(c+1)} = n_1^{c/(c+1)} n_1^{-1/(c+1)} \leq 13^{-c/(c+1)} n_1^{c/(c+1)} \leq k \tag{3.3}$$

We now show that $n_1^{1-\varepsilon} \leq n_1^{(c-1)/(c+1)}$. Observe that $c \geq 2/\varepsilon$ and thus $c+1 \geq 2/\varepsilon$. Hence

$$1 - \varepsilon \leq \frac{c+1-2}{c+1} = \frac{c-1}{c+1}$$

and hence by Inequality 3.3, $n_1^{1-\varepsilon} \leq k$.

Assume that A is an approximation algorithm for $(2,2)$-HRC with a performance guarantee of $n_1^{1-\varepsilon} \leq k$. Let B be an instance of (2,2)-E3-SAT and construct an instance I'' of $(2,2)$-HRC from B as described above. If B admits a satisfying truth assignment, then A must return a matching in I'' that admits at most k blocking pairs. Otherwise, B does not admit a satisfying assignment and A must return a matching admitting at least $k+1$ blocking pairs. Thus algorithm A may be used to determine whether B admits a satisfying truth assignment in polynomial time, a contradiction. Hence, no such polynomial approximation algorithm can exist, unless P=NP. □

3.4 Complexity results for $(2,3)$-HRC

We now establish that the problem of deciding whether an instance of $(2,3)$-HRC admits a stable matching is NP-complete.

Lemma 3.4.1. *Given an instance of $(2,3)$-HRC, the problem of deciding whether the instance admits a stable matching is NP-complete. The result holds even if there are no single residents and each hospital has capacity one.*

Proof. The proof of this result uses a reduction from a restricted version of SAT. More specifically, let (2,2)-E3-SAT denote the problem of deciding, given a Boolean formula B in CNF over a set of variables V, whether B is satisfiable, where B has the following properties: (i) each clause contains exactly three literals and (ii) for each $v_i \in V$, each of literals v_i and $\bar{v}_i$ appears exactly twice in B. Berman et al. [9] showed that (2,2)-E3-SAT is NP-complete.

The problem $(2, 3)$-HRC is clearly in NP, as a given assignment may be verified to be a stable matching in polynomial time. To show NP-hardness, we now present a polynomial time reduction from an instance of (2,2)-E3-SAT to an instance of $(2, 3)$-HRC. Let B be an instance of (2,2)-E3-SAT. Let $V = \{v_1, v_2, \ldots, v_n\}$ and $C = \{c_1, c_2, \ldots, c_m\}$ be the set of variables and clauses respectively in B. Then for each $v_i \in V$, each of literals v_i and $\bar{v}_i$ appears exactly twice in B. Also $|c_j|= 3$ for each $c_j \in C$. Hence $m = \frac{4n}{3}$. We form an instance I of $(2, 3)$-HRC from an instance of (2,2)-E3-SAT as follows.

The set of residents in I is $X \cup P \cup Q$, where $X = \cup_{i=1}^{n} X_i$, $X_i = \{x_i^r, \bar{x}_i^r : 1 \leq r \leq 2\}$ $(1 \leq i \leq n)$, $P = \cup_{j=1}^{m} P_j$, $P_j = \{p_j^s : 1 \leq s \leq 3\}$ $(1 \leq j \leq m)$, $Q = \cup_{j=1}^{m} Q_j$, $Q_j = \{q_j^s : 1 \leq s \leq 3\}$ $(1 \leq j \leq m)$. There are no single residents in I and the pairing of the residents into couples is as shown in Figure 3.4. The set of hospitals in I is $H \cup Y$, where $H = \cup_{i=1}^{n} H_i$, $H_i = \{h_i^r : 1 \leq r \leq 2\}$ $(1 \leq i \leq n)$ and $Y = \cup Y_j$, $Y_j = \{y_j^s : 1 \leq s \leq 3\}$ $(1 \leq j \leq m)$ and each hospital has capacity one. The preference lists of the resident couples and hospitals in I are shown in Figure 3.4.

In the joint preference list of a couple (x_i^1, x_i^2) $(x_i^1 \in X, x_i^2 \in X)$ the symbol $y(x_i^r)$ $(1 \leq r \leq 2)$ denotes the hospital $y_j^s \in Y$ such that the rth occurrence of literal v_i appears at position s of clause c_j in B. Similarly in the joint preference list of a couple $(\bar{x}_i^1, \bar{x}_i^2)$ $(\bar{x}_i^1 \in X, \bar{x}_i^2 \in X)$ the symbol $y(\bar{x}_i^r)$ $(1 \leq r \leq 2)$ denotes the hospital $y_j^s \in Y$ such that the rth occurrence of literal $\bar{v}_i$ appears at position s of clause c_j in B. In the preference list of a hospital $y_j^s \in Y$, if literal v_i (respectively $\bar{v}_i$) appears at position s of clause $c_j \in C$, the symbol $x(y_j^s)$ denotes the resident x_i^1 or x_i^2 (respectively $\bar{x}_i^1$ or $\bar{x}_i^2$) according as this is the first or second occurrence of the literal in B. For each i $(1 \leq i \leq n)$, let $T_i = \{(x_i^1, y(x_i^1)), (x_i^2, y(x_i^2)), (\bar{x}_i^1, h_i^1), (\bar{x}_i^2, h_i^2)\}$ and let $F_i = \{(x_i^1, h_i^1), (x_i^2, h_i^2), (\bar{x}_i^1, y(\bar{x}_i^1)), (\bar{x}_i^2, y(\bar{x}_i^2))\}$. This completes the reduction.

We claim that B is satisfiable if and only if I admits a stable matching. First, let f be a satisfying truth assignment of B. From f we construct a matching M in I as follows. For each variable $v_i \in V$, if v_i is true under f, add the pairs in T_i to M, otherwise add the pairs in F_i to M. Let j $(1 \leq j \leq m)$ be given. Then c_j contains at least one literal that is true under f. Suppose this literal occurs at position s of c_j $(1 \leq s \leq 3)$, then $(x(y_j^s), y_j^s) \in M$. If no other literal in c_j is true, then add the pairs $\{(p_j^{s+1}, y_j^{s+1}), (q_j^{s+1}, y_j^{s+2})\}$ to M (where addition is taken modulo three).

It remains to show that M is stable. We prove this is the case by considering the resident pairs in turn and showing that no resident pair may be involved in a blocking pair in M and

Residents' Preferences

$(x_i^1, x_i^2):$	(h_i^1, h_i^2)	$(y(x_i^1), y(x_i^2))$	$(1 \le i \le n)$
$(\bar{x}_i^1, \bar{x}_i^2):$	(h_i^1, h_i^2)	$(y(\bar{x}_i^1), y(\bar{x}_i^2))$	$(1 \le i \le n)$
$(p_j^1, q_j^1):$	(y_j^1, y_j^2)		$(1 \le j \le m)$
$(p_j^2, q_j^2):$	(y_j^2, y_j^3)		$(1 \le j \le m)$
$(p_j^3, q_j^3):$	(y_j^3, y_j^1)		$(1 \le j \le m)$

Hospitals' Preferences

$h_i^1:$	$x_i^1 \ \bar{x}_i^1$	$(1 \le i \le n)$
$h_i^2:$	$\bar{x}_i^2 \ x_i^2$	$(1 \le i \le n)$
$y_j^1:$	$x(y_j^1) \ p_j^1 \ q_j^3$	$(1 \le j \le m)$
$y_j^2:$	$x(y_j^2) \ p_j^2 \ q_j^1$	$(1 \le j \le m)$
$y_j^3:$	$x(y_j^3) \ p_j^3 \ q_j^2$	$(1 \le j \le m)$

Figure 3.4: Preference lists in I, the constructed instance of $(2,3)$-HRC.

hence the M must be stable.

No resident pair (x_i^1, x_i^2) or $(\bar{x}_i^1, \bar{x}_i^2)$ may block M, as no matching in which (x_i^1, x_i^2) is assigned to (h_i^1, h_i^2) is blocked by $(\bar{x}_i^1, \bar{x}_i^2)$ with (h_i^1, h_i^2), and similarly no matching in which $(\bar{x}_i^1, \bar{x}_i^2)$ is assigned to (h_i^1, h_i^2) is blocked by (x_i^1, x_i^2) with (h_i^1, h_i^2). No resident pair (p_j^s, q_j^s) may block M as, if (p_j^s, q_j^s) is not assigned to (y_j^s, y_j^{s+1}) (where addition is taken modulo three), then at least one of y_j^s or y_j^{s+1} is assigned to its first choice and thus (p_j^s, q_j^s) may not block M with (y_j^s, y_j^{s+1}). Hence M is a stable matching in I.

Conversely suppose that M is a stable matching in I. We form a truth assignment f in B from M as follows. For any i $(1 \le i \le n)$, if h_i^1 and h_i^2 are unassigned, then M is blocked by (x_i^1, x_i^2) with (h_i^1, h_i^2). Thus, either $\{(x_i^1, h_i^1), (x_i^2, h_i^2)\} \subseteq M$ or $\{(\bar{x}_i^1, h_i^1), (\bar{x}_i^2, h_i^2)\} \subseteq M$. Now, suppose (x_i^1, x_i^2) are unassigned in M. Then (x_i^1, x_i^2) blocks M with $(y(x_i^1), y(x_i^2))$. Similarly $(\bar{x}_i^1, \bar{x}_i^2)$ must be assigned in M. Thus, $M \cap (X_i \times (H \cup Y)) = T_i$ or $M \cap (X_i \times (H \cup Y)) = F_i$. In the former case set $f(x_i) = T_i$ and in the latter case set $f(x_i) = F_i$.

Now let c_j be a clause in C $(1 \le j \le m)$. Suppose $(x(y_j^s), y_j^s) \notin M$ for all s $(1 \le s \le 3)$. If (p_j^1, q_j^1) is assigned to (y_j^1, y_j^2), then (p_j^2, q_j^2) blocks M with (y_j^2, y_j^3). If (p_j^2, q_j^2) is assigned to (y_j^2, y_j^3), then (p_j^3, q_j^3) blocks M with (y_j^3, y_j^1). If (p_j^3, q_j^3) is assigned to (y_j^3, y_j^1), then (p_j^1, q_j^1) blocks M with (y_j^1, y_j^2). If $\{(p_j^s, y_j^s), (q^s, y_j^{s+1})\} \not\subseteq M$ for some s $(1 \le s \le 3)$ (where addition is taken modulo three), then (p_j^1, q_j^1) blocks M with (y_j^1, y_j^2). Thus $(x(y_j^s), y_j^s) \in M$ for some s $(1 \le s \le 3)$ by the stability of M.

If $x(y_j^s) = x_i^r$ then $(x_i^r, y_j^s) \in T_i$. Thus v_i is true under f and it follows that c_j is true under

$$L_i^1 : (h_i^1, h_i^2)\ (y(x_i^1), y(x_i^2))\ (y(\bar{x}_i^1), y(\bar{x}_i^2)) \qquad (1 \le i \le n)$$

$$L_i^2 : (y_j^1, y_j^2)\ (y_j^2, y_j^3)\ (y_j^3, y_j^1) \qquad (1 \le j \le m)$$

$$\text{Master List} : L_1^1\ L_2^1 \dots L_n^1\ L_1^2\ L_2^2 \dots L_m^2$$

Figure 3.5: Master list of preferences for resident couples in $(2,3)$-HRC instance I.

$$L_i^3 : x_i^1\ \bar{x}_i^1\ \bar{x}_i^2\ x_i^2 \qquad (1 \le i \le n)$$

$$L_j^4 : p_j^1\ p_j^2\ p_j^3 \qquad (1 \le j \le m)$$

$$L_i^5 : q_j^1\ q_j^2\ q_j^3 \qquad (1 \le j \le m)$$

$$\text{Master List} : L_1^3\ L_2^3 \dots L_n^3\ L_1^4\ L_2^4 \dots L_m^4\ L_1^5\ L_2^5 \dots L_m^5$$

Figure 3.6: Master list of preferences for hospitals in $(2,3)$-HRC instance I.

f. Otherwise $x(y_j^s) = \bar{x}_i^r$, so $(\bar{x}_i^r, y_j^s) \in F_i$ and $\bar{v}_i$ is false under f and it follows that c_j is true under f. Hence f is a satisfying truth assignment of B. Thus, we have that that B is satisfiable if and only if I admits a stable matching and the result is proven. □

Theorem 3.4.2. *Given an instance of $(2,3)$-HRC, the problem of deciding whether the instance admits a stable matching exists is NP-complete. The result holds even in the case where there are no single residents, each hospital has capacity one and the preference list of each couple and hospital is derived from a strictly ordered master list of pairs of hospitals and residents respectively.*

Proof. We now demonstrate that NP-completeness holds under other restrictions on the structure of the agents' preference list than those stated in Lemma 3.4.1. Consider the instance I of $(2,3)$-HRC as constructed in the proof of Lemma 3.4.1, and let n and m be defined as in the proof of Lemma 3.4.1. The master lists shown in Figures 3.5 and 3.6 indicate that the preference list of each resident couple and hospital may be derived from a master list of hospital pairs and residents respectively. Since there are no single residents in I, no preferences are expressed for individual hospitals in I.

As we have established in Lemma 3.4.1 that deciding whether an instance of $(2,3)$-HRC of this form admits a stable matching is NP-complete, the theorem is proven.

□

3.5 Complexity results for $(3,3)$-HRC DUAL MARKET

Lemma 3.5.1. *Given an instance of $(3,3)$-HRC, the problem of deciding whether the instance admits a complete stable matching is NP-complete. The result holds even if each hospital has capacity one.*

Proof. The problem is clearly in NP, as a given assignment may be verified to be a complete, stable matching in polynomial time. This proof uses a polynomial-time reduction from a restricted version of SAT. More specifically, let (2,2)-E3-SAT denote the problem of deciding, given a Boolean formula B in CNF over a set of variables V, whether B is satisfiable, where B has the following properties: (i) each clause contains exactly three literals and (ii) for each $v_i \in V$, each of literals v_i and $\bar{v}_i$ appears exactly twice in B. Berman et al. [9] showed that (2,2)-E3-SAT is NP-complete.

Let B be an instance of (2,2)-E3-SAT. We construct an instance I of $(3,3)$-HRC using a similar reduction to that employed by Irving et al. [41]. Let $V = \{v_0, v_1, \ldots, v_{n-1}\}$ and $C = \{c_1, c_2, \ldots, c_m\}$ be the set of variables and clauses respectively in B. Then for each $v_i \in V$, each of literals v_i and $\bar{v}_i$ appears exactly twice in B. Also $|c_j|= 3$ for each $c_j \in C$. (Hence $m = \frac{4n}{3}$.)

The set of residents in I is $X \cup K \cup P \cup Q \cup T$, where $X = \cup_{i=0}^{n-1} X_i$, $X_i = \{x_{4i+r} : 0 \leq r \leq 3\}$ $(0 \leq i \leq n-1)$, $K = \cup_{i=0}^{n-1} K_i$, $K_i = \{k_{4i+r} : 0 \leq r \leq 3\}$ $(0 \leq i \leq n-1)$, $P = \cup_{j=1}^{m} P_j$, $P_j = \{p_j^r : 1 \leq r \leq 6\}$ $(1 \leq j \leq m)$, $Q = \{q_j : c_j \in C\}$ and $T = \{t_j : c_j \in C\}$. The residents in $Q \cup T$ are single and the residents in $X \cup K \cup P$ are involved in couples.

The set of hospitals in I is $Y \cup L \cup C' \cup Z$, where $Y = \cup_{i=0}^{n-1} Y_i$, $Y_i = \{y_{4i+r} : 0 \leq r \leq 3\}$ $(0 \leq i \leq n-1)$, $L = \cup_{i=0}^{n-1} L_i$, $L_i = \{l_{4i+r} : 0 \leq r \leq 3\}$ $(0 \leq i \leq n-1)$, $C' = \{c_j^s : c_j \in C \wedge 1 \leq s \leq 3\}$ and $Z = \{z_j^r : 1 \leq j \leq m \wedge 1 \leq r \leq 5\}$. Each hospital has capacity one.

In the joint preference list of a couple $(x_{4i+r}, k_{4i+r}) \in X \times K$, if $r \in \{0, 1\}$, the symbol $c(x_{4i+r})$ denotes the hospital $c_j^s \in C'$ such that the $(r+1)$th occurrence of literal v_i appears at position s of c_j. Similarly if $r \in \{2, 3\}$, then the symbol $c(x_{4i+r})$ denotes the hospital $c_j^s \in C'$ such that the $(r-1)$th occurrence of literal $\bar{v}_i$ appears at position s of c_j. Also in the preference list of a hospital $c_j^s \in C'$, if literal v_i appears at position s of clause $c_j \in C$, the symbol $x(c_j^s)$ denotes the resident x_{4i+r-1} where $r = 1, 2$ according as this is the first or second occurrence of literal v_i in B. Otherwise if literal $\bar{v}_i$ appears at position s of clause $c_j \in C$, the symbol $x(c_j^s)$ denotes the resident x_{4i+r+1} where $r = 1, 2$ according as this is the first or second occurrence of literal $\bar{v}_i$ in B. The preference lists of the residents and hospitals in I are shown in Figure 3.7 and diagrammatically in Figures 3.8 and 3.9. In these diagrams any two residents joined by a dashed line represent a couple in I. The numbers on each outgoing edge from a single resident in the diagram represent the preference ordering over the single hospitals expressed by the resident. For a couple, the numbers on each outgoing edge from each member of the couple represent that preference order over the pairs of hospitals expressed by the couple. For example, for the couple (p_j^1, p_j^4), the first outgoing edge from p_j^1 is z_j^1 and the first outgoing edge from p_j^4 is z_j^2. Thus, the first joint preference of the couple (p_j^1, p_j^4) is (z_j^1, z_j^2). Clearly each preference list is of length at most three.

For each i $(0 \leq i \leq n-1)$, let $T_i = \{(x_{4i+r}, y_{4i+r}) : 0 \leq r \leq 3\} \cup \{(k_{4i+r}, l_{4i+r}) : 0 \leq r \leq 3\}$ and $F_i = \{(x_{4i+r}, y_{4i+r+1})\} : 0 \leq r \leq 3\} \cup \{(k_{4i+r}, l_{4i+r+1})\} : 0 \leq r \leq 3\}$, where addition is taken modulo four.

We claim that B is satisfiable if and only if I admits a complete stable matching. First, suppose that B is satisfiable and let f be a satisfying truth assignment of B. Define a complete matching M in I as follows. For each variable $v_i \in V$, if v_i is true under f, add the pairs in T_i to M, otherwise add the pairs in F_i to M. Let j $(1 \leq j \leq m)$ be given. Now, since f is a satisfying truth assignment of B, c_j contains at least one literal that is true under f. Suppose this literal occurs at position s of c_j $(1 \leq s \leq 3)$. Then add the pairs (q_j, c_j^s), (p_j^s, z_j^1) and (p_j^{s+3}, z_j^2) to M. For each b $(1 \leq b \leq 3, b \neq s)$ add the pairs (p_j^b, c_j^b) and (p_j^{b+3}, z_j^{b+2}) to M. Finally add the pair (t_j, z_j^{s+2}) to M.

It remains to prove that M is stable. We now consider in turn those agents who might block M and prove that no agent can be involved in a blocking pair. No resident in Q may form part of a blocking pair since he can only potentially prefer a hospital in C, that ranks him last. Moreover no resident in T can form part of a blocking pair since he can only potentially prefer a hospital z_j $(3 \leq j \leq 5)$ that ranks him last.

Suppose that a resident couple (x_{4i+r}, k_{4i+r}) blocks M with $(c(x_{4i+r}), l_{4i+a})$, where $0 \leq i \leq n-1$, $0 \leq r \leq 3$ and $1 \leq a \leq 3$. Then (x_{4i+r}, k_{4i+r}) is assigned to their third choice pair.

Case (i): $r \in \{0, 1\}$. Then $f(v_i) = F$ since $(x_{4i+r}, y_{4i+r+1}) \in M$. It follows that $(k_{4i+r}, l_{4i+r+1}) \in M$. Let $c_j^s = c(x_{4i+r})$ $(1 \leq s \leq 3 \wedge 1 \leq j \leq m)$. As v_i does not make c_j true then $(p_j^s, c_j^s) \in M$. This means that c_j^s is assigned in M to the resident in first place in its preference list and thus cannot form part of a blocking pair, a contradiction.

Case (ii): $r \in \{2, 3\}$ Then $f(v_i) = T$ since $(x_{4i+r}, y_{4i+r}) \in M$. It follows that $(k_{4i+r}, l_{4i+r}) \in M$. Let $c_j^s = c(x_{4i+r})$ $(1 \leq s \leq 3, 1 \leq j \leq m)$. As $\bar{v}_i$ does not make c_j true then $(p_j^s, c_j^s) \in M$. This means that c_j^s is assigned in M its the resident in first place in its preference list and thus cannot form part of a blocking pair, a contradiction.

Now suppose that a resident couple (p_j^s, p_j^{s+3}) blocks M in I. Then both $(p_j^s, c_j^s) \in M$ and $(p_j^{s+3}, z_j^{s+2}) \in M$, and (p_j^s, p_j^{s+3}) prefers (z_j^1, z_j^2) to their joint assignee in M. At most one of $\{z_j^1, z_j^2\}$ can prefer the relevant member of $\{p_j^s, p_j^{s+3}\}$ to their partner, whilst the other prefers their current assignee in M and thus could not form part of a blocking pair, a contradiction. Hence, M is a complete stable matching in I.

Residents' Preferences

$(x_{4i}, k_{4i}) :$	(y_{4i}, l_{4i}) $(c(x_{4i}), l_{4i+1})$ (y_{4i+1}, l_{4i+1})	$(0 \leq i \leq n-1)$
$(x_{4i+1}, k_{4i+1}) :$	(y_{4i+1}, l_{4i+1}) $(c(x_{4i+1}), l_{4i+2})$ (y_{4i+2}, l_{4i+2})	$(0 \leq i \leq n-1)$
$(x_{4i+2}, k_{4i+2}) :$	(y_{4i+3}, l_{4i+3}) $(c(x_{4i+2}), l_{4i+2})$ (y_{4i+2}, l_{4i+2})	$(0 \leq i \leq n-1)$
$(x_{4i+3}, k_{4i+3}) :$	(y_{4i}, l_{4i}) $(c(x_{4i+3}), l_{4i+3})$ (y_{4i+3}, l_{4i+3})	$(0 \leq i \leq n-1)$
$(p_j^1, p_j^4) :$	(z_j^1, z_j^2) (c_j^1, z_j^3)	$(1 \leq j \leq m)$
$(p_j^2, p_j^5) :$	(z_j^1, z_j^2) (c_j^2, z_j^4)	$(1 \leq j \leq m)$
$(p_j^3, p_j^6) :$	(z_j^1, z_j^2) (c_j^3, z_j^5)	$(1 \leq j \leq m)$
$q_j :$	c_j^1 c_j^2 c_j^3	$(1 \leq j \leq m)$
$t_j :$	z_j^3 z_j^4 z_j^5	$(1 \leq j \leq m)$

Hospitals' Preferences

$y_{4i} :$	x_{4i} x_{4i+3}	$(0 \leq i \leq n-1)$
$y_{4i+1} :$	x_{4i+1} x_{4i}	$(0 \leq i \leq n-1)$
$y_{4i+2} :$	x_{4i+1} x_{4i+2}	$(0 \leq i \leq n-1)$
$y_{4i+3} :$	x_{4i+2} x_{4i+3}	$(0 \leq i \leq n-1)$
$l_{4i} :$	k_{4i+3} k_{4i}	$(0 \leq i \leq n-1)$
$l_{4i+1} :$	k_{4i} k_{4i+1}	$(0 \leq i \leq n-1)$
$l_{4i+2} :$	k_{4i+2} k_{4i+1}	$(0 \leq i \leq n-1)$
$l_{4i+3} :$	k_{4i+3} k_{4i+2}	$(0 \leq i \leq n-1)$
$z_j^1 :$	p_j^1 p_j^2 p_j^3	$(1 \leq j \leq m)$
$z_j^2 :$	p_j^6 p_j^5 p_j^4	$(1 \leq j \leq m)$
$z_j^3 :$	p_j^4 t_j	$(1 \leq j \leq m)$
$z_j^4 :$	p_j^5 t_j	$(1 \leq j \leq m)$
$z_j^5 :$	p_j^6 t_j	$(1 \leq j \leq m)$
$c_j^s :$	p_j^s $x(c_j^s)$ q_j	$(1 \leq j \leq m \wedge 1 \leq s \leq 3)$

Figure 3.7: Preference lists in I, the constructed instance of $(3,3)$-HRC.

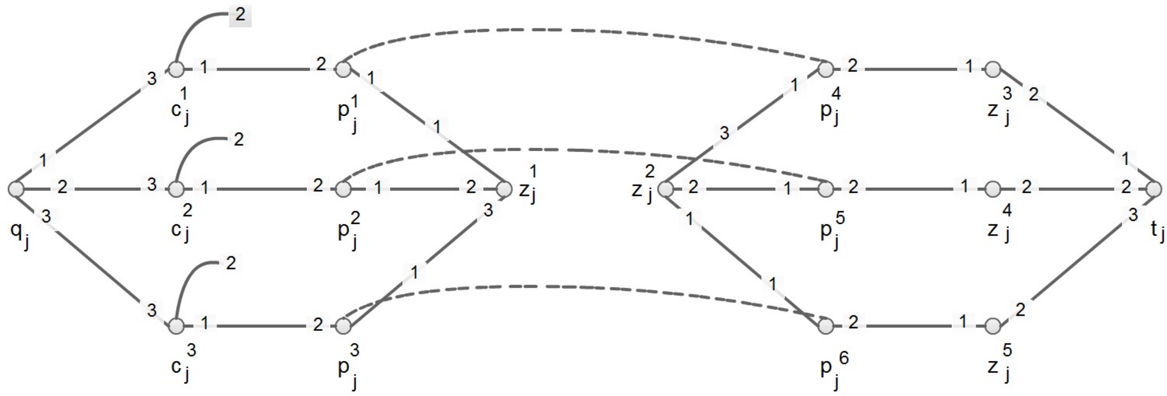

Figure 3.8: Diagram of subset of agents in I, the constructed instance of (3, 3)-HRC.

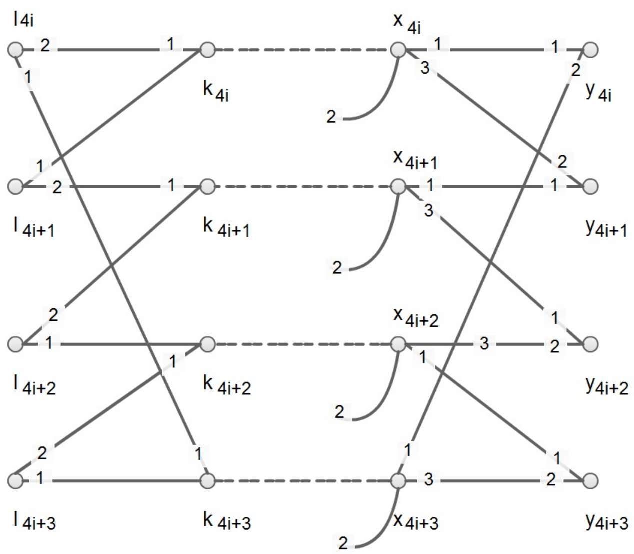

Figure 3.9: Diagram of subset of agents in I, the constructed instance of (3, 3)-HRC.

Conversely suppose that M is a complete stable matching in I. We form a truth assignment f in B as follows. For each i $(0 \leq i \leq n-1)$, $M \cap ((X_i \times Y_i) \cup (K_i \times L_i))$ is a perfect matching of $(X_i \cup Y_i) \cup (K_i \cup L_i)$. If $M \cap ((X_i \times Y_i) \cup (K_i \times L_i)) = T_i$, set v_i to be true under f. Otherwise $M \cap ((X_i \times Y_i) \cup (K_i \times L_i)) = F_i$, in which case we set v_i to be false under f.

Now let c_j be a clause in C $(1 \leq j \leq m)$. There exists some s $(1 \leq s \leq 3)$ such that $(q_j, c_j^s) \in M$. Let $x_{4i+r} = x(c_j^s)$, for some i $(0 \leq i \leq n-1)$ and r $(0 \leq r \leq 3)$. If $r \in \{0, 1\}$, then $(x_{4i+r}, y_{4i+r}) \in M$ and $(k_{4i+r}, l_{4i+r}) \in M$. Thus variable v_i is true under f, and hence clause c_j is true under f, since literal v_i occurs in c_j. If $r \in \{2, 3\}$, then $(x_{4i+r}, y_{4i+r+1}) \in M$ and $(k_{4i+r}, l_{4i+r+1}) \in M$ (where addition is taken modulo four). Thus variable v_i is false under f, and hence clause c_j is true under f, since literal $\bar{v}_i$ occurs in c_j. Hence f is a satisfying truth assignment of B. Thus, the claim holds and B is satisfiable if and only if I admits a complete stable matching. □

By adding an additional 'gadget' to the instance constructed in the proof of Lemma 3.5.1 we may prove that not only is the problem of deciding whether an instance of $(3, 3)$-HRC admits a complete stable matching NP-complete, but, the problem of deciding whether an instance of $(3, 3)$-HRC admits a stable matching, complete or otherwise, is also NP-complete. We state this result formally as Lemma 3.5.5. First we define an additional 'gadget' that will be added to the instance constructed in the proof of Lemma 3.5.1 to ensure the property that the only stable matchings admitted by the resulting instance are complete stable matchings.

Let I be the instance of $(3, 3)$-HRC as constructed in the proof of Lemma 3.5.1. We add additional residents and hospitals to I to obtain a new instance I' of $(3, 3)$-HRC as follows. For every $y_{4i+r} \in Y$ add further residents $U = \cup_{i=0}^{n-1} U_i, U_i = \{u_{4i+r}^s : 1 \leq s \leq 5 \wedge 0 \leq r \leq 3\}$ and further hospitals $H = \cup_{i=0}^{n-1} H_i, H_i = \{h_{4i+r}^s : 1 \leq s \leq 4 \wedge 0 \leq r \leq 3\}$ where each hospital has capacity one. The preference lists of the agents added in this fashion for a given $y_{4i+r} \in Y$ are shown in Figure 3.10 and also diagrammatically in Figure 3.11 where Φ_{4i+r} represents those preferences expressed by y_{4i+r} in $I' \setminus (U \cup H)$.

We now show that each y_{4i+r} $(0 \leq i \leq n-1, 0 \leq r \leq 3)$ must be assigned to some resident in X in any stable matching in I'. We state this result formally as Lemma 3.5.2 below.

Lemma 3.5.2. *In any stable matching M in I', for every $y_{4i+r} \in Y$, $M(y_{4i+r}) \in X$.*

Proof. Suppose not. Then y_{4i+r} is either unassigned in M or $M(y_{4i+r}) = u_{4i+r}^5$. If y_{4i+r} is unassigned, then (u_{4i+r}^5, y_{4i+r}) blocks M in I', a contradiction. Hence, $M(y_{4i+r}) = u_{4i+r}^5$ and thus $(u_{4i+r}^5, h_{4i+r}^1) \notin M$.

Assume that h_{4i+r}^1 is unassigned in M. Then (u_{4i+r}^1, u_{4i+r}^2) is unassigned in M and $M((u_{4i+r}^3, u_{4i+r}^4)) \neq (h_{4i+r}^1, h_{4i+r}^4)$. Thus, either (u_{4i+r}^3, u_{4i+r}^4) is unassigned in M or $M((u_{4i+r}^3,$

Residents' Preferences

(u^1_{4i+r}, u^2_{4i+r}) :	(h^1_{4i+r}, h^2_{4i+r})	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$
(u^3_{4i+r}, u^4_{4i+r}) :	(h^1_{4i+r}, h^4_{4i+r}) (h^3_{4i+r}, h^2_{4i+r})	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$
u^5_{4i+r} :	y_{4i+r} h^1_{4i+r}	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$

Hospitals' Preferences

y_{4i+r} :	Φ_{4i+r} u^5_{4i+r}	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$
h^1_{4i+r} :	u^5_{4i+r} u^1_{4i+r} u^3_{4i+r}	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$
h^2_{4i+r} :	u^4_{4i+r} u^2_{4i+r}	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$
h^3_{4i+r} :	u^3_{4i+r}	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$
h^4_{4i+r} :	u^4_{4i+r}	$(0 \leq i \leq n-1, 0 \leq r \leq 3)$

Figure 3.10: Added agents in I', the constructed instance of $(3,3)$-HRC.

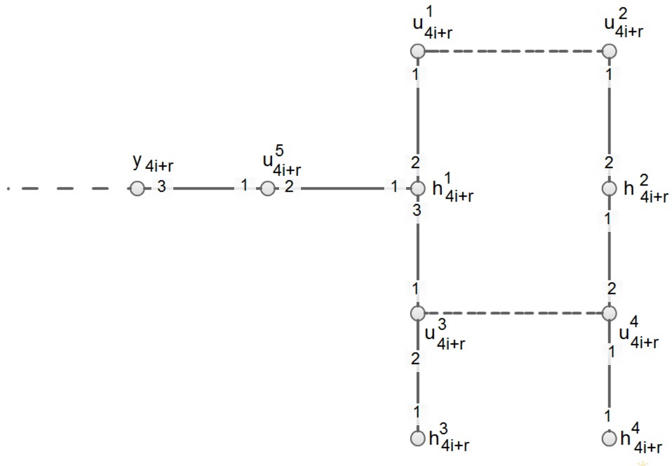

Figure 3.11: Diagram of added agents in I', the constructed instance of $(3,3)$-HRC.

$u^4_{4i+r})) = (h^3_{4i+r}, h^2_{4i+r})$. If (u^3_{4i+r}, u^4_{4i+r}) is unassigned in M, then (u^1_{4i+r}, u^2_{4i+r}) blocks M with (h^1_{4i+r}, h^2_{4i+r}) in I', a contradiction. Hence (u^3_{4i+r}, u^4_{4i+r}) must be assigned to (h^3_{4i+r}, h^2_{4i+r}) in M. However, now (u^3_{4i+r}, u^4_{4i+r}) blocks M with (h^1_{4i+r}, h^4_{4i+r}) in I', a contradiction. Thus, we have that h^1_{4i+r} must be assigned in M.

Now, assume that h^1_{4i+r} is assigned in M through the joint assignment of (u^1_{4i+r}, u^2_{4i+r}) to (h^1_{4i+r}, h^2_{4i+r}). Now M is blocked by (u^3_{4i+r}, u^4_{4i+r}) with (h^3_{4i+r}, h^2_{4i+r}) in I', a contradiction. Thus, h^1_{4i+r} must be assigned in M through the joint assignment of (u^3_{4i+r}, u^4_{4i+r}) to (h^1_{4i+r}, h^4_{4i+r}). However, now M is blocked by (u^1_{4i+r}, u^2_{4i+r}) with (h^1_{4i+r}, h^2_{4i+r}) in I', a contradiction. □

We now show through the following two Lemmas that if M' is a stable matching in I' and

$$M = M' \setminus \{(u^p_{4i+r}, h^q_{4i+r}) : 0 \leq i \leq n-1, 0 \leq r \leq 3, 1 \leq p \leq 5, 1 \leq q \leq 5\},$$

then M is a complete stable matching in I, the reduced $(3, 3)$-HRC instance obtained by removing all of the agents in $U \cup H$ from I'.

Lemma 3.5.3. *No hospital in $Z \cup C$ may be unassigned and no resident in $P \cup T \cup Q$ may be unassigned in any stable matching M in I'.*

Proof. Assume z^1_j is unassigned in M for some j $(1 \leq j \leq m)$. Thus $(p^s_j, z^2_j) \notin M$ $(4 \leq s \leq 6)$ as z^2_j must also be unassigned. Hence, (z^1_j, z^2_j) are unassigned and find (p^1_j, p^4_j) acceptable. Further, (p^1_j, p^4_j) prefer (z^1_j, z^2_j) to any other pair. Hence (z^1_j, z^2_j) blocks M with (p^1_j, p^4_j) in I', a contradiction. Thus, z^1_j must be assigned in any stable matching in I'. By a similar argument, z^2_j must be assigned in any stable matching in I'.

Assume t_j is unassigned in M for some j $(1 \leq j \leq m)$. If some z^s_j $(3 \leq s \leq 5)$ is unassigned, then (t^s_j, z^s_j) blocks M in I', a contradiction. Thus, $\{(p^4_j, z^3_j), (p^5_j, z^4_j), (p^6_j, z^5_j)\} \subseteq M$. It follows that z^2_j is unassigned, a contradiction. Thus, t_j must be assigned in any stable matching in I'.

Assume some resident p^s_j $(1 \leq j \leq m, 1 \leq s \leq 3)$ is unassigned in M. Then (p^s_j, p^{s+3}_j) is unassigned. Hence, (p^s_j, p^{s+3}_j) blocks M in I' with (c^s_j, z^{s+2}_j), a contradiction. Thus, all p^s_j $(1 \leq s \leq 6)$ must be assigned in any stable matching in I'.

Assume some z^s_j $(1 \leq j \leq m, 3 \leq s \leq 5)$ is unassigned in M. It follows that either t_j is unassigned in M or p^{s+1}_j is unassigned in M. As shown previously, t_j cannot be unassigned in a stable matching in I', thus $(t_j, z^b_j) \in M$ for some $b \in \{3, 4, 5\} \setminus \{s\}$. Further, as shown previously p^{s+1}_j cannot be unassigned so $(p^{s+1}_j, z^2_j) \in M$, thus p^{b+1}_j is unassigned in M, a contradiction. Thus, each z^s_j $(1 \leq j \leq m, 3 \leq s \leq 5)$ must be assigned in any stable matching in I'

Observe that no c_j^s $(1 \leq s \leq 3, 1 \leq j \leq m)$ can be assigned to $x(c_j^s)$, for otherwise $M(y_{4i+r}) \notin X$ for some $y_{4i+r} \in Y$ a contradiction to Lemma 3.5.2. Since z_j^1 must be assigned to some p_j^s $(1 \leq s \leq 3)$ and since no resident in P may be unassigned, then for $s' \in \{1,2,3\}$ such that $s' \neq s$, $c_j^{s'}$ must be assigned to the corresponding $p_j^{s'}$. Thus, $(q_j, c_j^s) \in M$ for otherwise (q_j, c_j^s) blocks M in I'. Thus all residents in Q and hospitals in C must be assigned in any stable matching admitted by I'. □

Lemma 3.5.4. *No hospital in $L \cup Y$ may be unassigned and no resident in $K \cup X$ may be unassigned in any stable matching M in I'.*

Proof. By Lemma 3.5.2, $M(y_{4i+r}) \in X$ for all $y_{4i+r} \in Y$. Hence $M(x_{4i+r}) \in Y$ for all $x_{4i+r} \in X$. It follows that $M(k_{4i+r}) \in L$ for all $k_{4i+r} \in K$ and thus $M(l_{4i+r}) \in K$ for all $l_{4i+r} \in L$. □

The proof of the previous three Lemmas allows us to now state the following more general Lemma.

Lemma 3.5.5. *Given an instance I' of $(3,3)$-HRC, the problem of deciding whether I' admits a stable matching is NP-complete.*

Proof. Let B be an instance of (2,2)-E3-SAT. Construct an instance I of $(3,3)$-HRC as described in the proof of Theorem 3.5.1 and as illustrated in Figure 3.7 and extend this instance to obtain the instance I' of $(3,3)$-HRC as described above using the gadget described in Figure 3.10.

Let f be a satisfying truth assignment of B. Define a matching M in I as in the proof of Theorem 3.5.1. Define a matching M' in I' as follows:

$$M' = M \cup \{(u_{4i+r}^5, h_{4i+r}^1), (u_{4i+r}^3, h_{4i+r}^3), (u_{4i+r}^4, h_{4i+r}^2) : 0 \leq i \leq n-1, 0 \leq r \leq 3\}$$

As shown previously no agent in $X \cup K \cup L \cup P \cup Q \cup T \cup Z \cup C$ can block M' in I'. By Lemma 3.5.2, $M'(y_{4i+r}) \in X$ for any stable matching M' in I'. Thus, it follows that $M'(u_{4i+r}^5) = h_{4i+r}^1$ otherwise (u_{4i+r}^5, h_{4i+r}^1) would block M' in I'. Further $M'((u_{4i+r}^3, u_{4i+r}^4)) = (h_{4i+r}^3, h_{4i+r}^2)$ for otherwise M' must admit a blocking pair amongst the agents in the sub instance S. Thus, no agent in $Y \cup U \cup H$ can block M' in I'. Thus M' is a stable matching in I'.

Conversely, suppose that M' is a stable matching in I'. By Lemma 3.5.2, every y_{4i+r} is assigned in M' to a resident in X. By Lemmas 3.5.3 and 3.5.4 every agent in $K \cup X \cup P \cup T \cup Q \cup Z \cup C$ is assigned in M'.

Now let

$$M = M' \setminus \{(u^p_{4i+r}, h^q_{4i+r}) : 0 \leq i \leq n-1, 0 \leq r \leq 3, 1 \leq p \leq 5, 1 \leq q \leq 5\}$$

Then M is a complete stable matching in I, the reduced $(3,3)$-HRC instance obtained by removing all of the agents in $U \cup H$ from I'. By the proof of Theorem 3.5.1 we can obtain a satisfying truth assignment for B from M. □

Recall that an instance of $(3,3)$-HRC DUAL MARKET is an instance of HRC DUAL MARKET in which no resident, couple or hospital has a preference list of length greater than three. We now show that the instance described in Theorem 3.5.5 represents a dual market and thus we are able to show that deciding whether a stable matching exists in an instance of $(3,3)$-HRC DUAL MARKET is also NP-complete.

Theorem 3.5.6. *Given an instance of* $(3,3)$-HRC DUAL MARKET, *the problem of deciding whether a stable matching exists is NP-complete. The result holds even if each hospital has capacity one and the preference list of each single resident, couple and hospital is derived from a strictly ordered master list of hospitals, pairs of hospitals and residents respectively.*

Proof. Let I' be the instance of $(3,3)$-HRC constructed in the proof of Lemma 3.5.5. The residents in I' can be partitioned into two disjoint sets, $R_1 = X \cup P_1 \cup Q \cup U_1$, where $P_1 = \{p^s_j : 1 \leq s \leq 3\}$ and $U_1 = \{u^s_{4i+r} : s \in \{1,3,5\}\}$, and $R_2 = K \cup P_2 \cup T \cup U_2$, where $P_2 = \{p^s_j : 4 \leq s \leq 6\}$ and $U_2 = \{u^s_{4i+r} : s \in \{2,4\}\}$. Further, the hospitals in I' may also be partitioned into two disjoint sets, $A_1 = Y \cup Z_1 \cup C \cup H_1$, where $Z_1 = \{z^1_j : 1 \leq j \leq m\}$ and $H_1 = \{h^s_{4i+r} : 0 \leq i \leq n-1, 0 \leq r \leq 3, s \in \{1,3\}\}$, and $A_2 = L \cup Z_2 \cup H_2$, where $Z_2 = \{z^s_j : 2 \leq s \leq 5 \wedge 1 \leq j \leq m\}$ and $H_2 = \{h^s_{4i+r} : 0 \leq i \leq n-1, 0 \leq r \leq 3, s \in \{2,4\}\}$.

A resident $r \in R_i$ finds acceptable only those hospitals in A_i and a hospital $h \in A_i$ finds acceptable only those residents in R_i. From this construction it can be seen that the instance I' represents a dual market consisting of two disjoint markets, $(R_1 \cup A_1)$ and $(R_2 \cup A_2)$.

The master lists shown in Figures 3.12, 3.13 and 3.14 indicate that the preference list of each single resident, couple and hospital may be derived from a master list of hospital pairs, residents and hospitals respectively. Thus, the result follows immediately from Lemma 3.5.5. □

$$L_i^1 : (y_{4i}, l_{4i}), (c(x_{4i}), l_{4i+1}), (y_{4i+1}, l_{4i+1}), (c(x_{4i+1}), l_{4i+2}), (c(x_{4i+3}), l_{4i+3}), (y_{4i+3}, l_{4i+3}), (c(x_{4i+2}), l_{4i+2}), (y_{4i+2}, l_{4i+2}) \quad (0 \le i \le n-1)$$

$$L_{i,r}^2 : (h_{4i+r}^1, h_{4i+r}^2), (h_{4i+r}^1, h_{4i+r}^4), (h_{4i+r}^3, h_{4i+r}^2) \quad (0 \le i \le n-1, 0 \le r \le 3) \quad L_j^3 : (z_j^1, z_j^2), (c_j^1, z_j^3), (c_j^2, z_j^4), (c_j^3, z_j^5) \quad (1 \le j \le m)$$

$$\text{Master List} : L_0^1\ L_1^1 \dots L_{n-1}^1\ L_{0,0}^2\ L_{0,1}^2 \dots L_{0,3}^2\ L_{1,0}^2\ L_{1,1}^2 \dots L_{(n-1),3}^2\ L_1^3 L_2^3 \dots L_m^3$$

Figure 3.12: Master list of preferences for resident couples in $(3,3)$-COM HRC instance I.

$$L_j^4 : c_j^1, c_j^2, c_j^3, z_j^3, z_j^4, z_j^5 \quad (1 \le j \le m) \quad L_{i,r}^5 : y_{4i+r}, h_{4i+r}^1 \quad (0 \le i \le n-1, 0 \le r \le 3)$$

$$\text{Master List} : L_1^4 L_2^4 \dots L_m^4 L_{0,0}^5\ L_{0,1}^5 \dots L_{0,3}^5\ L_{1,0}^5\ L_{1,1}^5 \dots L_{(n-1),3}^5$$

Figure 3.13: Master list of preferences for single residents in $(3,3)$-COM HRC instance I.

$$L_i^6 : x_{4i+1}, x_{4i}, x_{4i+2}, x_{4i+3} \quad (0 \le i \le n-1) \quad L_i^7 : k_{4i+3}, k_{4i}, k_{4i+2}, k_{4i+1} \quad (0 \le i \le n-1)$$

$$L_j^8 : p_j^1, p_j^2, p_j^3 \quad (0 \le j \le m) \quad L_j^9 : p_j^6, p_j^5, p_j^4 \quad (0 \le j \le m)$$

$$L_j^{10} : q_1, q_2 \dots q_j \quad (0 \le j \le m) \quad L_j^{11} : t_1, t_2 \dots t_j \quad (0 \le j \le m)$$

$$L_{i,r}^{12} : u_{4i+r}^5, u_{4i+r}^1, u_{4i+r}^3, u_{4i+r}^4, u_{4i+r}^2 \quad (0 \le i \le n-1, 0 \le r \le 3)$$

$$\text{Master List} : L_1^8\ L_2^8 \dots L_m^8 L_0^6 L_1^6 \dots L_{n-1}^6 L_0^7 L_1^7 \dots L_{n-1}^7 L_1^9\ L_2^9 \dots L_m^9\ L_1^{10} \dots L_m^{10}\ L_1^{11} \dots L_m^{11}\ L_{0,0}^{12}\ L_{0,1}^{12} \dots L_{0,3}^{12}\ L_{1,0}^{12}\ L_{1,1}^{12} \dots L_{1,3}^{12}\ L_{2,0}^{12} \dots L_{(n-1),3}^{12}$$

Figure 3.14: Master list of preferences for hospitals in $(3,3)$-COM HRC instance I.

3.6 Complexity results for $(\infty, 1, \infty)$-HR PA

3.6.1 $(\infty, 1, \infty)$-HR PA is NP-complete

We denote by $(\infty, 1, \infty)$-HR PA the variant of HR PA in which each resident's preference list contains either: (i) exactly one application to an unordered pair of hospitals and no applications to individual hospitals *or* (ii) an unbounded number of applications to individual hospitals and no applications to unordered hospital pairs. We now establish that the problem of deciding whether an instance of $(\infty, 1, \infty)$-HR PA admits a stable matching is NP-complete.

Theorem 3.6.1. *Given an instance of $(\infty, 1, \infty)$-HR PA, the problem of deciding whether there exists a stable matching is NP-complete. The result holds even if each hospital has capacity one.*

Proof. The proof of this result uses a reduction from $(\infty, 1, \infty)$-HRC. Theorem 3.2.1 in Section 3.2.1 proves that deciding whether an instance of $(\infty, 1, \infty)$-HRC admits a stable matching is NP-complete even if all of the hospitals have capacity one.

The problem $(\infty, 1, \infty)$-HR PA is in NP, as a given assignment may be verified to be a stable matching in polynomial time. To show NP-hardness we now present a polynomial-time reduction from an instance of $(\infty, 1, \infty)$-HRC to an instance of $(\infty, 1, \infty)$-HR PA. Let I be an instance of $(\infty, 1, \infty)$-HRC with residents $R = \{r_1, r_2, \ldots r_{n_1}\}$ and hospitals $H = \{h_1, h_2, \ldots h_{n_2}\}$ where each hospital $h_j \in H$ has a capacity $c_j = 1$. Without loss of generality let the first $2c$ residents in I be involved in couples of the form (r_{2s-1}, r_{2s}) $(1 \leq s \leq c)$. Each couple (r_{2s-1}, r_{2s}) $(1 \leq s \leq c)$ in I expresses exactly one joint preference for a hospital pair (h_{j_1}, h_{j_2}) $(h_{j_1} \in H \wedge h_{j_2} \in H \wedge h_{j_1} \neq h_{j_2})$. Each single resident r_t $(2c+1 \leq t \leq n_1)$ has a strictly ordered preference list over some subset of the hospitals in H.

We form an instance J of $(\infty, 1, \infty)$-HR PA as follows. For each couple (r_{2s-1}, r_{2s}) $(1 \leq s \leq c)$ in I create a resident a_s in J. For each single resident r_t $(2c + 1 \leq t \leq n_1)$ create a resident a_{t-c} in J. For each hospital $h_j \in H$ create a hospital b_j in J with capacity one. Let $A = \{a_1, a_2, \ldots a_{n_3}\}$ be the $n_3 = n_1 - c$ residents created in J by this process and let $B = \{b_1, b_2, \ldots b_{n_2}\}$ be the hospitals created in J in this process.

The preference list of a_s $(1 \leq s \leq c)$ in J is constructed from (r_{2s-1}, r_{2s})'s preference list in I as follows. Let the pair (h_{j_1}, h_{j_2}) $(h_{j_1} \in H \wedge h_{j_2} \in H \wedge h_{j_1} \neq h_{j_2})$ be (r_{2x-1}, r_{2x})'s acceptable hospital pair. Then, the preference list of a_x in J contains only the pair $\{b_{j_1}, b_{j_2}\}$. The preference list of each a_t $(c+1 \leq t \leq n_3)$ in J is constructed from r_{t+c}'s preference list in I as follows. For each p $(1 \leq p \leq l(r_{t+c}))$ such that $pref(r_{t+c}, p) = h_j$, let $pref(a_t, p) = b_j$ in J. The preference list of b_j in J is constructed from h_j's preference list in I as follows. For each q $(1 \leq q \leq l(h_j)$ suppose $pref(h_j, q) = r_i$. If $i \geq 2c+1$, then r_i is a single resident in I and we let $pref(b_j, q) = a_{i-c}$ in J. Otherwise, $i \leq 2c$ and r_i is a member of a couple in I. If i is even, then let $pref(b_j, q) = a_k$ where $k = i/2$, otherwise i is odd and we let $pref(b_j, q) = a_k$ where $k = (i+1)/2$.

Thus, each resident $a_k \in A$ has a preference list of length $l(a_k)$ containing either exactly one pair of hospitals or an unbounded number applications to individual hospitals. Further, each hospital $b_j \in B$ has capacity one. This completes the reduction. Clearly, this reduction can be carried out in polynomial-time.

We claim that I admits a stable matching if and only if J admits a stable matching. Suppose that M_I is a stable matching in I. We form a set of pairs M_J in J from M_I as follows. If (r_{2s-1}, r_{2s}) $(1 \leq s \leq c)$ is assigned to the hospital pair (h_{j_1}, h_{j_2}) in M_I, then add the pairs (a_s, b_{j_1}) and (a_s, b_{j_2}) to M_J. If a single resident r_t $(2c+1 \leq t \leq n_1)$ is assigned to an individual hospital h_j in M_I, then add the pair (a_{t-c}, b_j) to M_J.

Since each couple in I is assigned in M_I to at most one acceptable pair or is unassigned, and each individual resident in I is assigned in M_I to at most one hospital or is unassigned, each resident in J is assigned to at most one application. Since no hospital is oversubscribed in M_I the same is true of the hospital in M_J. Hence M_J is a matching in J.

It remains to prove that M_J is stable. Assume not. Firstly assume that there exists in J some resident a_k $(c+1 \leq k \leq n_3)$ and a hospital b_j $(1 \leq j \leq n_2)$ where $pref(a_k, p) = b_j$ for some p $(1 \leq p \leq l(a_k))$, such that a_k prefers b_j to $M_J(a_k)$ or is unassigned in M_J and also b_j prefers a_k to $M_J(b_j)$ or is undersubscribed. Now we have that in I a resident r_{k+c} is unassigned or prefers h_j to $M_I(r_{k+c})$ and h_j prefers r_{k+c} to $M_I(h_j)$ or is undersubscribed, a contradiction to the stability of M_I in I.

Now assume there exists in J some resident a_k $(1 \leq k \leq c)$ and a pair of hospitals $\{b_{j_1}, b_{j_2}\}$ $(1 \leq j_1 \leq n_2, 1 \leq j_2 \leq n_2, j_1 \neq j_2)$ where $pref(a_k, p) = \{b_{j_1}, b_{j_2}\}$ for some p $(1 \leq p \leq l(a_k))$ such that a_k is unassigned, b_{j_1} prefers a_k to $M_J(b_{j_1})$ or is undersubscribed and b_{j_2} prefers a_k to $M_J(b_{j_2})$ or is undersubscribed. Now we have in I that (r_{2k-1}, r_{2k}) is unassigned in M_I and finds acceptable one of two possible ordered hospital pairs, namely either (h_{j_1}, h_{j_2}) or (h_{j_2}, h_{j_1}). Assume firstly that (r_{2k-1}, r_{2k}) $(1 \leq k \leq c)$ finds (h_{j_1}, h_{j_2}) acceptable. Then h_{j_1} prefers r_{2k-1} to $M_I(h_{j_1})$ and h_{j_2} prefers r_{2k} to $M_I(b_{j_2})$, a contradiction to the stability of M_I. Now, assume (r_{2k-1}, r_{2k}) finds (h_{j_2}, h_{j_1}) acceptable. Then h_{j_2} prefers r_{2k-1} to $M_I(h_{j_2})$ and h_{j_1} prefers r_{2k} to $M_I(b_{j_1})$, a contradiction to the stability of M_I. Thus no agent who finds acceptable a pair of hospitals can block M_J in J. Hence, M_J must be a stable matching in J.

Conversely, suppose that M_J is a stable matching in J. Define a set of pairs M_I in I as follows. If a_k $(1 \leq k \leq n_3)$ is assigned to an application $\{b_{j_1}, b_{j_2}\}$ $(1 \leq j_1 \leq n_2, 1 \leq j_2 \leq n_2, j_1 \neq j_2)$ in M_J, then the couple (r_{2k-1}, r_{2k}) $(1 \leq k \leq c)$ in I finds acceptable one of two hospital pairs, namely either (h_{j_1}, h_{j_2}) or (h_{j_2}, h_{j_1}). In the former case add the pairs

(r_{2k-1}, h_{j_1}) and (r_{2k}, h_{j_2}) to M_I. In the latter case add the pairs (r_{2k-1}, h_{j_2}) and (r_{2k}, h_{j_1}) to M_I. If $a_k \in A$ is assigned to a single hospital b_j in M_J, then add the pair (r_{k+c}, h_j) to M_I. It follows that: each couple is assigned to exactly one hospital pair or is unassigned (but not both); each single resident is assigned in I to exactly one hospital or is unassigned (but not both); and each hospital $h_j \in H$ has at most one assignee in M_I. Thus M_I is a matching in I.

It remains to prove that M_I is stable. Assume not. Suppose firstly that M_I is blocked by a single resident r_y $(2c + 1 \leq y \leq n_1)$ and an acceptable hospital $h_j \in H$. Then, r_y is unassigned or prefers h_j to $M_I(r_y)$ and h_j prefers r_y to $M_I(h_j)$ or is undersubscribed. Now we have in J, that a_{y-c} prefers b_j to $M_J(a_{y-c})$ or is unassigned and b_j prefers a_{y-c} to $M_J(b_j)$ or is unassigned, a contradiction to the stability of M_J. Thus, M_I is not blocked by a single resident R and a hospital H.

Now, suppose M_I is blocked by a couple (r_{2x-1}, r_{2x}) $(1 \leq x \leq c)$ and a hospital pair (h_{j_1}, h_{j_2}). Since (r_{2x-1}, r_{2x}) expresses only one joint preference and is part of a blocking pair it follows that (r_{2x-1}, r_{2x}) must be unassigned in M_I. Further, h_{j_1} prefers r_{2x-1} to $M_I(h_{j_1})$ and h_{j_2} prefers r_{2x} to $M_I(h_{j_2})$. Now we have that in J, a_x is unassigned in M_J and finds $\{b_{j_1}, b_{j_2}\}$ acceptable. Further b_{j_1} prefers a_x to $M(h_{j_1})$ and b_{j_2} prefers a_x to $M(h_{j_2})$, in contradiction to the stability of M_J. Thus, M_I is not blocked by a couple and a hospital pair. Thus none of the agents in I may block M_I in I and the result is proven. □

3.7 Efficiently solvable variants of HRC

In this section we describe two highly restricted variants of HRC and prove that we can find a maximum cardinality stable matching or report that no stable matching exists in polynomial-time. First, in Section 3.7.1 we define formally the concept of a fixed assignment. Intuitively a fixed assignment is a resident-hospital pair that must be assigned in any stable matching in an instance of HRC. We apply the concept of a fixed assignment to show that any instance of $(\infty, \infty, 1)$-HRC admits exactly one stable matching which can be found in polynomial-time. In Section 3.7.2 we rely on the property that all the fixed assignments in an arbitrary instance of HRC may be satisfied in polynomial-time to prove that we can find a maximum cardinality stable matching in an instance of $(2, 1, 2)$-HRC or report that no stable matching exists in polynomial-time.

3.7.1 Fixed assignments in HRC

In an instance of HRC some agents may prefer each other to any other possible partner and hence must be assigned to each other in any stable matching in the instance. We describe such a pair as a *fixed assignment* and define this concept formally in Lemma 3.7.1.

Lemma 3.7.1. *Let I be an arbitrary instance of* HRC*. Any pair of either of the following two types must be assigned to each other in any stable matching in I. We describe such a pair as a* fixed assignment*:*

(i) If a single resident r_i has a hospital h_j in first place on his preference list and r_i is within the first c_j places on h_j's preference list (where c_j is the capacity of h_j) then (r_i, h_j) must be in any stable matching in I.

(ii) If a resident couple (r_i, r_j) has a hospital pair (h_p, h_q) in first place on its joint preference list and r_i is within the first c_p places on h_p's preference list (where c_p is the capacity of h_p) and also r_j is within the first c_q places on h_q's preference list (where c_q is the capacity of h_q) then (r_i, r_j) must be assigned to (h_p, h_q) in any stable matching in I.

Proof. The proof of the Lemma follows immediately from the fact that any matching M in which (r_i, r_j) and (h_p, h_q) are not assigned to each other will be blocked by (r_i, r_j) with (h_p, h_q) and similarly any matching M in which r_i is not assigned to h_j will be blocked by (r_i, h_j). □

We *satisfy* a fixed assignment (r_i, h_j) in an arbitrary instance I of HRC by ensuring that $(r_i, h_j) \in M$ for any stable matching M in I. Now, if $(r_i, h_j) \in M$ then clearly no other hospital may be assigned in M to r_i and hence r_i can be deleted from the preference list of each other hospital in which he appears. Moreover, in the event that h_j becomes fully subscribed by accepting r_i as an assignee, h_j can be deleted from the preference list of each resident other than r_i in which it appears. However, satisfying an arbitrary fixed assignment in I and making the corresponding deletions from the preference lists may expose another fixed assignment in the resulting reduced instance of HRC which must then also be satisfied in any stable matching. If we continue satisfying fixed assignments until no more fixed assignments are exposed then we say all fixed assignments have been *iteratively satisfied* in I. We use the concept of a fixed assignment in the proof of Proposition 3.7.2 to show that a stable solution can be found in an instance of $(\infty, \infty, 1)$-HRC in polynomial time.

Proposition 3.7.2. *An instance I of $(\infty, \infty, 1)$-HRC admits exactly one stable matching and this unique stable matching may be found in time polynomial in the number of residents in I.*

Proof. Consider an arbitrary single resident r in I. Let the hospital in first place on resident r's preference list be h. Since r must be in first place in h's preference list (as it is the only preference expressed by h), the pair (r, h) represents a fixed assignment in I. Thus, any single resident in I must be part of exactly one fixed assignment in I and this may be satisfied by assigning each single resident to the hospital in first place on his preference list.

Now, consider an arbitrary couple (r_i, r_j) in I. Let the hospital pair (h_p, h_q) be in first place on couple (r_i, r_j)'s joint preference list. Clearly, since r_i (respectively r_j) is in first place on h_p's (respectively h_q's) preference list, (r_i, r_j) with (h_p, h_q) represents a fixed assignment in I. Thus, any resident couple in I must be part of exactly one fixed assignment in I and this may be satisfied by assigning each couple to the hospital pair in first place on their joint preference list.

Hence, the fixed assignments involving both the single residents and the couples in I may be identified in time polynomial in the number of residents in I and thus the unique stable matching admitted by I found in polynomial time. □

3.7.2 $(2, 1, 2)$-HRC is efficiently solvable

Let I be an instance of $(2, 1, 2)$-HRC in which there are no remaining unsatisfied fixed assignments. In Lemma 3.7.3 we use the absence of fixed assignments in I to infer that I must be constructed from the union of a finite number of disjoint discrete subinstances of $(2, 1, 2)$-HRC and further that each disjoint subinstance $I' \subseteq I$ must be of the form shown in Figure 3.15. Let $I' \subseteq I$ be one of these disjoint subinstances of I. We prove by Lemma 3.7.4 that whether I' admits a stable matching is determined by the number of couples involved in I'; moreover I' admits a stable matching if and only if the number of couples involved in I' is even.

Lemma 3.7.3. *An arbitrary instance of $(2, 1, 2)$-HRC involving at least one couple and in which all fixed assignments have been iteratively satisfied must be constructed from subinstances of the form shown in Figure 3.15 in which all of the hospitals have capacity one.*

Proof. Let I be an arbitrary instance of $(2, 1, 2)$-HRC in which all fixed assignments have been iteratively satisfied. Observe that if a couple expresses a preference for a hospital pair (h_p, h_p) this would represent a fixed assignment, a contradiction. Thus, no couple may express such a preference in I. We now show how the absence of fixed assignments in I allows us to infer the preference lists for all of the agents involved in I.

Let $(r^1_{c_1}, r^2_{c_1})$ be a couple in I and further let $(h^0_{c_1}, h^1_{c_1})$ be the hospital pair for which $(r^1_{c_1}, r^2_{c_1})$ expresses a preference. Since all fixed assignments have been iteratively satisfied by construction, it cannot be the case that *both*:

(i) $h^0_{c_1}$ has capacity two or has $r^1_{c_1}$ in first place in its preference list *and*

(ii) $h^1_{c_1}$ has capacity two or has $r^2_{c_1}$ in first place in its preference list.

Residents			Hospitals		
$(r^1_{c_1}, r^2_{c_1})$	:	$(h^0_{c_1}, h^1_{c_1})$	$h^0_{c_1}$	:	$r^1_{c_1}$ $r^{n_N}_{s_N}$
$r^1_{s_1}$	:	$h^2_{c_1}$ $h^1_{c_1}$	$h^1_{c_1}$	:	$r^1_{s_1}$ $r^2_{c_1}$
$r^2_{s_1}$	:	$h^3_{c_1}$ $h^2_{c_1}$	$h^2_{c_1}$	:	$r^2_{s_1}$ $r^1_{s_1}$
	⋮		⋮		
$r^{n_1}_{s_1}$	:	$h^{n_1+1}_{c_1}$ $h^{n_1}_{c_1}$	$h^{n_1}_{c_1}$	:	$r^{n_1}_{s_1}$ $r^{n_1-1}_{s_1}$
$(r^1_{c_2}, r^2_{c_2})$	:	$(h^{n_1+1}_{c_1}, h^1_{c_2})$	$h^{n_1+1}_{c_1}$	:	$r^1_{c_2}$ $r^{n_1}_{s_1}$
$r^1_{s_2}$	:	$h^2_{c_2}$ $h^1_{c_2}$	$h^1_{c_2}$	:	$r^1_{s_2}$ $r^2_{c_2}$
$r^2_{s_2}$	:	$h^3_{c_2}$ $h^2_{c_2}$	$h^2_{c_2}$	:	$r^2_{s_2}$ $r^1_{s_2}$
	⋮		⋮		
$r^{n_2}_{s_2}$	:	$h^{n_2+1}_{c_2}$ $h^{n_2}_{c_2}$	$h^{n_2}_{c_2}$	:	$r^{n_2}_{s_2}$ $r^{n_2-1}_{c_2}$
$(r^1_{c_3}, r^2_{c_3})$	:	$(h^{n_2+1}_{c_2}, h^1_{c_3})$	$h^{n_2+1}_{c_2}$	:	$r^1_{c_3}$ $r^{n_2}_{s_2}$
$r^1_{s_3}$	:	$h^2_{c_3}$ $h^1_{c_3}$	$h^1_{c_3}$	:	$r^1_{s_3}$ $r^2_{c_3}$
$r^2_{s_3}$	:	$h^3_{c_3}$ $h^2_{c_3}$	$h^2_{c_3}$	:	$r^2_{s_3}$ $r^1_{s_3}$
	⋮		⋮		
$r^{n_{N-1}}_{s_{N-1}}$	:	$h^{n_{N-1}+1}_{c_{N-1}}$ $h^{n_{N-1}}_{c_{N-1}}$	$h^{n_{N-1}}_{c_{N-1}}$	:	$r^{n_{N-1}}_{s_{N-1}}$ $r^{n_{N-1}-1}_{c_{N-1}}$
$(r^1_{c_N}, r^2_{c_N})$	:	$(h^{n_{N-1}+1}_{c_{N-1}}, h^1_{c_N})$	$h^{n_{N-1}+1}_{c_{N-1}}$	:	$r^1_{c_N}$ $r^{n_{N-1}}_{s_{N-1}}$
$r^1_{s_N}$	:	$h^2_{c_N}$ $h^1_{c_N}$	$h^1_{c_N}$	:	$r^1_{s_N}$ $r^2_{c_N}$
$r^2_{s_N}$	:	$h^3_{c_N}$ $h^2_{c_N}$	$h^2_{c_N}$	:	$r^2_{s_N}$ $r^1_{s_N}$
	⋮		⋮		
$r^{n_N}_{s_N}$	:	$h^0_{c_1}$ $h^{n_N}_{c_N}$	$h^{n_N}_{c_N}$	:	$r^{n_N}_{s_N}$ $r^{n_{N-1}}_{s_N}$

Figure 3.15: An instance of $(2, 1, 2)$-HRC containing an arbitrary number of couples and an arbitrary number of residents that has no unsatisfied fixed assignments.

Without loss of generality, assume that $h^1_{c_1}$ has capacity one and does not have $r^2_{c_1}$ in first place in its preference list. Hence there exists some other resident r_x who is preferred by $h^1_{c_1}$. Clearly, this resident is either a member of a couple or is a single resident. We now consider both of these cases and show that the we must arrive at the same outcome in either case. In what follows n_k $(1 \leq k \leq n_N)$ represens the number of single residents generated following couple c_k as the preference lists of the residents are inferred in the proof.

Case (i): r_x is single and thus $n_1 > 0$. In this case let $r_x = r^1_{s_1}$. Since $r^1_{s_1}$ is in first place in the preference list of $h^1_{c_1}$, to prevent a fixed assignment, there must exist another hospital that is preferred by $r^1_{s_1}$; let this be $h^2_{c_1}$. If $h^2_{c_1}$ has capacity two then $(r^1_{s_1}, h^2_{c_1})$ represents a fixed assignment, a contradiction. Hence, $h^2_{c_1}$ must have capacity one.

Now, since $r^1_{s_1}$ has $h^2_{c_1}$ in first place in its preference list, there must exist some other resident who is preferred by $h^2_{c_1}$. We consider first the case where each newly generated resident is single. Hence, let this new resident be a single resident, $r^2_{s_1}$. Since $r^2_{s_1}$ is in first place on

Residents

$(r^1_{c_1}, r^2_{c_1})$	:	$(h^0_{c_1}, h^1_{c_1})$	
$(r^1_{c_k}, r^2_{c_k})$	:	$(h^{n_{k-1}+1}_{c_{k-1}}, h^1_{c_k})$	$2 \leq k \leq N, n_{k-1} > 0$
$(r^1_{c_k}, r^2_{c_k})$	:	$(h^1_{c_{k-1}}, h^1_{c_k})$	$2 \leq k \leq N, n_k = 0$
$r^a_{s_k}$	:	$h^{a+1}_{c_k} \quad h^a_{c_k}$	$1 \leq k \leq N, 1 \leq a \leq n_k, n_k > 0$

Hospitals

$h^0_{c_1}$	:	$r^1_{c_1} \quad r^{n_N}_{s_N}$	if $n_N > 0$
$h^0_{c_1}$	:	$r^1_{c_1} \quad r^2_{c_N}$	if $n_N = 0$
$h^1_{c_1}$	:	$r^1_{s_1} \quad r^2_{c_1}$	if $n_1 > 0$
$h^1_{c_1}$	:	$r^1_{c_2} \quad r^2_{c_1}$	if $n_1 = 0$
$h^1_{c_k}$	:	$r^1_{s_k} \quad r^2_{c_k}$	$2 \leq k \leq N$, if $n_k > 0$
$h^1_{c_k}$	:	$r^1_{c_{k+1}} \quad r^2_{c_k}$	$2 \leq k \leq N$, if $n_k = 0$
$h^a_{c_k}$	:	$r^{a+1}_{s_k} \quad r^a_{s_k}$	$1 \leq k \leq N, 1 \leq a \leq n_k, n_k > 0$

Figure 3.16: An exactly equivalent description of the instance shown in Figure 3.15

the preference list of $h^2_{c_1}$ there must exist another hospital which is preferred by $r^2_{s_1}$; let this new hospital be $h^3_{c_1}$. Assume $h^3_{c_1}$ has capacity two. In that case $(r^1_{s_1}, h^2_{c_1})$ represents a fixed assignment, a contradiction. Hence, $h^3_{c_1}$ must have capacity one.

We may continue constructing a sequence of distinct single residents and hospitals of capacity one, but as the number of single residents is finite, ultimately we must eventually arrive at a resident who is a member of a couple; let this resident be $r^1_{c_2}$. Without loss of generality suppose that $r^1_{c_2}$ is the first member of the couple to which he belongs. Let $r^{n_1}_{s_1}$ be the final single resident constructed in the preceding sequence.

It follows that $r^{n_1}_{s_1}$ prefers some hospital $h^{n_1+1}_{c_1}$ of capacity one to $h^{n_1}_{c_1}$. If $h^{n_1+1}_{c_1} = h^0_{c_1}$ then I contains precisely one couple and the instance is of the form shown in Figure 3.15 where $N = 1$ and $n_1 > 0$. Otherwise $h^{n_1+1}_{c_1}$ is a new hospital of capacity one that prefers $r^1_{c_2}$ to $r^{n_1}_{s_1}$. Since $h^{n_1+1}_{c_1}$ has $r^1_{c_2}$ in first place on its preference list, it must be the case that $r^1_{c_2}$ expresses a joint preference as part of the couple $(r^1_{c_2}, r^2_{c_2})$ for a hospital pair involving $h^{n_1+1}_{c_1}$; let this pair be $(h^{n_1+1}_{c_1}, h^1_{c_2})$. Since $h^{n_1+1}_{c_1}$ has $r^1_{c_2}$ in first place on its preference list, $h^1_{c_2}$ must be of capacity one and prefer some other resident to $r^2_{c_2}$, otherwise $(r^1_{c_2}, r^2_{c_2})$ represents a fixed assignment with $(h^{n_1+1}_{c_1}, h^1_{c_2})$, a contradiction. Now, let this other resident be r_y.

Case(ii): r_x is a member of a couple and thus $n_1 = 0$. Let $r_x = r^1_{c_2}$. Then $h^1_{c_1}$ prefers $r^1_{c_2}$ to $r^2_{c_1}$. Assume that $r^1_{c_2}$ is part of a couple $(r^1_{c_2}, r^2_{c_2})$ and further assume that $(r^1_{c_2}, r^2_{c_2})$ finds

$(h^1_{c_1}, h^1_{c_2})$ acceptable. If $h^1_{c_2} = h^0_{c_1}$ then I contains exactly two couples and is of the form shown in Figure 3.15 with $N = 2$ and $n_1 = n_2 = 0$. (In this case $h^0_{c_1}$ prefers $r^1_{c_1}$ to $r^2_{c_2}$.) Otherwise, $h^1_{c_2}$ is a new hospital which must be of capacity one, or $(r^1_{c_2}, r^2_{c_2})$ represents a fixed assignment with $(h^1_{c_1}, h^1_{c_2})$, and moreover $h^1_{c_2}$ must prefer some other resident to $r^2_{c_2}$; let this resident be r_y.

Thus in both cases we have that if $h^1_{c_2} \neq h^0_{c_1}$ then $h^1_{c_2}$ is of capacity one and prefers some resident r_y to $r^2_{c_2}$. Clearly, r_y is either a member of a couple or is a single resident. As before, we consider both of these cases and show that we must arrive at the same outcome in either case.

Case(i): r_y is single and thus $n_2 > 0$; In this case let $r_y = r^1_{s_2}$. Since $r^1_{s_2}$ is in first place on the preference list of $h^1_{c_2}$, it follows that $h^1_{c_2}$ cannot be in first place in the preference list of $r^1_{s_2}$. Hence, there must exist another hospital preferred by $r^1_{s_2}$; let this be $h^2_{c_2}$. Further, $h^2_{c_2}$ must be of capacity one and have a resident other than $r^1_{s_2}$ in first place in its preference list; let this resident be $r^2_{s_2}$. We consider first the case where each newly generated resident is single. Suppose $r^2_{s_2}$ is single. Since $r^2_{s_2}$ is in first place on the preference list of $h^2_{c_2}$ there must exist another hospital which is preferred by $r^2_{s_2}$; let this new hospital be $h^3_{c_2}$. Hospital $h^3_{c_2}$ must have capacity one, otherwise $(r^2_{s_2}, h^3_{c_2})$ would represent a fixed assignment.

We may continue generating a sequence of distinct single residents and hospitals of capacity one, but since the number of residents is finite, we must eventually arrive at a resident who is a member of a couple; let this resident be $r^1_{c_3}$. Without loss of generality suppose that $r^1_{c_3}$ is the first member of the couple to which he belongs. Let $r^{n_2}_{s_2}$ be the final single resident in the previously generated sequence. Then $r^{n_2}_{s_2}$ prefers some hospital $h^{n_2+1}_{c_2}$ to $h^{n_2}_{c_2}$ and $h^{n_2+1}_{c_2}$ must be of capacity one. If $h^{n_2+1}_{c_2} = h^0_{c_1}$ then I contains precisely two couples. Otherwise $h^{n_2+1}_{c_2}$ is a new hospital of capacity one and prefers $r^1_{c_3}$ to $r^{n_2}_{s_2}$.

Since $h^{n_2+1}_{c_2}$ has $r^1_{c_3}$ in first place on its preference list, it must be the case that $r^1_{c_3}$ expresses a joint preference as part of the couple $(r^1_{c_3}, r^2_{c_3})$ for a hospital pair involving $h^{n_2+1}_{c_2}$; let this pair be $(h^{n_2+1}_{c_2}, h^1_{c_3})$.

Since $h^1_{c_3}$ has $r^2_{c_3}$ in first place on its preference list, $h^2_{c_3}$ must be of capacity one and prefer some other resident to $r^2_{c_3}$; let this resident be r_z.

Case(ii): r_y is a member of a couple and thus $n_2 = 0$. Let $r_y = r^1_{c_2}$. Then $h^1_{c_2}$ prefers $r^1_{c_3}$ to $r^2_{c_2}$. Assume that $r^1_{c_3}$ is part of a couple $(r^1_{c_3}, r^2_{c_3})$ and further assume that $(r^1_{c_3}, r^2_{c_3})$ finds $(h^1_{c_2}, h^1_{c_3})$ acceptable. If $h^1_{c_3} = h^0_{c_1}$ then I contains three couples and is of the form shown in Figure 3.15 with $N = 3$ and $n_3 = 0$. (In this case $h^0_{c_1}$ prefers $r^1_{c_1}$ to $r^2_{c_3}$.) Otherwise, $h^1_{c_3}$ is a new hospital which must be of capacity one (or else $(r^1_{c_3}, r^2_{c_3})$ represents a fixed assignment with $(h^1_{c_2}, h^1_{c_3})$ and must prefer some resident to $r^2_{c_3}$; let this resident be r_z.

Now, in both cases we have that if $h^1_{c_3} \neq h^0_{c_1}$ then $h^1_{c_3}$ is of capacity one and prefers some resident r_z to $r^2_{c_3}$. As before, we may continue generating a sequence of distinct residents, couples and hospitals in this fashion, but since the number of residents and couples is finite, we must eventually reach some resident, either single or a member of a couple who must be in second place in $h^0_{c_1}$'s preference list and a complete instance of $(2,1,2)$-HRC is formed. Thus, the instance I must be of the form shown in Figure 3.15. □

Now, we have that if I is an instance of $(2,1,2)$-HRC in which all fixed assignments have been iteratively satisfied then I is the union of a finite number of disjoint discrete subinstances of $(2,1,2)$-HRC of the form shown in Figure 3.15. Let $I' \subseteq I$ be one of these arbitrary disjoint subinstances of I. Lemma 3.7.4 proves that I' admits a stable matching if and only if the number of couples involved in I' is even.

Lemma 3.7.4. *An instance I of $(2,1,2)$-HRC of the form shown in Figure 3.15 admits a stable matching if and only if the number of couples involved in I is even.*

Proof. Let M be a stable matching in I. It is either the case that $(r^1_{c_1}, r^2_{c_1})$ is assigned in M or $(r^1_{c_1}, r^2_{c_1})$ is unassigned in M. We now consider each of these cases and show that in either case if I contains an odd number of couples then I cannot admit a stable matching.

Case (i): Assume $(r^1_{c_1}, r^2_{c_1})$ is assigned in M and therefore $(r^1_{c_1}, h^0_{c_1}) \in M$. Clearly either $n_1 = 0$ or $n_1 > 0$. We now show that whether $n_1 = 0$ or $n_1 > 0$, if $(r^1_{c_1}, r^2_{c_1})$ is assigned in M then $(r^1_{c_2}, r^2_{c_2})$ is unassigned in M.

If $n_1 = 0$ and the instance contains exactly one couple, then $(r^1_{c_1}, r^2_{c_1})$ represents a fixed assignment with $(h^0_{c_1}, h^1_{c_1})$, a contradiction. Thus, I contains more than one couple. Let the second couple in I be $(r^1_{c_2}, r^2_{c_2})$ such that $h^1_{c_1}$ has $r^1_{c_2}$ in first place on its preference list. We now have that $(r^1_{c_2}, r^2_{c_2})$ expresses a preference for $(h^1_{c_1}, h^1_{c_2})$ and since $(r^2_{c_1}, h^1_{c_1}) \in M$, clearly $(r^1_{c_2}, r^2_{c_2})$ cannot be assigned in M.

If $n_1 > 0$ then $h^1_{c_1}$ has $r^1_{s_1}$ in first place on its preference list. Now, if $r^1_{s_1}$ is unassigned in M then $(r^1_{s_1}, h^1_{c_1})$ blocks M. Hence $r^1_{s_1}$ must be assigned in M and moreover $(r^1_{s_1}, h^2_{c_1}) \in M$. In similar fashion we may confirm that each $r^a_{s_1}$ $(1 \leq a < n_1)$ is assigned to the hospital $h^{a+1}_{c_1}$ in first place on its preference list.

Now consider, $r^{n_1}_{s_1}$. Again $r^{n_1}_{s_1}$ must be assigned to the hospital in first place in its preference list. If I contains exactly one couple then this hospital must be $h^0_{c_1}$ by Lemma 3.7.3. However, by assumption $(r^1_{c_1}, h^0_{c_1}) \in M$, a contradiction. Thus I must contain more than one couple. Now, let $h^{n_1+1}_{c_1}$ be the hospital in first place on $r^{n_1}_{s_1}$'s preference list. Since $(r^{n_1}_{s_1}, h^{n_1+1}_{c_1}) \in M$, clearly $(r^1_{c_2}, r^2_{c_2})$ cannot be assigned in M as the only pair they find acceptable is $(h^{n_1+1}_{c_1}, h^1_{c_2})$. Thus, we have that whether $n_1 = 0$ or $n_1 > 0$, if $(r^1_{c_1}, r^2_{c_1})$ is assigned in M then $(r^1_{c_2}, r^2_{c_2})$ is not assigned in M.

Now, either $n_2 = 0$ or $n_2 > 0$. We now show that whether $n_2 = 0$ or $n_2 > 0$, if $(r^1_{c_2}, r^2_{c_2})$ is unassigned in M then $(r^1_{c_3}, r^2_{c_3})$ must be assigned in M. If $n_2 = 0$ and the instance contains exactly two couples then $(r^1_{c_2}, r^2_{c_2})$ expresses a preference for either $(h^1_{c_1}, h^0_{c_1})$ if $n_1 = 0$ (or $(h^{n_1+1}_{c_1}, h^0_{c_1})$ if $n_1 > 0$) and $h^0_{c_1}$ has $r^2_{c_2}$ in second place on its preference list. In this case, the instance admits exactly two stable matchings of equal cardinality. If $n_2 = 0$ and the instance contains more than two couples then $(r^1_{c_3}, r^2_{c_3})$ expresses a preference for $(h^1_{c_2}, h^1_{c_3})$. Now assume, $h^1_{c_2}$ is unassigned in M. Then $(r^1_{c_2}, r^2_{c_2})$ blocks M with $(h^1_{c_1}, h^0_{c_1})$ if $n_1 = 0$ (or $(h^{n_1+1}_{c_1}, h^0_{c_1})$ if $n_1 > 0$), a contradiction. Thus we have that if $(r^1_{c_2}, r^2_{c_2})$ is not assigned in M then $(r^1_{c_3}, r^2_{c_3})$ must be assigned to $(h^1_{c_2}, h^1_{c_3})$ in M.

If $n_2 > 0$ then $h^1_{c_2}$ has $r^1_{s_2}$ in first place on its preference list. Now, if $r^1_{s_2}$ is not assigned in M then $(r^1_{s_2}, h^1_{c_2})$ blocks M, a contradiction. Hence $r^1_{s_2}$ must be assigned in M and moreover $(r^1_{s_2}, h^2_{c_2}) \in M$. In similar fashion we may confirm that each $r^a_{s_2}$ $(1 \leq a \leq n_2)$ must be assigned in M to the hospital $h^{a+1}_{s_2}$ in first place in its preference list.

Now consider, $r^{n_2}_{s_2}$. If the instance contains exactly two couples then the hospital in first place in the preference list of $r^{n_2}_{s_2}$ must be $h^0_{c_1}$ and the result follows. However, if the instance contains more than two couples then the hospital in first place in the preference list of $r^{n_2}_{s_2}$ must be a new hospital $h^{n_2+1}_{c_2}$. Now let the next couple be $(r^1_{c_3}, r^2_{c_3})$. Assume $(r^1_{c_3}, r^2_{c_3})$ is unassigned in M. Then $(r^{n_2}_{s_2}, h^{n_2+1}_{c_2})$ must block M, so $(r^1_{c_3}, r^2_{c_3})$ must be assigned to $(h^{n_2+1}_{c_2}, h^1_{c_3})$ in M. Thus, whether $n_2 = 0$ or $n_2 > 0$, if $(r^1_{c_2}, r^2_{c_2})$ is unassigned in M then $(r^1_{c_3}, r^2_{c_3})$ must be assigned in M.

In similar fashion either $n_3 = 0$ or $n_3 > 0$. Again, we show that whether $n_3 = 0$ or $n_3 > 0$, if $(r^1_{c_3}, r^2_{c_3})$ is assigned in M then $(r^1_{c_4}, r^2_{c_4})$ is not assigned in M. If $n_3 = 0$ and the instance contains exactly three couples then $(r^1_{c_3}, r^2_{c_3})$ is assigned to $(h^1_{c_2}, h^0_{c_1})$ if $n_2 = 0$ (or $(h^{n_2+1}_{c_1}, h^0_{c_1}$ if $n_2 > 0$) and $h^0_{c_1}$ has $r^2_{c_3}$ in second place on its preference list. However, by assumption $(r^2_{c_1}, h^0_{c_1}) \in M$, a contradiction. Thus, I contains more than three couples and $(r^1_{c_4}, r^2_{c_4})$ expresses a preference for $(h^1_{c_3}, h^1_{c_4})$ and since $(r^1_{c_3}, h^1_{c_3}) \in M$, $(r^1_{c_4}, r^2_{c_4})$ cannot be assigned in M.

If $n_3 > 0$ then $h^1_{c_3}$ has $r^1_{s_3}$ in first place on its preference list. Now, if $r^1_{s_3}$ is not assigned in M then $(r^1_{s_3}, h^1_{c_3})$ blocks M, a contradiction. Hence $r^1_{s_3}$ must be assigned in M and moreover $(r^1_{s_3}, h^2_{c_3}) \in M$. In similar fashion we may confirm that each $r^a_{s_3}$ $(1 \leq a < n_3)$ is assigned to the hospital $h^{a+1}_{c_3}$ in first place on its preference list.

Now consider $r^{n_3}_{s_3}$. If the instance contains exactly three couples then the hospital in first place in the preference list of $r^{n_3}_{s_3}$ must be $h^0_{c_1}$. However, by construction, $(r^2_{c_1}, h^0_{c_1}) \in M$, a contradiction. Hence, the instance must have more than three couples and the hospital in first place in the preference list of $r^{n_3}_{s_3}$ must be a new hospital $h^{n_3+1}_{c_3}$. Now let the next couple be $(r^1_{c_4}, r^2_{c_4})$. Since $(r^{n_3}_{s_3}, h^{n_3+1}_{c_3}) \in M$, $(r^1_{c_4}, r^2_{c_4})$ cannot be assigned in M. Thus, whether $n_3 = 0$ or $n_3 > 0$, if $(r^1_{c_3}, r^2_{c_3})$ is assigned in M then $(r^1_{c_4}, r^2_{c_4})$ is not assigned in M.

Finally we consider whether $n_4 = 0$ or $n_4 > 0$. If $n_4 = 0$ and the instance contains exactly four couples then $(r^1_{c_4}, r^2_{c_4})$ expresses a preference for the hospital pair $(h^1_{c_4}, h^0_{c_1})$ and $h^0_{c_1}$

has $r^2_{c_4}$ in second place on its preference list and the result follows. Otherwise the instance contains more than four couples and $(r^1_{c_5}, r^2_{c_5})$ expresses a preference for $(h^1_{c_4}, h^1_{c_5})$. Now assume, $h^1_{c_4}$ is unassigned in M. Then $(r^1_{c_4}, r^2_{c_4})$ blocks M with $(h^{n_4+1}_{c_4}, h^1_{c_4})$, a contradiction. Thus $(r^1_{c_5}, r^2_{c_5})$ must be assigned to $(h^1_{c_4}, h^1_{c_5})$ in M.

If $n_4 > 0$ then $h^1_{c_4}$ has $r^1_{s_4}$ in first place on its preference list. If $r^1_{s_4}$ is not assigned in M then $(r^1_{s_4}, h^1_{c_4})$ blocks M, a contradiction. Hence $r^1_{s_4}$ must be assigned in M and moreover $(r^1_{s_4}, h^2_{c_4}) \in M$. In similar fashion we may confirm that each $r^a_{s_4}$ $(1 \leq a \leq n_4)$ must be assigned in M to the hospital $h^{a+1}_{s_4}$ in first place in its preference list. Now consider, $r^{n_4}_{s_4}$. If the instance contains exactly four couples then the hospital in first place in the preference list of $r^{n_4}_{s_4}$ must be $h^0_{c_1}$ and the result follows.

At this point we observe that argument is similar for the case that the number of couples is larger than four. As the preceding argument shows, if the number of couples is odd, then no stable matching exists, a contradiction.

Case (ii): Now suppose that $(r^1_{c_1}, r^2_{c_1})$ is unassigned in M. Then essentially $(r^1_{c_1}, r^2_{c_1})$ plays the role of $(r^1_{c_2}, r^2_{c_2})$ in the proof above and we may continue to generate a sequence of couples, every second of which is unassigned in M. Again, the same proof above can be used to infer that if the number of couples is odd, then no stable matching can exist.

Conversely, we show that if the number of couples in I is even then I admits a stable matching. For ease of exposition we use the description of the instance I shown in Figure 3.16 for this part of the proof. For clarity, this instance is exactly equivalent to the instance shown in Figure 3.15. Let M be the following matching in I where $h^{n_N+1}_{c_N} = h^0_{c_1}$ if $n_N > 0$ and $h^1_{c_N} = h^0_{c_1}$ if $n_N = 0$:

$$\begin{aligned} M &= \{(r^1_{c_1}, h^0_{c_1}), (r^2_{c_1}, h^1_{c_1})\} \\ &\bigcup \{(r^1_{c_k}, h^{n_{k-1}+1}_{c_{k-1}}), (r^2_{c_k}, h^1_{c_k}) : 2 \leq k \leq N, n_{k-1} > 0, k \bmod 2 \neq 0\} \\ &\bigcup \{(r^1_{c_k}, h^2_{c_{k-1}}), (r^2_{c_k}, h^1_{c_k}) : 2 \leq k \leq N, n_{k-1} = 0, k \bmod 2 \neq 0\} \\ &\bigcup \{(r^a_{s_k}, h^{a+1}_{c_k}) : 1 \leq k \leq N, 1 \leq a \leq n_k, n_k > 0\} \end{aligned}$$

Assume M is unstable. Then there must exist a blocking pair of M in I.

Clearly no single resident $r^a_{s_k}$ $(1 \leq k \leq N, 1 \leq a \leq n_k, n_k > 0)$ can form part of a blocking pair for M in I as he is assigned in M to his first preference. Further, no couple $(r^1_{c_k}, r^2_{c_k})$ $(2 \leq k \leq N, k \bmod 2 \neq 0\}$ can form part of a blocking pair for M in I as they are assigned to the hospital pair in first place on their joint preference list, $(h^{n_{k-1}+1}_{c_{k-1}}, h^1_{c_k})$ if $n_k > 0$ or $(h^2_{c_{k-1}}, h^1_{c_k})$ if $n_k = 0$.

Now, assume that $(r^1_{c_k}, r^2_{c_k})$ $(2 \leq k \leq N, k \bmod 2 = 1)$ blocks M. If $n_{k-1} > 0$ then $(r^1_{c_k}, r^2_{c_k})$ blocks with $(h^{n_{k-1}+1}_{c_{k-1}}, h^1_{c_k})$. However, $h^1_{c_k}$ is assigned in M to its first preference $r^1_{s_k}$ and so cannot form part of a blocking pair, a contradiction. If $n_k = 0$ then $(r^1_{c_k}, r^2_{c_k})$ blocks M with $(h^2_{c_{k-1}}, h^1_{c_k})$. However, $h^1_{c_k}$ is assigned in M to its first preference (either $r^1_{s_k}$ if $n_k > 0$, or

$r^1_{c_{k+1}}$ if $n_k = 0$) and so cannot form part of a blocking pair, a contradiction. Since no other possible blocking pairs exist for M in I it must be the case that M is a stable matching in I and the result is proven. □

Now by Lemma 3.7.3 we have that for any instance I of $(2, 1, 2)$-HRC in which all fixed assignments have been iteratively satisfied, I is constructed from disjoint subinstances of the form shown in Figure 3.15. Further, by Lemma 3.7.4 each disjoint subinstance $I' \subseteq I$ of I admits a stable matching if and only if the number of couples in I is even. However, if every subinstance $I' \subseteq I$ contains an even number of couples and we assign the residents in I' as in the proof of Theorem 3.7.4 then we form a stable matching in I.

Since all of the stable matchings admitted by I are of the same size, a stable matching is a maximum cardinality stable matching. Hence we may find a maximum cardinality stable matching or report that none exists in an instance $(2, 1, 2)$-HRC in polynomial time. We state this result formally as Theorem 3.7.5.

Theorem 3.7.5. *Given an instance I of $(2, 1, 2)$-HRC we can find a maximum cardinality stable matching or report that none exists in polynomial time.*

Corollary 3.7.6. *An instance I of $(2, 1, 2)$-HRC of the form shown in Figure 3.15 which does not admit a stable matching can be transformed to an instance which does admit a stable matching by increasing the capacity of any of the hospitals in I by exactly one.*

Chapter 4

Integer programming model for HRC variants under MM-stability

4.1 Introduction

In this chapter we present the first IP model for finding a maximum cardinality stable matching in an arbitrary instance of HRC under MM-stability. In Section 4.2 we present and prove the correctness of an IP model for finding a maximum cardinality stable matching in instances of HR and then show in Section 4.3 how this model can be extended to the HRC context by presenting and proving the correctness of a model for finding a maximum cardinality stable matching in an arbitrary instances of HRC under MM-stability. In Section 4.4 we demonstrate by means of an example instance how an instance of the IP model is constructed from an instance of HRC. We further show in Section 4.5 how the HRC model can be extended to HRCT, the variant of HRC in which an agent might be indifferent between agents in its preference list. Finally in Section 4.6 we present and prove the correctness of an IP model for finding a maximum cardinality 'most stable' matching in an arbitrary instance of MIN BP HRC, the minimisation variant of HRC.

4.2 An IP formulation for HR

The IP models presented in this thesis are designed around a series of linear inequalities that establish the absence of blocking pairs. The variables are defined for each resident and for each element on his preference list (with the possibility of being unassigned). Recall from the definition of HR in Section 2.2 that an instance I of HR involves a set $R = \{r_1, r_2, \ldots, r_{n_1}\}$ containing residents and a set $H = \{h_1, h_2, \ldots, h_{n_2}\}$ containing hospitals. Further, each resident $r_i \in R$, has a preference list of length $l(r_i)$ consisting of individual hospitals $h_j \in H$, each hospital $h_j \in H$ has a preference list of individual residents $r_i \in R$ of length $l(h_j)$ and each hospital h_j has a capacity c_j, representing the maximum number of residents that h_j can be assigned. We describe the variables and constraints in the IP model for HR in Sections 4.2.1 and 4.2.2 respectively and in Section 4.2.3 we prove the correctness of the model.

4.2.1 Variables in the IP model for HR

Let J be the following integer programming formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

For $p = l(r_i) + 1$ define a variable $x_{i,p}$ whose intuitive meaning is that resident r_i is unassigned. Thus we also have

$$x_{i,l(r_i)+1} = \begin{cases} 1 & \text{if } r_i \text{ is unassigned} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(r_i) + 1\}$. Let $pref(r_i, p)$ denote the hospital at position p in r_i's preference list where $1 \leq i \leq n_1$ and $1 \leq p \leq l(r_i)$. Further for an acceptable resident-hospital pair (r_i, h_j), let $rank(h_j, r_i) = q$ be an integer denoting the rank that hospital h_j assigns resident r_i, for a given i, j $(1 \leq j \leq n_2, 1 \leq i \leq n_1)$. It follows that, $1 \leq q \leq l(h_j)$.

4.2.2 Constraints in the IP model for HR

The following constraint simply confirms that each variable $x_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i) + 1)$:

$$x_{i,p} \in \{0, 1\} \tag{4.1}$$

As each resident $r_i \in R$ is assigned to exactly one hospital or is unassigned, we introduce the following constraint for all i $(1 \leq i \leq n_1)$:

$$\sum_{p=1}^{l(r_i)+1} x_{i,p} = 1 \tag{4.2}$$

Since a hospital h_j may be assigned at most c_j residents, it follows that $x_{i,p} = 1$ where $pref(r_i, p) = h_j$ for at most c_j residents. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : pref(r_i, p) = h_j\} \leq c_j \tag{4.3}$$

In a stable matching M in I, if a single resident $r_i \in R$ has a worse partner than some hospital $h_j \in H$, where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$, then h_j must be fully subscribed with better partners than r_i. Hence, either r_i is assigned to h_j or a better partner, in which case $\sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = 0$ or h_j is fully subscribed with better partners than r_i and $\sum_{q'=1}^{q-1} \{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} = c_j$. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$:

$$c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} \leq \sum_{q'=1}^{q-1} \{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} \tag{4.4}$$

Objective Function - A maximum cardinality stable matching M in I is a stable matching in which the maximum number of residents are assigned taken over all of the stable matchings admitted by I. To maximise the size of the stable matching output from the model we apply the following objective function:

$$\max \sum_{i=1}^{n} \sum_{p=1}^{l(r_i)} x_{i,p} \tag{4.5}$$

4.2.3 Proof of correctness of the IP model for HR

We now establish the correctness of the IP model presented in Sections 4.2.1 and 4.2.2.

Theorem 4.2.1. *Given an instance I of HR, let J be the corresponding IP model as defined in Section 4.2.1 and Section 4.2.2. A stable matching in I is exactly equivalent to a feasible solution to J.*

Proof. Consider a stable matching M in I. From M we form an assignment of values to the variables $\mathbf{x}$ as follows. Initially $x_{i,p} = 0$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i) + 1)$. Then for each $(r_i, h_j) \in M$, $x_{i,p} = 1$, where $h_j = pref(r_i, p)$. If r_i is unassigned, then $x_{i,l(r_i)+1} = 1$. As each resident is either assigned or unassigned (but not both), for a given i $(1 \leq i \leq n_1)$, it follows that $x_{i,p} = 1$ for exactly one value of p in the range $1 \leq p \leq$

$r(i)+1$, and for each other value of p in the same range, $x_{i,p} = 0$. Hence, Constraint 4.2 holds in the assignment derived from M. Further, since each hospital has at most c_j assignees in M, Constraint 4.3 also holds in the assignment derived from M.

Let (r_i, h_j) be an acceptable pair not in M where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$. If (r_i, h_j) blocks M in I it follows that r_i is either unassigned or has a partner worse than rank p while simultaneously h_j is either undersubscribed or has an assignee worse than rank q. Now, if r_i has a partner worse than h_j, then $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = c_j$. Otherwise, $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = 0$. Further, if h_j has c_j partners better than r_i, then $\sum_{q'=1}^{q-1}\{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} = c_j$. Otherwise, $\sum_{q'=1}^{q-1}\{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} < c_j$.

Now, suppose that $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = c_j$. Then r_i is unassigned or has a worse partner than h_j in M. Thus, by the stability of M, h_j is full and prefers all of its assignees to r_i. Hence $\sum_{q'=1}^{q-1}\{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} = c_j$ and Constraint 4.4 is satisfied by the assignment derived from M.

Now, suppose $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = 0$. Then r_i is assigned to a better partner than h_j in M. Further, since $\sum_{q'=1}^{q-1}\{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} \geq 0$, Constraint 4.4 is trivially satisfied by the assignment derived from M. Thus, all of the constraints in J hold for an assignment of values to the variables in $\langle \mathbf{x} \rangle$ derived from a stable matching M. Hence, a stable matching M in I represents a feasible solution to J.

Conversely, consider a feasible solution $\langle \mathbf{x} \rangle$ to J. From such a solution we form a set of pairs, M, as follows. Initially let $M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ if $x_{i,p} = 1$, then add (r_i, h_j) to M where $h_j = pref(r_i, p)$. As $\langle \mathbf{x} \rangle$ satisfies Constraints 4.1, 4.2 and 4.3, each resident in M has exactly one partner or is unassigned (but not both) and each hospital h_j in M has at most c_j assignees. Hence the set of pairs M created from $\langle \mathbf{x} \rangle$ is a matching in I.

It remains to show that M is stable. Assume a mutually acceptable resident hospital pair (r_i, h_j) blocks M where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$. Thus, r_i has a worse partner than h_j or is unassigned and h_j is undersubscribed or prefers r_i to one of its assignees. Now we have that in $\langle \mathbf{x} \rangle$, $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = c_j$ and $\sum_{q'=1}^{q-1}\{x_{i',p''} : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j)\} < c_j$ and thus Constraint 4.4 is not satisfied in $\langle \mathbf{x} \rangle$, a contradiction to the feasibility of $\langle \mathbf{x} \rangle$. Hence no such (r_i, h_j) can block M. Thus a matching M in I derived from a feasible solution $\langle \mathbf{x} \rangle$ for J is stable and the theorem is proven. □

The existence of the objective function (given by Inequality 4.5) immediately leads to the following corollary.

Corollary 4.2.2. *Given an instance I of* HR*, let J be the corresponding IP model as defined in Section 4.2.1 and Section 4.2.2. A maximum cardinality stable matching in I is exactly* *equivalent to an optimal solution to J.*

4.3 An IP formulation for HRC

The IP model presented in this section extends the model for HR presented in Section 4.2. This extended model is designed around a series of linear inequalities that establish the absence of blocking pairs according to each of the different parts of Definition 2.3.1. The variables are defined for each resident, whether single or a member of a couple, and for each element on his preference list (with the possibility of being unassigned). A further consistency constraint ensures that each member of a couple obtains hospitals from the same pair in their list, if assigned. Finally, the objective of the IP is to maximise the size of a stable matching, if one exists. The model presented is more complex than existing IP formulations in the literature for stable matching problems [78, 68, 62, 47] simply because of the number of blocking pair cases in Definition 2.3.1 required to adequately take account of couples.

We now define an instance of HRC and show how the *projected preference lists* for each of the two residents involved in a couple may be derived from the couple;s joint preference lists. Let I be an instance of HRC with residents $R = \{r_1, r_2, \ldots, r_{n_1}\}$ and hospitals $H = \{h_1, h_2, \ldots, h_{n_2}\}$. Without loss of generality, suppose residents $r_1, r_2 \ldots r_{2c}$ are in couples. Again, without loss of generality, suppose that the couples are (r_{2i-1}, r_{2i}) $(1 \leq i \leq c)$. Suppose that the joint preference list of a couple $c_i = (r_{2i-1}, r_{2i})$ is:

$$c_i \ : \ (h_{\alpha_1}, h_{\beta_1}), (h_{\alpha_2}, h_{\beta_2}) \ldots (h_{\alpha_l}, h_{\beta_l})$$

From this list we create the following *projected preference list* for resident r_{2i-1}:

$$r_{2i-1} \ : \ h_{\alpha_1}, h_{\alpha_2} \ldots h_{\alpha_l}$$

and the following projected preference list for resident r_{2i}:

$$r_{2i} \ : \ h_{\beta_1}, h_{\beta_2} \ldots h_{\beta_l}$$

Clearly, the projected preference list of the residents r_{2i-1} and r_{2i} are the same length as the preference list of the couple $c_i = (r_{2i-1}, r_{2i})$. Let $l(c_i)$ denote the length of the preference list of c_i and let $l(r_{2i-1})$ and $l(r_{2i})$ denote the lengths of the projected preference lists of r_{2i-1} and r_{2i} respectively. Now we have that $l(r_{2i-1}) = l(r_{2i}) = l(c_i)$. A given hospital h_j may appear more than once in the projected preference list of a linked resident in a couple $c_i = (r_{2i-1}, r_{2i})$.

Let the single residents be $r_{2c+1}, r_{2c+2} \ldots r_{n_1}$, where each single resident r_i, has a preference list of length $l(r_i)$ consisting of individual hospitals $h_j \in H$. Each hospital $h_j \in H$ has a preference list of individual residents $r_i \in R$ of length $l(h_j)$. Further, each hospital $h_j \in H$ has capacity $c_j \geq 1$, the maximum number of residents to which it may be assigned.

When considering the exact nature of a blocking pair in this model, the stability definition due to Manlove and McDermid [55] (MM-stability) is applied in all cases. The text in bold before the definition of a constraint shows the section of the MM-stability definition with which the constraint corresponds. Hence, a constraint preceded by '**Stability 1**' is intended to prevent blocking pairs described by part 1 of the MM-stability definition shown in Definition 2.3.1 in Section 2.3.

We describe the variables and constraints in the IP model for HRC under MM-stability in Sections 4.3.1 and 4.3.2 respectively and in Section 4.3.3 we prove the correctness of the model.

4.3.1 Variables in the IP model for HRC

Let J be the following IP formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

For $p = l(r_i) + 1$ define a variable $x_{i,p}$ whose intuitive meaning is that resident r_i is unassigned. Thus we also have that

$$x_{i,l(r_i)+1} = \begin{cases} 1 & \text{if } r_i \text{ is unassigned} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1, 1 \leq p \leq l(r_i) + 1\}$. Let $\mathit{pref}(r_i, p)$ denote the hospital at position p of a single resident r_i's preference list or on the projected preference list of a resident belonging to a couple for a given $1 \leq i \leq n_1$ and $1 \leq p \leq l(r_i)$. Let $\mathit{pref}((r_{2i}, r_{2i-1}), p)$ denote the hospital pair at position p on the joint preference list of (r_{2i-1}, r_{2i}).

For ease of exposition we define some additional notation. For each j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ let the set $R(h_j, q)$ contain the resident-position pairs (r_i, p) such that r_i is assigned a rank of q $(1 \leq q \leq l(h_j))$ by h_j and h_j is in position p $(1 \leq p \leq l(r_i))$ on r_i's projected preference list. Hence:

$$R(h_j, q) = \{(r_i, p) \in R \times \mathbb{Z} : \mathit{rank}(h_j, r_i) = q \wedge 1 \leq p \leq l(r_i) \wedge \mathit{pref}(r_i, p) = h_j\}$$

Now, for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new variable $\alpha_{j,q} \in \{0, 1\}$. The intuitive meaning of a variable $\alpha_{j,q}$ is that if h_j is fully subscribed with assignees better than rank q, then $\alpha_{j,q}$ may take the value 0 or 1. However, if h_j is not full with assignees better than rank q, then $\alpha_{j,q} = 1$. Constraints 4.7 and 4.17 described in Section 4.3.2 are applied to enforce this property.

Now, for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new variable $\beta_{j,q} \in \{0, 1\}$. The intuitive meaning of a variable $\beta_{j,q}$ is that if h_j has $c_j - 1$ or more assignees better than rank q, then $\beta_{j,q}$ may take a value of zero or one. However, if h_j has fewer than $c_j - 1$ assignees better than rank q, then $\beta_{j,q} = 1$. Constraints 4.8 and 4.18 described in Section 4.3.2 are applied to enforce this property.

4.3.2 Constraints in the IP model for HRC

The following constraint simply confirms that each variable $x_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i) + 1)$:

$$x_{i,p} \in \{0, 1\} \tag{4.6}$$

Similarly, the following constraint confirms that each variable $\alpha_{j,q}$ must be binary valued for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\alpha_{j,q} \in \{0, 1\} \tag{4.7}$$

Also, the following constraint confirms that each variable $\beta_{j,q}$ must be binary valued for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\beta_{j,q} \in \{0, 1\} \tag{4.8}$$

As each resident $r_i \in R$ is assigned to exactly one hospital or is unassigned (but not both), we introduce the following constraint for all i $(1 \leq i \leq n_1)$:

$$\sum_{p=1}^{l(r_i)+1} x_{i,p} = 1 \tag{4.9}$$

Since a hospital h_j may be assigned at most c_j residents, $x_{i,p} = 1$ where $\mathit{pref}(r_i, p) = h_j$ for at most c_j residents. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : \mathit{pref}(r_i, p) = h_j\} \leq c_j \tag{4.10}$$

For each couple (r_{2i-1}, r_{2i}), if resident r_{2i-1} is assigned to the hospital in position p in their projected preference list, then r_{2i} must also be assigned to the hospital in position p in their projected preference list. We thus obtain the following constraint for all i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}) + 1)$:

$$x_{2i-1,p} = x_{2i,p} \tag{4.11}$$

Stability 1 - In a stable matching M in I, if a single resident $r_i \in R$ has a worse partner than some hospital $h_j \in H$ where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$, then h_j must be fully subscribed with better partners than r_i. Hence, either r_i is assigned to h_j or a better partner and thus $\sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = 0$ or h_j is fully subscribed with better partners than r_i and $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} = c_j$. Thus, for each i $(2c + 1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$:

$$c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} \leq \sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \tag{4.12}$$

Stability 2(a) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ prefer hospital pair (h_{j_1}, h_{j_2}), at position p_1 in c_i's joint preference list, to $(M(r_{2i-1}), M(r_{2i}))$, at position p_2, then, if $h_{j_2} = M(r_{2i})$, h_{j_1} cannot be undersubscribed or prefer r_{2i-1} to one of its assignees in M. In the special case where $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_1}$, if $h_{j_1} = h_{j_2} = M(r_{2i})$, then h_{j_1} cannot be undersubscribed or prefer r_{2i-1} to one of its assignees in M other than r_{2i}. Thus, for the general case, we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ such that $pref(r_{2i}, p_1) = pref(r_{2i}, p_2)$ and $rank(h_{j_1}, r_{2i-1}) = q$:

$$c_{j_1} x_{2i,p_2} \leq \sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} \tag{4.13}$$

For the special case in which $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_1}$ we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ such that $pref(r_{2i}, p_1) = pref(r_{2i}, p_2)$ and $rank(h_{j_1}, r_{2i-1}) = q$:

$$(c_{j_1} - 1)x_{2i,p_2} \leq \sum_{q'=1}^{q-1}\{x_{i',p''} \in X : q' \neq rank(h_{j_1}, r_{2i}) \wedge (r_{i'}, p'') \in R(h_{j_1}, q')\} \tag{4.14}$$

Stability 2(b) - A similar constraint is required for the second resident in each couple. Thus, for the general case, we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 where

$(1 \leq p_1 < p_2 \leq l(r_{2i}))$ such that $\mathit{pref}(r_{2i-1}, p_1) = \mathit{pref}(r_{2i-1}, p_2)$ and $\mathit{rank}(h_{j_2}, r_{2i}) = q$:

$$c_{j_2} x_{2i-1,p_2} \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_2}, q')\} \tag{4.15}$$

Again, for the special case in which $\mathit{pref}(r_{2i-1}, p_1) = \mathit{pref}(r_{2i}, p_1) = h_{j_2}$ we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i}))$ such that $\mathit{pref}(r_{2i-1}, p_1) = \mathit{pref}(r_{2i-1}, p_2)$ and $\mathit{rank}(h_{j_2}, r_{2i}) = q$:

$$(c_{j_1} - 1) x_{2i-1,p_2} \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : q' \neq \mathit{rank}(h_{j_2}, r_{2i-1}) \wedge (r_{i'}, p'') \in R(h_{j_2}, q')\} \tag{4.16}$$

Now, we define a variable $\alpha_{j,q}$ such that if h_j is full with assignees better than rank q, then $\alpha_{j,q}$ may take the value of zero or one. Otherwise, h_j is not full with assignees better than rank q and $\alpha_{j,q} = 1$. Hence, we obtain the following constraint for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\alpha_{j,q} \geq 1 - \frac{\sum_{q'=1}^{q-1} \{x_{i,p} \in X : (r_i, p) \in R(h_j, q')\}}{c_j} \tag{4.17}$$

Next we define a variable $\beta_{j,q}$ such that if h_j has $c_j - 1$ or more assignees better than rank q, then $\beta_{j,q}$ may take a value of zero or one. Otherwise, h_j has fewer than $c_j - 1$ assignees better than rank q and $\beta_{j,q} = 1$. Hence, we obtain the following constraint all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\beta_{j,q} \geq 1 - \frac{\sum_{q'=1}^{q-1} \{x_{i,p} \in X : (r_i, p) \in R(h_j, q')\}}{(c_j - 1)} \tag{4.18}$$

Stability 3(a) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse hospital pair than (h_{j_1}, h_{j_2}) (where $h_{j_1} \neq h_{j_2}$) it must be the case that for some $t \in \{1, 2\}$, h_{j_t} is full or prefers its worst assignee to r_{2i-2+t}. Thus we obtain the following constraint for all i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ where $h_{j_1} = \mathit{pref}(r_{2i-1}, p)$, $h_{j_2} = \mathit{pref}(r_{2i}, p)$, $h_{j_1} \neq h_{j_2}$, $\mathit{rank}(h_{j_1}, r_{2i-1}) = q_1$ and $\mathit{rank}(h_{j_2}, r_{2i}) = q_2$:

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \alpha_{j_1,q_1} + \alpha_{j_2,q_2} \leq 2 \tag{4.19}$$

Stability 3(b) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$, then h_j must not have two or more free posts available.

Stability 3(c) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$, then h_j must not prefer at least one of r_{2i-1} or r_{2i} to some assignee of h_j in M while simultaneously having a single free post.

Both of the preceding stability definitions may be modelled by a single constraint. Thus, we obtain the following constraint for i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ such that $pref(r_{2i-1}, p) = pref(r_{2i}, p)$ and $h_j = pref(r_{2i-1}, p)$ where $q = \min\{rank(h_j, r_{2i}), rank(h_j, r_{2i-1})\}$:

$$c_j \sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} - \frac{\sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\}}{(c_j - 1)}$$

$$\leq \sum_{q'=1}^{l(h_j)} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \tag{4.20}$$

Stability 3(d) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$, then h_j must not be fully subscribed and also have two assigned partners r_s and r_t (where $s \neq t$) such that h_j strictly prefers r_{2i-1} to r_s and also prefers r_{2i} to r_t. For each (h_j, h_j) acceptable to (r_{2i-1}, r_{2i}), let r_{min} be the better of r_{2i-1} and r_{2i} according to hospital h_j with $rank(h_j, r_{min}) = q_{min}$. Analogously, let r_{max} be the worse of r_{2i} and r_{2i-1} according to hospital h_j with $rank(h_j, r_{max}) = q_{max}$. Thus we obtain the following constraint for i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ such that $pref(r_{2i-1}, p) = pref(r_{2i}, p) = h_j$.

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \alpha_{j,q_{max}} + \beta_{j,q_{min}} \leq 2 \tag{4.21}$$

Objective Function - A maximum cardinality stable matching M in I is a stable matching in which the maximum number of residents are assigned taken over all of the stable matchings admitted by I. Thus, to maximise the size of the stable matching found we apply the following objective function:

$$\max \sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} x_{i,p} \tag{4.22}$$

4.3.3 Proof of correctness of constraints in the IP model for HRC

We now establish the correctness of the IP model presented in Sections 4.3.1 and 4.3.2.

Theorem 4.3.1. *Given an instance I of HRC, let J be the corresponding IP model as defined in Section 4.3.1 and Section 4.3.2. A stable matching in I is exactly equivalent to a feasible solution to J.*

Proof. Consider a stable matching M in I. We construct an assignment of values $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ to the variables $\mathbf{x}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$ as follows. Initially $x_{i,p} = 0$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i)+1)$. For each $(r_i, h_j) \in M$ where r_i is a single resident, $x_{i,p} = 1$, where $pref(r_i, p) = h_j$. If r_i is unassigned, then $x_{i,l(r_i)+1} = 1$. If some $r_k \in R$ is a member of a couple, assume without loss of generality that $r_k = r_{2i-1}$ (respectively r_{2i}) for some i $(2c+1 \leq i \leq n_1)$ then $x_{2i-1,p} = 1$ (respectively $x_{2i,p} = 1$) where $pref((r_{2i-1}, r_{2i}), p) = (h_{j_1}, h_{j_2})$ where $h_{j_1} = M(r_{2i-1})$ and $h_{j_2} = M(r_{2i})$. If (r_{2i-1}, r_{2i}) is unassigned, then $x_{2i-1,l(r_{2i-1})+1} = 1$ and $x_{2i,l(r_{2i})+1} = 1$.

For each $\alpha_{j,q}$ where j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$, if h_j is full with assignees better than rank q, then $\alpha_{j,q} = 1$. Otherwise $\alpha_{j,q} = 0$. For each $\beta_{j,q}$ where j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$, if h_j has fewer than $c_j - 1$ assignees better than rank q, then $\beta_{j,q} = 1$. Otherwise $\beta_{j,q} = 0$.

We now show that $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ satisfies each of the constraints in the model. As each resident is assigned or unassigned (but not both) for a given i $(1 \leq i \leq n_1)$, for exactly one value of p in the range $1 \leq p \leq l(r_i)+1$, $x_{i,p} = 1$, and for all p' $(1 \leq p' \leq l(r_i)+1, p' \neq p)$, $x_{i,p'} = 0$, and thus Constraint 4.9 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$. Since, each hospital h_j is assigned to at most c_j acceptable residents in M, Constraint 4.10 is also satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$.

For each couple (r_{2i-1}, r_{2i}) in I, let p $(1 \leq p \leq l(r_{2i-1}))$ be given. If r_{2i-1} is assigned to $h_{j_1} = pref(r_{2i-1}, p)$ in M, then r_{2i} is assigned to $h_{j_2} = pref(r_{2i}, p)$ in M. Similarly, for each couple (r_{2i-1}, r_{2i}) in I, if r_{2i-1} is not assigned to $h_{j_1} = pref(r_{2i-1}, p)$ in M, then r_{2i} is not assigned to $h_{j_2} = pref(r_{2i}, p)$ in M. Thus, in the assignment derived from M, $x_{2i-1,p} = x_{2i,p}$ for all i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1})+1)$ (where $l(r_{2i-1}) = l(r_{2i})$) and Constraint 4.11 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$.

Assume $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ does not satisfy Constraint 4.12. Let i $(2c+1 \leq i \leq n_1)$, j $(1 \leq j \leq n_2)$ and p $(1 \leq p \leq l(r_{2i-1}))$ be given such that (r_i, h_j) is an acceptable pair not in M where $h_j = pref(r_i, p)$ and $rank(h_j, r_i) = q$. If $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = 0$, then Constraint 4.12 is trivially satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ since $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \geq 0$. Hence, $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = c_j$ and it follows that r_i must be unassigned or have a partner worse than h_j.

Now, if $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \geq c_j$, then Constraint 4.12 is satisfied, thus it follows that $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j$ and further h_j is either undersubscribed or has an assignee worse than r_i. Thus (r_i, h_j) blocks M, a contradiction. Hence, Constraint 4.12 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$.

Assume $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$ does not satisfy Constraint 4.13. For all x_{2i,p_2} such that i $(1 \leq i \leq c)$, p_1, p_2 $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ where $h_{j_1} = \mathit{pref}(r_{2i-1}, p_1)$, $\mathit{pref}(r_{2i}, p_1) = \mathit{pref}(r_{2i}, p_2) = h_{j_2}$ and $\mathit{rank}(h_{j_1}, r_{2i-1}) = q$. Now, if $c_{j_1} x_{2i,p_2} = 0$, then Constraint 4.13 is trivially satisfied. It follows that $c_{j_1} x_{2i,p_2} = c_{j_1}$. Now, if $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} \geq c_{j_1}$, then Constraint 4.13 is satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$. Thus, it follows that $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} < c_{j_1}$.

Now since $c_{j_1} x_{2i,p_2} = c_{j_1}$ in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, (r_{2i-1}, r_{2i}) is assigned to a worse partner than (h_{j_1}, h_{j_2}). Further, since $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} < c_{j_1}$ in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, h_{j_1} is either undersubscribed in M or prefers r_{2i-1} to some member of $M(h_{j_1})$ and thus (r_{2i-1}, r_{2i}) blocks M with (h_{j_1}, h_{j_2}), a contradiction. Hence Constraint 4.13 holds in the assignment derived from M.

Assume $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$ does not satisfy Constraint 4.14. For all x_{2i,p_2} such that i $(1 \leq i \leq c)$ and p_1, p_2 $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$, where $h_j = \mathit{pref}(r_{2i-1,p_1}) = \mathit{pref}(r_{2i,p_1})$, $\mathit{pref}(r_{2i}, p_1) = \mathit{pref}(r_{2i}, p_2) = h_j$ and $\mathit{rank}(h_j, r_{2i-1}) = q$. Now, if $(c_j - 1)x_{2i,p_2} = 0$, then Constraint 4.14 is trivially satisfied. Hence, $(c_j - 1)x_{2i,p_2} = c_j - 1$. Now, if $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \geq c_j - 1$, then Constraint 4.14 is satisfied. Thus, it follows that $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j - 1$.

Now, since $(c_j - 1)x_{2i,p_2} = c_j - 1$ in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, (r_{2i-1}, r_{2i}) is assigned to a worse partner than (h_j, h_j). Further, since $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < (c_j - 1)$ in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, h_j is either undersubscribed in M or prefers r_{2i-1} to some assignee in $M(h_j)$ other than r_{2i}. Thus, (r_{2i-1}, r_{2i}) blocks M with (h_j, h_j), a contradiction. Hence Constraint 4.14 holds in the assignment derived from M.

A similar argument for the second member of each couples in M ensures that Constraints 4.15 and 4.16 are also satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$.

For each $h_j \in H$, either h_j has c_j assignees better than rank q or it does not. We now show that Constraint 4.17 is satisfied in either case. If h_j has fewer than c_j assignees better than rank q, then $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j$ and it follows that

$$\frac{\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\}}{c_j} < 1.$$

Thus $\alpha_{j,q} = 1$ and Constraint 4.17 is satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$. Otherwise h_j has c_j assignees better than rank q and $\alpha_{j,q} \geq 0$ and Constraint 4.17 is satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$.

Similarly, for each $h_j \in H$, either h_j has $c_j - 1$ assignees better than rank q or it does not. We now show that Constraint 4.18 is satisfied in either case. If h_j has fewer than $c_j - 1$ assignees better than rank q, then $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j - 1$ and it follows that

$$\frac{\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'},p'') \in R(h_j,q')\}}{(c_j-1)} < 1.$$

Thus $\beta_{j,q} = 1$ and Constraint 4.18 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$. Otherwise h_j has $c_j - 1$ assignees better than rank q and $\beta_{j,q} \geq 0$ and Constraint 4.18 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$.

Assume $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ does not satisfy Constraint 4.19. For all i ($1 \leq i \leq c$) and p ($1 \leq p \leq r_{2i-1}$), where $h_{j_1} = \mathit{pref}(r_{2i-1}, p)$, $h_{j_2} = \mathit{pref}(r_{2i}, p)$, $h_{j_1} \neq h_{j_2}$, $\mathit{rank}(h_{j_1}, r_{2i-1}) = q_1$ and $\mathit{rank}(h_{j_2}, r_{2i}) = q_2$. Now, if $\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 0$, then Constraint 4.19 must be satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$. It follows that $\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 1$. Now, if $\alpha_{j_1,q_1} = 0$ (similarly $\alpha_{j_2,q_2} = 0$), then Constraint 4.19 must be satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$. Hence $\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 1$, $\alpha_{j_1,q_1} = 1$ and $\alpha_{j_2,q_2} = 1$.

Since $\alpha_{j_1,q_1} = 1$, $\sum_{q'=1}^{q_1-1}\{x_{i',p''} \in X : (r_{i'},p'') \in R(h_{j_1},q')\} < c_j$. Thus h_{j_1} is undersubscribed in M or prefers r_{2i-1} to some assignee in $M(h_{j_1})$. Similarly, if $\alpha_{j_2,q_2} = 1$, then h_{j_2} is undersubscribed in M or prefers r_{2i} to some assignee in $M(h_{j_2})$. Also in M, (r_{2i-1}, r_{2i}) is unassigned or is assigned to a worse partner than (h_{j_1}, h_{j_2}). Thus, (r_{2i-1}, r_{2i}) blocks M with (h_{j_1}, h_{j_2}), a contradiction. Thus Constraint 4.19 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$.

Let

$$s = c_j \sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} - \frac{\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'},p'') \in R(h_j,q')\}}{(c_j-1)}$$

Further, let

$$t = \sum_{q'=1}^{l(h_j)}\{x_{i',p''} \in X : (r_{i'},p'') \in R(h_j,q')\}$$

and let

$$u = \sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'},p'') \in R(h_j,q')\}$$

Assume $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ does not satisfy Constraint 4.3.2. Hence, $s > t$. If $c_j \sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 0$, then $s \leq 0$. However, $t \geq 0$, a contradiction. Hence $c_j \sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = c_j$.

Clearly, $0 \leq u \leq c_j$. Assume $u = c_j$. Hence, $s = c_j - (c_j/(c_j-1)) = c_j(c_j-2)/(c_j-1)$. A simple argument shows that $c_j - 2 < s < c_j - 1$. Thus $t \leq c_j - 2$. It follows that h_j has two free posts in M and (r_{2i-1}, r_{2i}) is unassigned or is assigned to a worse partner than (h_j, h_j). Thus, (r_{2i-1}, r_{2i}) blocks M with (h_j, h_j) , a contradiction. Now, assume $u = c_j - 1$. Hence $s = c_j - 1$. Thus, $t \leq c_j - 2$. Again, it follows that h_j has two vacant posts in M and (r_{2i-1}, r_{2i}) is unassigned or is assigned to a worse partner than (h_j, h_j). Thus, (r_{2i-1}, r_{2i}) blocks M with (h_j, h_j), a contradiction.

Hence, $u < c_j - 1$ and it follows that $c_j - 1 < s \leq c_j$. Thus, $t \leq c_j - 1$. It follows that h_j has a vacant post in M, moreover, h_j prefers r_{2i-1} or r_{2i} to at least one of its assignees and (r_{2i-1}, r_{2i}) is unassigned or is assigned to a worse partner than (h_j, h_j). Hence, (r_{2i-1}, r_{2i}) blocks M with (h_j, h_j), a contradiction. Hence Constraint 4.3.2 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$.

Assume $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ does not satisfy Constraint 4.21. For some i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ where $pref(r_{2i-1}, p) = pref(r_{2i}, p) = h_j$, let r_{min} be the better of r_{2i} and r_{2i-1} according to hospital h_j with $rank(h_j, r_{min}) = q_{min}$. Analogously, let r_{max} be the worse of r_{2i} and r_{2i-1} according to hospital h_j with $rank(h_j, r_{max}) = q_{max}$.

Hence $\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 1$, $\alpha_{j,q_{max}} = 1$ and $\beta_{j,q_{min}} = 1$. Since $\alpha_{j,q_{max}} = 1$,

$$\sum_{q'=1}^{q_{max}-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j$$

and it follows that h_j is undersubscribed in M or prefers r_{2i-1} to some assignee, r_x, in $M(h_j)$. Similarly, if $\beta_{j,q_{min}} = 1$, then

$$\sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j - 1$$

and it follows that h_j is undersubscribed in M or prefers r_{2i} to some assignee, r_y, in $M(h_j)$.

Thus (r_{2i-1}, r_{2i}) is assigned in M to a worse partner than (h_j, h_j). Further, h_j prefers r_{2i-1} to some $r_s \in M(h_j)$ and also prefers r_{2i} to some $r_t \in M(h_j)$ where $s \neq t$. Moreover, in M, (r_{2i-1}, r_{2i}) is unassigned or is assigned to a worse partner than (h_j, h_j). Thus, (r_{2i-1}, r_{2i}) blocks M with (h_j, h_j), a contradiction. Thus Constraint 4.21 is satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$. As all of the constraints in J are satisfied by an assignment derived from a stable matching M, a stable matching M in I is equivalent to a feasible solution to J.

Conversely, consider a feasible solution, $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$, to J. From such a solution we form a set of pairs, M, as follows. Initially let $M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, if $x_{i,p} = 1$, then add (r_i, h_j) to M where $h_j = pref(r_i, p)$. As $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ satisfies Constraints 4.6, 4.9 and 4.10, each resident in M is assigned to exactly one hospital or is unassigned (but not both) and each hospital in M must have at most c_j assignees.

As $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ satisfies Constraint 4.11 each couple (r_{2i-1}, r_{2i}) is either assigned to a hospital pair (h_{j_1}, h_{j_2}), where $pref((r_{2i-1}, r_{2i}), p) = (h_{j_1}, h_{j_2})$ for some p $(1 \leq p \leq l(r_{2i-1}))$, and thus both $(r_{2i-1}, h_j) \in M$ and $(r_{2i}, h_{j_2}) \in M$, or is unassigned and thus both r_{2i-1} and r_{2i} are unassigned in M. Thus the set of pairs M created from the solution $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta} \,\rangle$ to J, is a matching in I. It remains to show that M is stable.

Type 1 Blocking Pair - Assume (r_i, h_j) blocks M as a Type 1 blocking pair, where r_i is a single resident. Let $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$. Thus, r_i is unassigned or, has a worse partner than h_j and h_j is undersubscribed or prefers r_i to some member of $M(h_j)$. It follows that in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, $c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = c_j$ and $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : rank(h_j, r_{i'}) = q' \wedge pref(r_{i'}, p'') = h_j\} < c_j$ and hence Constraint 4.12 is not satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, a contradiction. Thus no such (r_i, h_j) can block M as a Type 1 blocking pair.

Type 2 Blocking Pair - Assume (r_{2i-1}, r_{2i}) blocks M as a Type 2 blocking pair with (h_{j_1}, h_{j_2}) where $pref((r_{2i-1}, r_{2i}), p_1) = (h_{j_1}, h_{j_2})$, $pref((r_{2i-1}, r_{2i}), p_2) = (M(r_{2i-1}), M(r_{2i}))$, $1 \leq p_1 < p_2 \leq l(r_{2i-1})$, $pref(r_{2i}, p_1) = pref(r_{2i}, p_2) = h_{j_2}$ and $rank(h_{j_1}, r_{2i-1}) = q$. Hence, r_{2i} has the same hospital in positions p_1 and p_2, and h_{j_1} is undersubscribed or prefers r_{2i-1} to some member of $M(h_{j_1})$.

Further assume $pref(r_{2i-1}, p_1) \neq pref(r_{2i}, p_1)$. Hence $c_{j_1} x_{2i,p_2} = c_{j_1}$ and

$$\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} < c_{j_1}$$

as h_{j_1} is undersubscribed or prefers r_{2i-1} to some member of $M(h_{j_1})$. Hence in J, the RHS of Constraint 4.13 is at most $(c_{j_1} - 1)$ and the LHS is equal to c_{j_1} and thus Constraint 4.13 is not satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, a contradiction. Hence no such $((r_{2i-1}, r_{2i}), (h_{j_1}, h_{j_2}))$ can block M.

Thus, $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1)$. Hence $(c_{j_1} - 1)x_{2i,p_2} = (c_{j_1} - 1)$ and moreover $\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : q' \neq rank(h_{j_1}, r_{2i}) \wedge (r_{i'}, p'') \in R(h_{j_1}, q')\} < c_{j_1} - 1$ as h_{j_1} is undersubscribed or prefers r_{2i-1} to some member of $M(h_{j_1})$ other than r_{2i}. Hence in J, the RHS of Constraint 4.14 is at most $c_{j_1} - 2$ and the LHS is equal to $c_{j_1} - 1$ and thus Constraint 4.14 is not satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, a contradiction. Hence no such $((r_{2i-1}, r_{2i}), (h_{j_1}, h_{j_2}))$ can block M.

A similar argument can be used to show that the second member of each couple cannot improve in such a blocking pair in M and thus Constraint 4.15 and 4.16 are both satisfied in the assignment derived from M.

Type 3 Blocking Pairs - Suppose that (r_{2i-1}, r_{2i}) blocks M as a Type 3 blocking pair with (h_{j_1}, h_{j_2}) where $pref((r_{2i-1}, r_{2i}), (h_{j_1}, h_{j_2})) = p$, $rank(h_{j_1}, r_{2i-1}) = q_1$ and $rank(h_{j_2}, r_{2i}) = q_2$. Hence, (r_{2i-1}, r_{2i}) is unassigned or prefers (h_{j_1}, h_{j_2}) to $(M(r_{2i-1}), M(r_{2i}))$ where $h_{j_1} \neq M(r_{2i-1})$ and $h_{j_2} \neq M(r_{2i})$.

Type 3(a) Blocking Pair - Assume $h_{j_1} \neq h_{j_2}$. Hence, h_{j_1} is undersubscribed or prefers r_{2i-1} to some member of $M(h_{j_1})$ and h_{j_2} is also undersubscribed or prefers r_{2i} to some member of $M(h_{j_2})$. However, this implies that both,

$$\sum_{q'=1}^{q_1-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} < c_{j_1}$$

and

$$\sum_{q'=1}^{q_2-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_2}, q')\} < c_{j_2}$$

in J. Hence $\alpha_{j_1,q_1} = 1$, $\alpha_{j_2,q_2} = 1$ and

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 1$$

and thus Constraint 4.19 is not satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, a contradiction. Thus no such $((r_{2i-1}, r_{2i}), (h_{j_1}, h_{j_2}))$ can block M.

Type 3(b) Blocking Pair - Assume $h_{j_1} = h_{j_2} = h_j$ and h_j has two unassigned posts in M. It follows that in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$,

$$\sum_{q'=1}^{l(h_j)}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \leq c_j - 2$$

and further

$$\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \leq c_j - 2$$

Further,

$$c_j \sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = c_j$$

since (r_{2i-1}, r_{2i}) prefer (h_j, h_j) to $(M(r_{2i-1}, M(r_{2i})$. Hence in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, the RHS of Constraint 4.3.2 is at most $c_j - 2$ and the LHS is greater than $c_j - 1$ and thus Constraint 4.3.2 is not satisfied in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, a contradiction. Thus no such $((r_{2i-1}, r_{2i}), (h_j, h_j))$ can block M.

Type 3(c) Blocking Pair - Assume $h_{j_1} = h_{j_2} = h_j$ and h_j has a vacant post in M and moreover h_j prefers either r_{2i-1} or r_{2i} to some member of $M(h_j)$. Let $q = \min\{rank(h_j, r_{2i-1}),$ $rank(h_j, r_{2i})\}$. It follows that in $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$, $\sum_{q'=1}^{l(h_j)}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \leq c_j - 1$.

Since h_j prefers r_{2i-1} or r_{2i} to some member of $M(h_j)$ and h_j also has a free post it follows that

$$\sum_{q'=1}^{q-1}\{x_{i',p''} : (r_{i'}, p'') \in R(h_j, q')\} \leq (c_j - 2).$$

Further, since (r_{2i-1}, r_{2i}) is unassigned or prefers (h_{j_1}, h_{j_2}) to $(M(r_{2i-1}), M(r_{2i}))$ it follows that

$$c_j \sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = c_j.$$

Hence,

$$\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j - 1$$

Thus in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}\,\rangle$, the RHS of Constraint 4.3.2 is at most $c_j - 1$ and the LHS is greater than $c_j - 1$ and thus Constraint 4.3.2 is not satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}\,\rangle$, a contradiction. Thus no such $((r_{2i-1}, r_{2i}), (h_j, h_j))$ can block M.

Type 3(d) Blocking Pair - Assume $h_{j_1} = h_{j_2} = h_j$ and further h_j is fully subscribed and also has two assignees r_s and r_t (where $s \neq t$ and neither s nor t is equal to r_{2i-1} or r_{2i}) such that h_j prefers r_{2i-1} to r_s and h_j also prefers r_{2i} to r_t. Let r_{min} be the better of r_{2i} and r_{2i-1} according to hospital h_j with $rank(h_j, r_{min}) = q_{min}$. Analogously, let r_{max} be the worse of r_{2i} and r_{2i-1} according to hospital h_j with $rank(h_j, r_{max}) = q_{max}$.

Now, it follows that both

$$\sum_{q'=1}^{q_{min}-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j - 1$$

and

$$\sum_{q'=1}^{q_{max}-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} < c_j$$

in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}\,\rangle$. Hence $\beta_{j,q_{min}} = 1$ and $\alpha_{j,q_{max}} = 1$ and moreover

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} = 1$$

and thus Constraint 4.21 is not satisfied in $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}\,\rangle$, a contradiction. Hence no such $((r_{2i-1}, r_{2i}), (h_{j_1}, h_{j_2}))$ can block M and the result is proven. □

The existence of the objective function (given by Inequality 4.22) immediately leads to the following corollary.

Corollary 4.3.2. *Given an instance I of HRC, let J be the corresponding IP model as defined in Section 4.3.1 and Section 4.3.2. A maximum cardinality stable matching in I is exactly* *equivalent to an optimal solution to J.*

4.3.4 Complexity of the IP model for HRC

The model has $O(m)$ binary-valued variables and $O(m+cL^2)$ constraints where m is the total length of the single residents' preference lists and the coupled residents' projected preference lists, c is number of couples and L is the maximum length of a couple's preference list. The space complexity of the model is $O(m(m + cL^2))$ and the model can be built in $O(m(m + cL^2))$ time in the worst case for an arbitrary instance.

4.4 Creating the IP model from an example HRC instance

Residents

$$(r_1, r_2): \quad (h_1, h_2) \ (h_2, h_1) \ (h_2, h_3)$$

$$\begin{array}{ll} r_3: & h_1 \ h_3 \\ r_4: & h_2 \ h_3 \\ r_5: & h_2 \ h_1 \\ r_6: & h_1 \ h_2 \end{array}$$

Hospitals

$$\begin{array}{lll} h_1: & 2: & r_1 \ r_3 \ r_2 \ r_6 \ r_5 \\ h_2: & 2: & r_2 \ r_6 \ r_1 \ r_4 \ r_5 \\ h_3: & 2: & r_4 \ r_3 \ r_2 \end{array}$$

Figure 4.1: Example instance of HRC.

Let I be the example instance of HRC shown in Figure 4.1 where the capacity of each hospital in I is shown after the first colon, followed by the preference list after the second colon. We shall consider the creation of the corresponding IP model J for the example instance I. For each resident $r_i \in I$ $(1 \leq i \leq 6)$ construct a vector x_i consisting of $l(r_i)+1$ binary variables, $x_{i,p}$ $(1 \leq p \leq l(r_i) + 1)$, as shown in Figure 4.2, and apply the constraints as described in Section 4.3. Thus, we form an IP model J derived from I.

Let $\mathbf{x}^u$ denote the assignment of values to the variables in J shown in Figure 4.3. We will show that $\mathbf{x}^u$ is not a feasible solution for J and thus, by Theorem 4.3.1, does not correspond to a stable matching in I. However, as all instantiations of Constraints 4.6 - 4.11 hold

$$\begin{array}{ll} x_1: & \langle\, x_{1,1} \;\; x_{1,2} \;\; x_{1,3} \;\; x_{1,4} \,\rangle \\ x_2: & \langle\, x_{2,1} \;\; x_{2,2} \;\; x_{2,3} \;\; x_{2,4} \,\rangle \\ x_3: & \langle\, x_{3,1} \;\; x_{3,2} \;\; x_{2,3} \,\rangle \\ x_4: & \langle\, x_{4,1} \;\; x_{4,2} \;\; x_{4,3} \,\rangle \\ x_5: & \langle\, x_{5,1} \;\; x_{5,2} \;\; x_{5,3} \,\rangle \\ x_6: & \langle\, x_{6,1} \;\; x_{6,2} \;\; x_{6,3} \,\rangle \end{array}$$

Figure 4.2: Variables created in J from the instance of HRC shown in Figure 4.1.

for $\mathbf{x}^u$, $\mathbf{x}^u$ does correspond to a matching in I, namely $M_u = \{(r_1, h_2), (r_2, h_3), (r_3, h_1), (r_4, h_3), (r_5, h_1), (r_6, h_2)\}$. We shall demonstrate that several constraints in J are violated by $\mathbf{x}^u$ and that these constraints correspond to blocking pairs of M_u in I.

$$\begin{array}{ll} x_1: & \langle\, 0 \;\; 0 \;\; 1 \;\; 0 \,\rangle \\ x_2: & \langle\, 0 \;\; 0 \;\; 1 \;\; 0 \,\rangle \\ x_3: & \langle\, 1 \;\; 0 \;\; 0 \,\rangle \\ x_4: & \langle\, 0 \;\; 1 \;\; 0 \,\rangle \\ x_5: & \langle\, 0 \;\; 1 \;\; 0 \,\rangle \\ x_6: & \langle\, 0 \;\; 1 \;\; 0 \,\rangle \end{array}$$

Figure 4.3: The assignment of values, $\mathbf{x}^u$, to the variables in J corresponding to the unstable matching M_u in the instance of HRC shown in Figure 4.1.

Inequality 4.23 represents the instantiation of Constraint 4.12 in the case that $i = 6$ and $p = 1$. The LHS of Inequality 4.23 is the product of the capacity of h_1 and the values of the variables that represent r_6 being assigned to a worse partner than h_1 or being unassigned. The RHS of Inequality 4.23 is the summation of the values of the variables that indicate whether h_1 is assigned to partners it prefers to r_6.

$$c_1(x_{6,2} + x_{6,3}) \leq x_{1,1} + x_{3,1} + x_{2,2} \tag{4.23}$$

The acceptable pair (r_6, h_1) is a Type 1 blocking pair for M_u in I. In this case the LHS of Inequality 4.23 equals two and the RHS of Inequality 4.23 equals one. Hence Inequality 4.23 is not satisfied in $\mathbf{x}^u$ and thus $\mathbf{x}^u$ is not a feasible solution to J.

Inequality 4.24 represents the instantiation of Constraint 4.15 in the case that $p1 = 2$, $p2 = 3$ and $r = 3$. In this case the LHS of Inequality 4.24 is the product of the capacity of h_1 and the value of the variable that represents r_1 being assigned at position three on its projected preference list (and thus, since no instance of Constraint 4.11 is violated, (r_1, r_2) being assigned to the pair in position three on its joint projected preference list). The RHS of Inequality 4.24 is the summation of the values of variables which indicate whether h_1 is assigned to partners it prefers to r_2.

$$c_1(x_{1,3}) \leq x_{1,1} + x_{3,1} \tag{4.24}$$

The acceptable pair $((r_1, r_2), (h_2, h_1))$ is a Type 2 blocking pair of M_u in I. In this case the LHS of Inequality 4.24 equals 2 and the RHS of Inequality 4.24 equals 1. Hence Inequality 4.24 is not satisfied in $\mathbf{x}^u$ and thus $\mathbf{x}^u$ is not a feasible solution to J.

Inequality 4.25 represents the instantiation of Constraint 4.19 in the case that $i = 1$ and $p = 1$. In this case the summation on the LHS of Inequality 4.25 is over the variables that represent r_1 being assigned to a worse partner than h_1 in position one on its projected preference or unassigned. (Since no instance of Constraint 4.11 is violated in J these variables equally represent (r_1, r_2) being assigned to a worse joint partner than (h_1, h_2) or being unassigned). Also, $\alpha_{1,1}$ is a variable constrained to take a value of one in the case that h_1 prefers less than c_1 residents to r_1. Similarly, $\alpha_{2,1}$ is a variable constrained to take a value of one in the case that h_2 prefers less than c_2 residents to r_1.

$$(x_{1,2} + x_{1,3} + x_{1,4}) + \alpha_{1,1} + \alpha_{2,1} \leq 2 \tag{4.25}$$

The acceptable pair $((r_1, r_2), (h_1, h_2))$ is a Type 3(a) blocking pair of M_u in I. In this case the summation on the LHS of Inequality 4.25 equals one. Also, since r_1 is in first position on h_1's preference list and thus h_1 prefers no other assignees to r_1, $\alpha_{1,1} \geq (1 - (0/2))$ and hence $\alpha_{1,1} = 1$. Similarly, $\alpha_{2,1} \geq (1 - (0/2))$ since r_2 is in first position on h_2's preference list and hence $\alpha_{2,1} = 1$. Thus Inequality 4.25 is not satisfied in $\mathbf{x}^u$ and $\mathbf{x}^u$ is not a feasible solution to J.

$$\begin{array}{ll}
x_1: & \langle\, 1\ \ 0\ \ 0\ \ 0 \,\rangle \\
x_2: & \langle\, 1\ \ 0\ \ 0\ \ 0 \,\rangle \\
x_3: & \langle\, 1\ \ 0\ \ 0 \,\rangle \\
x_4: & \langle\, 0\ \ 1\ \ 0 \,\rangle \\
x_5: & \langle\, 0\ \ 0\ \ 1 \,\rangle \\
x_6: & \langle\, 0\ \ 1\ \ 0 \,\rangle
\end{array}$$

Figure 4.4: The assignment of values, $\mathbf{x}^s$, to the variables in J corresponding to the stable matching M_s in the instance of HRC shown in Figure 4.1.

Let $\mathbf{x}^s$ denote the assignment of values to the variables in the IP model J shown in Figure 4.4. $\mathbf{x}^s$ is a feasible solution to the IP model J and as such does correspond with a stable matching in I, namely $M_s = \{(r_1, h_1), (r_2, h_2), (r_3, h_1), (r_4, h_3), (r_6, h_2)\}$.

Consider a potential blocking pair of M_s. (r_5, h_2) is an acceptable pair in I and r_5 is unassigned in M_s. Inequality 4.26 represents the instantiation of Constraint 4.12 in the case that $i = 5$ and $p = 1$. The LHS of Inequality 4.26 is the product of the capacity of h_2 and the values of the variables that represent r_5 being assigned to a worse partner than h_2. The RHS of Inequality 4.26 is the summation of the values of the variables that indicate whether h_2 is assigned to the partners it prefers to r_5.

$$c_2(x_{5,2} + x_{5,3}) \leq x_{2,1} + x_{6,2} + x_{1,2} + x_{1,3} + x_{4,1} \tag{4.26}$$

In this case the LHS of Inequality 4.26 equals two since r_5 is unassigned and the RHS of Inequality 4.26 also equals 2 since h_2 has two assignees that it prefers to r_5. Hence Inequality 4.26 is satisfied in $\mathbf{x}^s$. A similar consideration of other possible blocking pairs of M_s in I shows that no constraint is violated by $\mathbf{x}^s$ and thus $\mathbf{x}^s$ is a feasible solution of J.

4.5 An integer programming formulation for HRCT

The *Hospitals / Residents Problem with Couples and Ties* (HRCT) is a generalisation of HRC in which hospitals (respectively residents or couples) may find some subsets of their acceptable residents (respectively hospitals or hospital pairs) equally preferable. Residents (respectively hospitals or couples) that are found equally preferable by a hospital (respectively resident) are *tied* with each other in the preference list of that hospital (respectively resident or couple). In this section we show how to extend the IP model for HRC presented in Section 4.3 to the HRCT context. In order to do so we first define some additional notation.

For an acceptable resident-hospital pair (r_i, h_j), where r_i is a single resident let $rank(r_i, h_j) = q$ denote the rank that resident r_i assigns hospital h_j where $1 \leq q \leq l(r_i)$. Thus, $rank(r_i, h_j)$ is equal to the number of hospitals that r_i prefers to h_j plus one.

For an acceptable pair $((r_s, r_t), (h_j, h_k))$ where $c = (r_s, r_t)$ is a couple, let $rank(c, (h_j, h_k)) = q$ denote the rank that the couple c assigns the hospital pair (h_j, h_k) where $1 \leq q \leq l(c)$. Thus, $rank(c, (h_j, h_k))$ is equal to the number of hospital pairs that (r_s, r_t) prefers to (h_j, h_k) plus one.

For each single resident $r_i \in R$ and integer p $(1 \leq p \leq l(r_i))$ let

$$p_i^+ = \max\{p' : 1 \leq p' \leq l(r_i) \wedge rank(r_i, pref(r_i, p)) = rank(r_i, pref(r_i, p'))\}$$

Similarly, in the case of a couple $c_{i,j}$ and integer p $(1 \leq p \leq l(c))$ let

$$p_{i,j}^+ = \max\{p' : 1 \leq p' \leq l(c) \wedge rank(c, pref(c, p)) = rank(c, pref(c, p'))\}$$

Intuitively, for a single resident r_i, p_i^+ is the best position on r_i's preference list of a hospital appearing in the same tie on r_i's list as the hospital in position p on r_i's preference list. Also, for a couple (r_i, r_j), $p^+_{i,j}$ is the largest position on (r_i, r_j)'s joint preference list of a hospital pair appearing in the same tie on (r_i, r_j)'s preference list as hospital pair in position p on (r_i, r_j)'s joint preference list.

To correctly construct an IP model for HRCT we must make the following alterations to the mechanism described in Section 4.3 for obtaining an IP model from an HRC instance. All constraints are as before unless otherwise noted. Since, a hospital h_j may rank some members of $M(h_j)$ equally with r_i in HRCT, the summations involving q in Constraints 4.12 - 4.16 and 4.3.2 and the Inequalities 4.17 and 4.18 must now range from 1 to q.

Also, since a resident r_i may rank $M(r_i)$ equally with h_j, the summations involving p in Constraints 4.12, 4.19, 4.3.2 and 4.21 must now range from $p^+_i + 1$ to $l(r_i) + 1$. Further, we must extend the definition of p_1 and p_2 in Constraints 4.13 - 4.16 such that $1 \leq p_1 \leq p^+_s < p_2 \leq l(r_s)$ where r_s is the resident involved in each case.

To give an example of a modified constraint we now give a full description of the constraint applied to ensure that no blocking pairs of Type 1 are admitted by a feasible solution in the HRCT context:

Stability 1 - In a stable matching M in I, if a single resident $r_i \in R$ has a partner worse than some hospital $h_j \in H$ where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$, then h_j must be fully subscribed with partners at least as good as r_i. Thus, either

$$\sum_{p'=p^+_i+1}^{l(r_i)+1} x_{i,p'} = 0$$

or h_j is fully subscribed with partners at least as good as r_i, i.e.

$$\sum_{q'=1}^{q} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} = c_j$$

Thus, for each i $(2c + 1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$:

$$c_j \sum_{p'=p^+_i+1}^{l(r_i)+1} x_{i,p'} \leq \sum_{q'=1}^{q} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \tag{4.27}$$

The other constraints in the HRCT model follow by adapting the remaining stability criteria in analogous fashion. Using a proof analogous to that of Theorem 4.3.1, the following results may be established.

Theorem 4.5.1. *Given an instance I of HRCT, let J be the corresponding IP model as defined in Section 4.5. A stable matching in I is exactly equivalent to a feasible solution to J.*

By applying the same objective function as that applied in the HRC model we may establish the following corollary showing that an optimal solution in the IP model for HRCT must be equivalent to a maximum cardinality stable matching in the corresponding instance of HRCT.

Corollary 4.5.2. *Given an instance I of HRCT, let J be the corresponding IP model as defined in Section 4.5. A maximum cardinality stable matching in I is exactly equivalent to an optimal solution to J.*

4.6 An integer programming formulation for MIN BP HRC

Let I be an instance of HRC. A matching M in I is a 'most stable' matching in I if it admits the minimum number of blocking pairs taken over all of the matchings admitted by I. Clearly a stable matching in I, if one exists, is a 'most stable' matching in I. Let MIN BP HRC be the problem of finding a 'most stable' matching in an instance of HRC.

Let J be the IP model derived from I as described in Section 4.3. In this section we show how to modify J to find a maximum cardinality 'most stable' matching in an instance of HRC.

We demonstrate in Section 4.6.1 the additional variables required to extend J. Further, Section 4.6.2 gives the extensions of the constraints from the original HRC models adapted for the MIN BP HRC context.

4.6.1 Additional variables in the IP model for MIN BP HRC

For all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ let there be a variable $\theta_{i,p}$ such that:

$$\theta_{i,p} \in \{0, 1\} \tag{4.28}$$

Intuitively, $\theta_{i,p} = 1$ if resident r_i is involved in a blocking pair with the hospital at position p on his preference list, either as a single residents or as part of a couple, and $\theta_{i,p} = 0$ otherwise.

4.6.2 Replacement constraints in the IP model for MIN BP HRC

Stability 1 - In the MIN BP HRC model we replace Constraint 4.12 in the original HRC model with Constraint 4.29 as follows. For each i $(2c + 1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$:

$$c_j \left(\left(\sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} \right) - \theta_{i,p} \right) \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \tag{4.29}$$

By Theorem 4.3.1, if Constraint 4.12 does not hold, then r_i is involved in a blocking pair with the hospital at position p on his preference list (either as a single resident or as part of a couple) for some i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$. Clearly, Constraint 4.29 may only hold if $\theta_{i,p} = 1$. In this way, we can count the number of blocking pairs using the $\theta_{i,p}$ values. A similar methodology is used in all replacement constraints for the remaining stability criteria that follow. Ultimately, the number of blocking pairs is the sum of the $\theta_{i,p}$ values, except that to avoid counting a blocking pair twice in the case of a couple, the model will assume that $\theta_{2i,p} = 0$ for all i $(1 \leq i \leq c)$ and for all p $(1 \leq p \leq l(r_{2i}))$.

Stability 2(a) - In the MIN BP HRC model we replace Constraints 4.13 and 4.14 in the original HRC model with Constraints 4.30 and 4.31 as follows. For each i $(1 \leq i \leq c)$ and p_1, p_2 $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ such that $pref(r_{2i}, p_1) = pref(r_{2i}, p_2)$ and $rank(h_{j_1}, r_{2i-1}) = q$:

$$c_{j_1}(x_{2i-1,p_2} - \theta_{2i-1,p_1}) \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_1}, q')\} \tag{4.30}$$

However, for the special case in which $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_1}$ we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ such that $pref(r_{2i}, p_1) = pref(r_{2i}, p_2)$ and $rank(h_{j_1}, r_{2i-1}) = q$:

$$(c_{j_1} - 1)(x_{2i-1,p_2} - \theta_{2i-1,p_1}) \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : q' \neq rank(h_{j_1}, r_{2i}) \wedge (r_{i'}, p'') \in R(h_{j_1}, q')\} \tag{4.31}$$

Stability 2(b) - In the MIN BP HRC model we replace Constraints 4.15 and 4.16 in the original HRC model with Constraints 4.32 and 4.33 as follows. For each i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i}))$ such that $pref(r_{2i-1}, p_1) = pref(r_{2i-1}, p_2)$ and $rank(h_{j_2}, r_{2i}) = q$:

$$c_{j_2}(x_{2i-1,p_2} - \theta_{2i-1,p_1}) \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_{j_2}, q')\} \tag{4.32}$$

Again, for the special case in which $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_2}$ we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i}))$ such that $pref(r_{2i-1}, p_1) = pref(r_{2i-1}, p_2)$ and $rank(h_{j_2}, r_{2i}) = q$:

$$(c_{j_1}-1)(x_{2i-1,p_2}-\theta_{2i-1,p_1}) \le \sum_{q'=1}^{q-1}\{x_{i',p''} \in X : q' \neq rank(h_{j_2}, r_{2i-1}) \wedge (r_{i'}, p'') \in R(h_{j_2}, q')\} \tag{4.33}$$

Stability 3(a) - In the MIN BP HRC model we replace Constraint 4.19 with Constraint 4.34 as follows. For all i $(1 \le i \le c)$ and p $(1 \le p \le l(r_{2i-1}))$ where $h_{j_1} = pref(r_{2i-1}, p)$, $h_{j_2} = pref(r_{2i}, p)$, $h_{j_1} \neq h_{j_2}$, $rank(h_{j_1}, r_{2i-1}) = q_1$ and $rank(h_{j_2}, r_{2i}) = q_2$:

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \alpha_{j_1,q_1} + \alpha_{j_2,q_2} - \theta_{2i-1,p} \le 2 \tag{4.34}$$

Stability 3(b) & 3(c) - In the MIN BP HRC model we replace Constraint 4.3.2 with Constraint 4.6.2 as follows. For i $(1 \le i \le c)$ and p $(1 \le p \le l(r_{2i-1}))$ such that $pref(r_{2i-1}, p)$ $= pref(r_{2i}, p)$ and $h_j = pref(r_{2i-1}, p)$ where $q = \min\{rank(h_j, r_{2i}), rank(h_j, r_{2i-1})\}$:

$$c_j\left(\left(\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'}\right) - \theta_{2i-1,p}\right) - \frac{\sum_{q'=1}^{q-1}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\}}{(c_j - 1)} \le \sum_{q'=1}^{l(h_j)}\{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \tag{4.35}$$

Stability 3(d) - In the MIN BP HRC model we replace Constraint 4.21 with Constraint 4.36 as follows. For each (h_j, h_j) acceptable to (r_{2i-1}, r_{2i}), let r_{min} be the better of r_{2i-1} and r_{2i} according to hospital h_j with $rank(h_j, r_{min}) = q_{min}$. Analogously, let r_{max} be the worse of r_{2i} and r_{2i-1} according to hospital h_j with $rank(h_j, r_{max}) = q_{max}$. Thus we obtain the following constraint for i $(1 \le i \le c)$ and p $(1 \le p \le l(r_{2i-1}))$ such that $pref(r_{2i-1}, p) =$ $pref(r_{2i}, p) = h_j$.

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \alpha_{j,q_{max}} + \beta_{j,q_{min}} - \theta_{2i-1,p} \le 2 \tag{4.36}$$

4.6.3 Objective functions in the IP model for MIN BP HRC

A maximum cardinality matching 'most stable' matching M is a matching in which the maximum number of residents is assigned in M subject to having the minimum possible number of blocking pairs taken over all of the matchings admitted by I. To this end we apply the following objective functions in sequence.

First we find an optimal solution to the IP model which minimises the number of blocking pairs. To this end we apply the objective function shown in Equation 4.37 below. A matching M in I with the minimum number of blocking pairs taken over all of the matchings in I requires that the minimum number of $\theta_{i,p}$ must take the value of one. To minimise the sum over all of the values of i and p we apply the following objective function:

$$\min \sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \theta_{i,p} \tag{4.37}$$

The matching M returned after finding an optimal solution during the first iteration will be a 'most stable' matching in I. Let k be the number of blocking pairs in M. Now we seek a maximum cardinality matching in I with at most k blocking pairs. Thus we apply a constraint that ensures that any solution in the second run also has at most k blocking pairs as follows:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \theta_{i,p} \leq k \tag{4.38}$$

A maximum cardinality 'most stable' matching M is a matching in which the maximum number of residents are assigned in M subject to having the minimum possible number of blocking pairs taken over all of the matchings admitted by I. To maximise the size of the matching found, subject to Constraint 4.38 holding, we also apply the following objective function:

$$\max \sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} x_{i,p} \tag{4.39}$$

.

4.6.4 Proof of correctness the IP model for MIN BP HRC

We now establish the correctness of the IP model presented in Sections 4.6.1, 4.6.2 and 4.6.3.

Theorem 4.6.1. *Given an instance I of* MIN BP HRC, *let J be the corresponding IP model as defined in Section 4.6.1 and Section 4.6.2. A 'most stable' matching in I is exactly equivalent to a feasible solution to J.*

Proof. Let M be a matching in I. As in the proof of Theorem 4.3.1, let $\langle\, \mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}\, \rangle$ be the corresponding assignment of boolean values to the variables in the IP model derived from I as constructed in Section 4.3. By Theorem 4.3.1, r_i is involved in a blocking pair with h_j, either as a single resident or as part of couple, if and only if the corresponding constraint is violated in the HRC model.

Assume that r_i is single and (r_i, h_j) blocks M where $pref(r_i, p) = h_j$. Hence, by Theorem 4.3.1, an instance of Constraint 4.12 is violated with respect to $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$ and thus the corresponding instance of Constraint 4.29 must be violated if $\theta_{i,p} = 0$. However, if $\theta_{i,p} = 1$, then the LHS of Constraint 4.29 becomes 0 and the constraint is satisfied.

Now, assume that r_i is part of a couple, without loss of generality assume this couple be (r_i, r_j). Further assume that $((r_i, r_j), (h_k, h_l))$ blocks M where $pref((r_i, r_j), p) = (h_k, h_l)$. Hence, by Theorem 4.3.1, one of Constraints 4.13 to 4.21 is violated with respect to $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}$ $\rangle$ and thus the corresponding instance of Constraint 4.30 to 4.36 must be violated if $\theta_{i,p} = 0$. However, if $\theta_{i,p} = 1$, then the corresponding instances of Constraints 4.30 to 4.36 must be satisfied.

Conversely, let M be a matching in I and let $\langle$ **x**, $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta}$ $\rangle$ be the corresponding assignment of boolean values to the variables in the IP model derived from I as constructed in Section 4.6.

Now assume that $\theta_{i,p} = 1$ for some i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$. If r_i is not involved in a blocking pair with h_j where $pref(r_i, p) = h_j$ (either as a single resident or part of a couple), then by Theorem 4.3.1, Constraints 4.12 to 4.21 are satisfied, and hence so are Constraints 4.29 to 4.36, with $\theta_{i,p} = 0$, in contradiction to the objective function Equation 4.37.

Thus, if $\theta_{i,p} = 1$ for some i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_{n_1}))$, then r_i must be involved in a blocking pair with the hospital in position p on his preference list. Moreover, if the optimal value of the solution obtained from the model when applying the objective function Equation 4.37 for a given model J is k, then the minimum number of blocking pairs admitted by any matching in the corresponding HRC instance I is $\leq k$. Hence, the result is proven. □

The existence of the objective function (given by Inequality 4.39) immediately leads to the following corollary.

Corollary 4.6.2. *Given an instance I of* MIN BP HRC, *let J be the corresponding IP model as defined in Section 4.6.1 and Section 4.6.2. A maximum cardinality 'most stable' matching in I is exactly equivalent to an optimal solution to J.*

Chapter 5

An integer programming model for HRC under BIS-stability

5.1 Introduction

In this chapter we show by means of a pair of example instances, that BIS-stability and MM-stability are not equivalent. First, in Section 5.2, we describe two HRC instances: one that admits an MM-stable matching, but no BIS-stable matching; and another that admits a BIS-stable matching, but no MM-stable matching. Further, in Section 5.3 we present a cloning methodology for HRC that can be used to construct an instance of one-to-one HRC from an instance of many-to-one HRC such that the MM-stable matchings in the many-to-one instance correspond to the MM-stable matchings in the one-to-one instance. We then prove that this cloning method applies only under MM-stability and is not applicable under BIS-stability. Finally, in Section 5.4 we present the first IP model for finding a maximum cardinality stable matching in an arbitrary instance of HRC under BIS-stability.

5.2 MM-stability and BIS-stability are not equivalent

MM-stability and BIS-stability are not equivalent. We demonstrate this by means of the two instances shown in Figure 5.1 and Figure 5.2. Consider the instance of HRC shown in Figure 5.1 where h has capacity two. The matching $M = \{(r_3, h)\}$ is BIS-stable, but the instance admits no MM-stable matching.

Now, consider the instance of HRC shown in Figure 5.2 due to Irving [37], where h_1 has capacity two and h_2 has capacity one. The instance admits three distinct matchings, namely $M_1 = \{(r_1, h_1), (r_2, h_1)\}$, $M_2 = \{(r_3, h_1), (r_4, h_1)\}$ and $M_3 = \{(r_3, h_1), (r_4, h_2)\}$. Clearly,

Residents		
(r_1, r_2)	:	(h, h)
r_3	:	h

Hospitals						
h	:	2	:	r_1	r_3	r_2

Figure 5.1: An instance of HRC that admits a BIS-stable matching but admits no MM-stable matching.

Residents			
(r_1, r_2)	:	(h_1, h_1)	
(r_3, r_4)	:	(h_1, h_1)	(h_1, h_2)

Hospitals							
h_1	:	2	:	r_3	r_1	r_2	r_4
h_2	:	1	:	r_4			

Figure 5.2: An instance of HRC that admits an MM-stable matching but admits no BIS-stable matching.

M_2 is MM-stable. However, M_1 is BIS-blocked by (r_3, r_4) with (h_1, h_2), M_2 is BIS-blocked by (r_1, r_2) with (h_1, h_1), and M_3 is BIS-blocked by (r_3, r_4) with (h_1, h_1).

5.3 A hospital cloning method for HRC that works under MM-stability but not under BIS-stability

For an arbitrary instance I of HR in which the hospitals may have capacity greater than one, Gusfield and Irving [31] describe a method of constructing a corresponding instance, I' of HR in which all of the hospitals have capacity one, such that a stable matching in I corresponds to a stable matching in I' and vice versa. In this section we describe a method for producing an instance I' of HRC, in which all of the hospitals have capacity one, from an arbitrary instance I of HRC, in which the hospitals may have capacity greater than one, such that an MM-stable matching in I corresponds to an MM-stable matching in I' and vice versa. We show that this correspondence breaks down in the case of BIS-stability.

Let I be an instance of HRC with residents $R = \{r_1, r_2, \ldots, r_{n_1}\}$ and hospitals $H = \{h_1, h_2, \ldots, h_{n_2}\}$. Let c_j denote the capacity of hospital $h_j \in H$ $(1 \leq j \leq n_2)$, the number of available posts it has to which residents may be assigned. Without loss of generality, suppose residents $r_1, r_2, \ldots, r_{2c}$ are in couples. Again, without loss of generality, suppose that the couples are (r_{2i-1}, r_{2i}) $(1 \leq i \leq c)$. Let the single residents be $r_{2c+1}, r_{2c+2} \ldots r_{n_1}$.

Each single resident $r_i \in R$ has a preference list of length $l(r_i)$ consisting of individual hospitals $h_j \in H$. Further, each couple (r_{2i-1}, r_{2i}) has a joint preference list of acceptable hospital pairs of length $l((r_{2i-1}, r_{2i}))$. Each hospital $h_j \in H$ has a preference list of individual residents $r_i \in R$ of length $l(h_j)$.

We construct an instance I' of one-to-one HRC from I as follows. For each $h_j \in H$ we create c_j clones in I', $h_{j,1}, h_{j,2} \ldots h_{j,c_j}$, each of capacity one, where each clone represents one of the individual posts in h_j. For each $r_i \in R$ in I' replace each h_j in the preference list of a single resident r_i with the following sequence of hospitals $h_{j,1}, h_{j,2} \ldots h_{j,c_j}$.

For each couple (r_{2i-1}, r_{2i}) in I', replace each (h_{j_1}, h_{j_2}) (where $j_1 \neq j_2$) in (r_{2i-1}, r_{2i})'s joint preference list with the following sequence of hospital pairs:

$$L_1 = (h_{j_1,1}, h_{j_2,1}), (h_{j_1,2}, h_{j_2,1}) \ldots (h_{j_1,c_{j_1}}, h_{j_2,1}),$$

$$(h_{j_1,1}, h_{j_2,2}), (h_{j_1,2}, h_{j_2,2}) \ldots (h_{j_1,c_{j_1}}, h_{j_2,2}) \ldots (h_{j_1,c_{j_1}}, h_{j_2,c_{j_2}})$$

containing all of the possible pairings of the individual clones of h_{j_1} and h_{j_2}. Further in I' we replace each (h_j, h_j) in (r_{2i-1}, r_{2i})'s joint preference list with the following sequence of hospital pairs:

$$L_2 = (h_{j,2}, h_{j,1}), (h_{j,3}, h_{j,1}) \ldots (h_{j,c_j}, h_{j,1}), (h_{j,1}, h_{j,2}), (h_{j,3}, h_{j,2}) \ldots$$

$$\ldots (h_{j,c_j}, h_{j,2}) \ldots (h_{j,c_j-1}, h_{j,c_j})$$

where $\{(h_{j,x}, h_{j,y}) : x = y\} \cap L_2 = \emptyset$. Thus L_2 contains all possible pairings of distinct individual clones of h_j. We prove by the following Lemma that MM-stable matchings are preserved under this correspondence.

Theorem 5.3.1. *I admits an MM-stable matching if and only if I' does.*

Proof. We first prove that if I admits a MM-stable matching then I' also admits a MM-stable matching. Let M be an MM-stable matching in I. From M we construct an MM-stable matching M' in I' as follows. Take any hospital $h_j \in H$ and list its assignees as $r_{i_1}, r_{i_2} \ldots r_{i_{t_j}}$ where $t_j \leq c_j$. Assume without loss of generality that $rank(h_j, r_{i_1}) < rank(h_j, r_{i_2}) \ldots < rank(h_j, r_{i_{t_j}})$. For each k $(1 \leq k \leq t_j)$ add $(r_{i_k}, h_{j,k})$ to M'.

We require to prove that M' is a matching in I' and further that M' is MM-stable. First we prove that M' is a matching in I'. Clearly, all single residents r_i who are assigned to a hospital h_j in M are assigned to an acceptable hospital clone $h_{j,k}$ in M', for some k $(1 \leq k \leq c_j)$.

All couples (r_{2i-1}, r_{2i}) assigned to some (h_{j_1}, h_{j_2}) in M are assigned in M' to $(h_{j_1,k_1}, h_{j_2,k_2})$ for some k_1 $(1 \leq k_1 \leq c_{j_1})$ and k_2 $(1 \leq k_2 \leq c_{j_2})$. Since (h_{j_1}, h_{j_2}) is an acceptable pair of hospitals for (r_{2i-1}, r_{2i}) in I, $(h_{j_1,k_1}, h_{j_2,k_2})$ must be an acceptable pair for (r_{2i-1}, r_{2i}) in I' (note that if $j_1 = j_2$ then $k_1 \neq k_2$). Thus, M' is a matching in I'.

It remains to prove that M' is MM-stable in I'. Suppose not. Then there exists some MM-blocking pair for M' in I'. We now prove that no such blocking pairs can exist for M' in I'. Since no preferences are expressed by a couple in I' for a hospital pair $(h_{j,k}, h_{j,k})$ consisting of two identical clones, only MM-blocking pairs of Types 1, Case 2(a), Case 2(b) and 3(a) shown in Definition 2.3.1 are possible in I'. We now consider each of these possible types of blocking pair in turn and show that M' can admit no blocking pair of each type in I'.

Type 1 Stability: Assume a single resident r_i MM-blocks M' in I' with a hospital clone $h_{j,k}$ as part of a Type 1 blocking pair as defined in Definition 2.3.1. Hence, in I' a resident r_i is either unassigned or prefers $h_{j,k}$ to $M'(r_i)$ and moreover $h_{j,k}$ is either undersubscribed or prefers r_i to $M'(h_{j,k})$. Now, either $M'(r_i) = h_{j,l}$ for some l $(1 \leq l \leq c_j)$ or $M'(r_i) \neq h_{j,l}$ for all l $(1 \leq l \leq c_j)$ in I'.

Assume $M'(r_i) = h_{j,l}$ for some l $(1 \leq l \leq c_j)$. Then r_i prefers $h_{j,k}$ to $h_{j,l}$ and $k < l$. Further, $h_{j,k}$ prefers r_i to its assignee $M'(h_{j,k})$, a contradiction to the construction of M' if $k < l$. Now assume $M'(r_i) \neq h_{j,l}$ for all l $(1 \leq l \leq c_j)$. It follows that r_i is either unassigned in M or prefers h_j to $M(r_i)$ and moreover h_j is either undersubscribed in M or prefers r_i to some member of $M(h_j)$. Thus (r_i, h_j) forms an MM-blocking pair of M in I, a contradiction. Hence $(r_i, h_{j,k})$ cannot MM-block M' in I' as part of a Type 1 blocking pair.

Type 2(a) stability: Assume a couple (r_{2i-1}, r_{2i}) MM-blocks M' with $(h_{j_1,k_1}, h_{j_2,k_2})$ in I' for some k_1, k_2 $(1 \leq k_1 \leq c_{j_1}, 1 \leq k_2 \leq c_{j_2})$ where $M'(r_{2i}) = h_{j_2,k_2}$ as part of a Type 2(a) blocking pair as defined in Definition 2.3.1. Clearly either $j_1 \neq j_2$ or $j_1 = j_2$.

First, we consider the case where $j_1 \neq j_2$. It follows that (r_{2i-1}, r_{2i}) prefers $(h_{j_1,k_1}, h_{j_2,k_2})$ to $(M'(r_{2i-1}), M'(r_{2i}))$ where $M'(r_{2i}) = h_{j_2,k_2}$ and h_{j_1,k_1} is either undersubscribed in M' or prefers r_{2i-1} to $M'(h_{j_1,k_1})$. Now, either $M'(r_{2i-1}) = h_{j_1,l}$ for some l $(1 \leq l \leq c_{j_1})$ or $M'(r_{2i-1}) \neq h_{j_1,l}$ for all l $(1 \leq l \leq c_{j_1})$.

Assume $M'(r_{2i-1}) = h_{j_1,l}$ for some l $(1 \leq l \leq c_{j_1})$. Then (r_{2i-1}, r_{2i}) prefers $(h_{j_1,k_1}, h_{j_2,k_2})$ to $(h_{j_1,l}, h_{j_2,k_2})$ and thus $k_1 < l$. Further, h_{j_1,k_1} prefers r_i to $M'(h_{j_1,k_1})$, a contradiction to the construction of M' if $k_1 < l$. Now assume $M'(r_{2i-1}) \neq h_{j_1,l}$ for all l $(1 \leq l \leq c_{j_1})$. Thus either h_{j_1,k_1} is undersubscribed in M' or h_{j_1,k_1} prefers r_{2i-1} to $M'(h_{j_1,k_1})$. Now, if h_{j_1,k_1} is undersubscribed in M' then h_{j_1} must be undersubscribed in M and (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_{j_1}, h_{j_2}), a contradiction. Further, if h_{j_1,k_1} prefers r_{2i-1} to $M'(h_{j_1,k_1})$ then h_{j_1} must prefer r_{2i-1} to some member of $M(h_{j_1})$ in M and (r_{2i-1}, r_{2i}) MM-blocks M with (h_{j_1}, h_{j_2}) in I, a contradiction. Thus we have shown that no such blocking pair can exist if $j_1 \neq j_2$.

We now consider the case in which $j_1 = j_2$ and let $j = j_1 = j_2$. Then (r_{2i-1}, r_{2i}) prefers (h_{j,k_1}, h_{j,k_2}) to $(M'(r_{2i-1}), M'(r_{2i}))$ where $M'(r_{2i}) = h_{j,k_2}$ and h_{j,k_1} is either undersubscribed in M' or prefers r_{2i-1} to $M'(h_{j,k_1})$. Now, either $M'(r_{2i-1}) = h_{j,l}$ for some l $(1 \leq l \leq c_{j_1})$ or $M'(r_{2i-1}) \neq h_{j,l}$ for all l $(1 \leq l \leq c_j)$.

Assume $M'(r_{2i-1}) = h_{j,l}$ for some l $(1 \leq l \leq c_j)$. Then (r_{2i-1}, r_{2i}) prefers (h_{j,k_1}, h_{j,k_2}) to $(h_{j,l}, h_{j,k_2})$ and $k_1 < l$. Further, h_{j,k_1} prefers r_i to its assignee $M'(h_{j,k})$, a contradiction to the construction of M' if $k_1 < l$.

Now assume $M'(r_{2i-1}) \neq h_{j,l}$ for all l $(1 \leq l \leq c_{j_1})$. Thus either h_{j,k_1} is undersubscribed in M' or h_{j,k_1} prefers r_{2i-1} to $M'(h_{j,k_1})$. If h_{j,k_1} is undersubscribed in M' then h_j must be undersubscribed in M and (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_j, h_j), a contradiction. Further, if h_{j,k_1} prefers r_{2i-1} to $M'(h_{j,k_1})$ then h_j must prefer r_{2i-1} to some member of $M(h_j)$ and (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_j, h_j), a contradiction. Thus we have shown that no such blocking pair can exist if $j_1 = j_2$.

Type 2(b) stability: A similar argument may be applied when considering the case that (r_{2i-1}, r_{2i}) prefers $(h_{j_1,k_1}, h_{j_2,k_2})$ to $(M'(r_{2i-1}), h_{j_2,k_2})$ in I' to show that no Type 2(b) MM-blocking pair, as defined in Definition 2.3.1, is admitted by M'.

Type 3(a) stability: Assume a couple (r_{2i-1}, r_{2i}) MM-blocks M' in I' as part of a Type 3(a) blocking pair as defined in Definition 2.3.1. It follows that (r_{2i-1}, r_{2i}) MM-blocks M' in I' with $(h_{j_1,k_1}, h_{j_2,k_2})$ (where $j_1 \neq j_2$) for some k_1 $(1 \leq k_1 \leq c_{j_1})$ and k_2 $(1 \leq k_2 \leq c_{j_2})$ where $M'(r_{2i-1}) \neq h_{j_1,k_1}$ and $M'(r_{2i}) \neq h_{j_2,k_2}$. Thus (r_{2i-1}, r_{2i}) is either unassigned in M' or assigned in M' to a worse hospital pair than $(h_{j_1,k_1}, h_{j_2,k_2})$ (where $j_1 \neq j_2$) for some k_1 $(1 \leq k_1 \leq c_{j_1}), k_2$ $(1 \leq k_2 \leq c_{j_2})$ and moreover both of h_{j_1,k_1} and h_{j_2,k_2} are either undersubscribed in M' or assigned to a worse partner than r_{2i-1} and r_{2i} respectively in M'.

Assume that both $M'(r_{2i-1}) \neq h_{j_1,l}$ for all l $(1 \leq l \leq c_{j_1})$ and $M'(r_{2i}) \neq h_{j_2,l}$ for all l $(1 \leq l \leq c_{j_2})$. Since (r_{2i-1}, r_{2i}) MM-blocks M' in I' with $(h_{j_1,k_1}, h_{j_2,k_2})$ it follows that (r_{2i-1}, r_{2i}) is unassigned in M or prefers (h_{j_1}, h_{j_2}) to $(M(r_{2i-1}), M(r_{2i}))$. Moreover both of h_{j_1} and h_{j_2} are either undersubscribed or prefer r_{2i-1} and r_{2i} respectively to some member of $M(h_{j_1})$ and $M(h_{j_2})$ respectively. Thus (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_{j_1}, h_{j_2}), a contradiction.

Now, assume that $M'(r_{2i-1}) = h_{j_1,l}$ for some l $(1 \leq l \leq c_j)$. Since (r_{2i-1}, r_{2i}) MM-blocks M' in I' with $(h_{j_1,k_1}, h_{j_2,k_2})$ it follows that (r_{2i-1}, r_{2i}) prefers $(h_{j_1,k_1}, h_{j_2,k_2})$ to $(h_{j_1,l}, M'(r_{2i}))$ and thus $k_1 < l$. Further, h_{j_1,k_1} prefers r_i to $M'(h_{j_1,k_1})$, in contradiction to the construction of M' if $k_1 < l$. A similar argument may be applied in the case that $M'(r_{2i}) = h_{j_2,l}$ for some l $(1 \leq l \leq c_{j_2})$. Thus we have that an MM-stable matching in M corresponds to an MM-stable matching in M'.

Conversely, we now prove that if I' admits a MM-stable matching then I also admits a MM-stable matching. Let M' be an MM-stable matching in I'. We construct a set of pairs M in I from M' as follows. For all pairs $(r_i, h_{j,k})$ in M' such that $h_j \in H$ and $1 \leq k \leq c_j$, add (r_i, h_j) to M. We require to prove that M is a matching in I and further that M is stable.

First we prove that M is a matching in I. For each $h_j \in H$, the assignees of h_j in M are simply the assignees of the k copies of h_j in I' and since $k \leq c_j$ no h_j is oversubscribed in M. All single residents r_i who are assigned in M' to a hospital clone $h_{j,k}$ are assigned to an acceptable hospital h_j in M. All couples (r_{2i-1}, r_{2i}) assigned to some $(h_{j_1,k_1}, h_{j_2,k_2})$ in M' are assigned in M to (h_{j_1}, h_{j_2}) (note that possibly $j_1 = j_2$). Since $(h_{j_1,k_1}, h_{j_2,k_2})$ is an acceptable pair of hospital clones for (r_{2i-1}, r_{2i}) in I', (h_{j_1}, h_{j_2}) must be an acceptable pair of hospitals for (r_{2i-1}, r_{2i}) in I. Hence, M is a matching in I.

It remains to prove that M is MM-stable in I. Suppose not. Then there exists an MM-blocking pair of M in I. We now consider each of the possible MM-blocking pair types, defined in Definition 2.3.1, in turn and show that no MM-blocking pair of each type is admitted by M.

Type 1 stability: Assume M admits a Type 1 MM-blocking pair in I as defined in Definition 2.3.1. It follows that a single resident r_i and hospital h_j MM-block M in I. Thus resident r_i is unassigned or prefers h_j to $M(r_i)$ and moreover h_j is undersubscribed or prefers r_i to some $r_p \in M(h_j)$ in M. If h_j is undersubscribed in M then some $h_{j,k}$ is undersubscribed in M'. Further, if h_j prefers r_i to some $r_p \in M(h_j)$ then let k $(1 \leq k \leq c_j)$ be such that $r_p = M'(h_{j,k})$. From the construction, it follows that in M', r_i is unassigned or prefers $h_{j,k}$ to $M'(r_i)$ and $h_{j,k}$ is also either undersubscribed or prefers r_i to $r_p = M'(h_{j,k})$, and thus $(r_i, h_{j,k})$ MM-blocks M' in I', a contradiction. Thus M admits no Type 1 MM-blocking pair in I.

Type 2(a) stability: Assume M admits a Type 2(a) MM-blocking pair in I as defined in Definition 2.3.1. It follows that a couple (r_{2i-1}, r_{2i}) MM-blocks M in I with $(h_{j_1}, M(h_{j_2}))$. Hence, h_{j_1} is either undersubscribed in M or prefers r_{2i-1} to some $r_p \in M(h_{j_1})$. Firstly, assume h_{j_1} is undersubscribed in M. Then there exists a hospital clone h_{j_1,k_1} for some k_1 $(1 \leq k_1 \leq c_{j_1})$ that is undersubscribed in M'. Further, since r_{2i} is assigned to h_{j_2} in M there must be a hospital clone h_{j_2,k_2} for some k_2 $(1 \leq k_2 \leq c_{j_2})$ assigned to r_{2i} in M'. Thus, (r_{2i-1}, r_{2i}) must MM-block M' in I' with $(h_{j_1,k_1}, h_{j_2,k_2})$, a contradiction. Secondly, assume that h_{j_1} prefers r_{2i-1} to some $r_p \in M(h_{j_1})$. Let k_1 $(1 \leq k_1 \leq c_{j_1})$ be such that $(r_p, h_{j_1,k_1}) \in M'$. Thus h_{j_1,k_1} prefers r_{2i-1} to r_p. Further, there exists some k_2 $(1 \leq k_1 \leq c_{j_2})$ such that $(r_{2i}, h_{j_2,k_2}) \in M'$. Hence (r_{2i-1}, r_{2i}) MM-blocks M' with $(h_{j_1,k_1}, h_{j_2,k_2})$ in I', a contradiction. Thus, M admits no Type 2(a) MM-blocking pair in I.

Type 2(b) stability: A similar argument may be applied in the symmetric case where $M(r_{2i-1}) = h_{j_1})$ to prove that M admits no Type 2(b) MM-blocking pair in I as defined in Definition 2.3.1.

Type 3(a) stability: Assume M admits a Type 3(a) MM-blocking pair in I as defined in Definition 2.3.1. It follows that a couple (r_{2i-1}, r_{2i}) MM-blocks M with (h_{j_1}, h_{j_2}) in I (where $j_1 \neq j_2$). Then (r_{2i-1}, r_{2i}) is either unassigned or assigned to a worse pair in M than (h_{j_1}, h_{j_2}) and moreover each of h_{j_1} and h_{j_2} is either undersubscribed or has a worse partner than r_{2i-1} and r_{2i} respectively amongst their assignees in M.

If h_{j_1} (respectively h_{j_2}) is undersubscribed in M then there exists a hospital clone h_{j_1,k_1} for some k_1 $(1 \leq k_1 \leq c_{j_1})$ (respectively h_{j_2,k_2} for some k_2 $(1 \leq k_2 \leq c_{j_2})$) that is undersubscribed in M'. Further, if h_{j_1} (respectively h_{j_2}) has amongst its assignees in M a worse partner than r_{2i-1} (respectively r_{2i}) then there must be some h_{j_1,k_1} $(1 \leq k_1 \leq c_{j_1})$ (respectively h_{j_2,k_2} $(1 \leq k_2 \leq c_{j_1})$ that has a worse assignee than r_{2i-1} (respectively r_{2i}) in M'. From the construction, it follows that in M', (r_{2i-1}, r_{2i}) is either unassigned or prefers $(h_{j_1,k_1}, h_{j_2,k_2})$ to $M'(r_{2i-1}, r_{2i})$ and also each of h_{j_1,k_1} and h_{j_2,k_2} is either undersubscribed in M' or prefers r_{2i-1} and r_{2i} to $M'(h_{j_1,k_1})$ and $M'(h_{j_2,k_2})$ respectively. Hence (r_{2i-1}, r_{2i}) MM-blocks M' in I' with $(h_{j_1,k_1}, h_{j_2,k_2})$, a contradiction. Thus, M admits no Type 3(a) MM-blocking pair in I.

Type 3(b) stability: Assume M admits a Type 3(b) MM-blocking pair in I as defined in Definition 2.3.1. It follows that a couple (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_j, h_j) and that h_j also has two free posts. Then (r_{2i-1}, r_{2i}) is either unassigned or assigned in M to a worse hospital pair than (h_j, h_j) in M. From the construction this means that in M' there are two hospital clones h_{j,k_1} and h_{j,k_2} for some $1 \leq k_1 \leq c_j$ and $1 \leq k_2 \leq c_j$ where $k_1 \neq k_2$ such that both h_{j,k_1} and h_{j,k_2} are undersubscribed in M'. Since (r_{2i-1}, r_{2i}) has a worse partner in M' than (h_{j,k_1}, h_{j,k_2}), it follows that (r_{2i-1}, r_{2i}) MM-blocks M' in I' with (h_{j,k_1}, h_{j,k_2}), a contradiction. Thus, M admits no Type 3(b) MM-blocking pair in I.

Type 3(c) stability: Assume M admits a Type 3(c) MM-blocking pair in I as defined in Definition 2.3.1. It follows that a couple (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_j, h_j) and h_j has one free post and prefers at least one of r_{2i-1} or r_{2i} to some $r_p \in M(h_{j_1})$. Thus, (r_{2i-1}, r_{2i}) is either unassigned or assigned to a worse hospital pair than (h_j, h_j). Now, if h_j prefers r_{2i-1} to r_p in I then (r_{2i-1}, r_{2i}) MM blocks M' with (h_{j,k_2}, h_{j,k_1}) in I', a contradiction. Otherwise h_j prefers r_{2i} to r_p in I and it follows that (r_{2i-1}, r_{2i}) MM-blocks M' with (h_{j,k_1}, h_{j,k_2}) in I', a contradiction. Thus, M admits no Type 3(c) MM-blocking pair in I.

Type 3(d) stability: Assume M admits a Type 3(d) MM-blocking pair in I as defined in Definition 2.3.1. Assume a couple (r_{2i-1}, r_{2i}) MM-blocks M in I with (h_j, h_j) where h_j is full and h_j prefers r_{2i-1} to some $r_p \in M(h_j)$ and also prefers r_{2i} to some $r_q \in M(h_j) \setminus \{r_p\}$. Then, (r_{2i-1}, r_{2i}) is either unassigned in M or assigned to a worse hospital pair than (h_j, h_j)

Now, if h_j prefers r_{2i-1} in I to one of its assignees in M there exists a hospital clone h_{j,k_1} for some k_1 $(1 \leq k_1 \leq c_j)$ that has an assignee worse than r_{2i-1} in M'. Let this worse assignee be r_p. Now, if h_j prefers r_{2i} to some member of $M(h_j) \setminus \{r_p\}$ there exists a hospital clone h_{j,k_2} for some k_2 ($1 \leq k_2 \leq c_j$) that has a worse assignee than r_{2i}. From the construction it follows that in I', there are two distinct hospital clones h_{j,k_1} and h_{j,k_2} such that (r_{2i-1}, r_{2i}) prefers (h_{j,k_1}, h_{j,k_2}) to $(M'(h_{j,k_1}), M'(h_{j,k_2}))$. Also, h_{j,k_1} prefers r_{2i-1} to $M'(h_{j,k_1})$ and h_{j,k_2} prefers r_{2i} to $M'(h_{j,k_2})$. Hence (r_{2i-1}, r_{2i}) MM-blocks M' in I' with (h_{j,k_1}, h_{j,k_2}), a contradiction. Thus, M admits no Type 3(d) MM-blocking pair in I. Hence, if I' admits a MM-stable matching then I also admits a MM-stable matching. It follows that I admits an MM-stable matching if and only if I' does and the result is proven. □

Corollary 5.3.2. *If I' admits a BIS-stable matching it need not be the case that I admits a BIS-stable matching.*

Proof. Let I be the instance of HRC as shown in Figure 5.2. Clearly, the instance admits no BIS-stable matching. We construct the instance I' from I as described in Lemma 5.3.1 by creating two distinct hospital clones $h_{1,1}$ and $h_{1,2}$ to represent the two posts in h_1 and amending the preference lists of the couples as shown in Figure 5.3.

The instance I' admits the BIS-stable matching $M' = \{(r_3, h_{1,1}), (r_4, h_{1,2})\}$. Since I admits no BIS-stable matching, M' clearly has no corresponding BIS-stable matching in I and the cloning method described above does not work under BIS-stability. □

Residents' Preferences

(r_1, r_2)	:	$(h_{1,1}, h_{1,2})$	$(h_{1,2}, h_{1,1})$		
(r_3, r_4)	:	$(h_{1,1}, h_{1,2})$	$(h_{1,2}, h_{1,1})$	$(h_{1,1}, h_2)$	$(h_{1,2}, h_2)$

Hospitals' Preferences

$h_{1,1}$	:	1	:	r_3	r_1	r_2	r_4
$h_{1,2}$	:	1	:	r_3	r_1	r_2	r_4
h_2	:	1	:	r_4			

Figure 5.3: An instance of HRC that shows that the cloning method does not work under BIS-stability.

5.4 An IP formulation for HRC under BIS stability

The IP model presented in this section extends the model for HR presented in Section 4.2 in a similar fashion to the IP model for HRC under MM-stability described in Section 4.3. This extended model is designed around a series of linear inequalities that establish the absence of blocking pairs according to each of the different parts of Definition 2.3.2. The variables are defined for each resident, whether single or a member of a couple, and for each element on his/her preference list (with the possibility of being unassigned). A further consistency constraint ensures that each member of a couple obtains hospitals from the same pair in their list, if assigned. Finally, the objective of the IP is to maximise the size of a stable matching, if one exists.

We now define an instance of HRC and show the *projected preference lists* for each of the two residents involved in a couple may be derived from the couples joint preference lists. Let I be an instance of HRC with residents $R = \{r_1, r_2, \ldots, r_{n_1}\}$ and hospitals $H = \{h_1, h_2, \ldots, h_{n_2}\}$. Without loss of generality, suppose residents $r_1, r_2 \ldots r_{2c}$ are in couples. Again, without loss of generality, suppose that the couples are (r_{2i-1}, r_{2i}) $(1 \leq i \leq c)$. Suppose that the joint preference list of a couple $c_i = (r_{2i-1}, r_{2i})$ is:

$$c_i \;:\; (h_{\alpha_1}, h_{\beta_1}), (h_{\alpha_2}, h_{\beta_2}) \ldots (h_{\alpha_l}, h_{\beta_l})$$

From this list we create the following *projected preference list* for resident r_{2i-1}:

$$r_{2i-1} \;:\; h_{\alpha_1}, h_{\alpha_2} \ldots h_{\alpha_l}$$

and the following projected preference list for resident r_{2i}:

$$r_{2i} \;:\; h_{\beta_1}, h_{\beta_2} \ldots h_{\beta_l}$$

Clearly, the projected preference list of the residents r_{2i-1} and r_{2i} are the same length as the preference list of the couple $c_i = (r_{2i-1}, r_{2i})$. Let $l(c_i)$ denote the length of the preference list of c_i and let $l(r_{2i-1})$ and $l(r_{2i})$ denote the lengths of the projected preference lists of r_{2i-1} and r_{2i} respectively. Now we have that $l(r_{2i-1}) = l(r_{2i}) = l(c_i)$. A given hospital h_j may appear more than once in the projected preference list of a linked resident in a couple $c_i = (r_{2i-1}, r_{2i})$.

Let the single residents be $r_{2c+1}, r_{2c+2} \ldots r_{n_1}$, where each single resident r_i, has a preference list of length $l(r_i)$ consisting of individual hospitals $h_j \in H$. Each hospital $h_j \in H$ has a preference list of individual residents $r_i \in R$ of length $l(h_j)$. Further, each hospital $h_j \in H$ has capacity $c_j \geq 1$, the maximum number of residents to which it may be assigned.

When considering the exact nature of a blocking pair in this model, the stability definition due to Biró et al. [12] (BIS-stability) is applied in all cases in this section. The text in bold before the definition of a constraint shows the section of the BIS-stability definition with which the constraint corresponds. Hence, a constraint preceded by '**Stability 1**' is intended to prevent blocking pairs described by part 1 of the BIS-stability definition shown in Definition 2.3.2 in Section 2.3.

We describe the variables and constraints in the IP model for HRC under BIS-stability in

Sections 5.4.1 and 5.4.2 respectively. We do not prove the correctness of this IP model. However, the correctness can be established by a proof similar to the proof of Theorem 4.3.1.

5.4.1 Variables in the IP model for HRC under BIS stability

Let I be an instance of HRC as described in Section 4.3. Let J be the following Integer Programming (IP) formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

For $p = l(r_i) + 1$ define a variable $x_{i,p}$ whose intuitive meaning is that resident r_i is unassigned. Therefore we also have

$$x_{i,l(r_i)+1} = \begin{cases} 1 & \text{if } r_i \text{ is unassigned} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(r_i) + 1\}$. Let $pref(r_i, p)$ denote the hospital at position p of a single resident r_i's preference list or on the projected preference list of a resident belonging to a couple where $1 \leq i \leq n_1$ and $1 \leq p \leq l(r_i)$. Let $pref((r_{2i}, r_{2i-1}), p)$ denote the hospital pair at position p on the joint preference list of (r_{2i-1}, r_{2i}).

For an acceptable resident-hospital pair (r_i, h_j), let $rank(h_j, r_i) = q$ denote the rank which hospital h_j assigns resident r_i where $1 \leq j \leq n_2$, $1 \leq i \leq n_1$ and $1 \leq q \leq l(h_j)$.

Now, for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new variable $\alpha_{j,q} \in \{0, 1\}$. The intuitive meaning of a variable $\alpha_{j,q}$ is that if h_j is full with assignees better than rank q then $\alpha_{j,q}$ may take the value 0 or 1. Otherwise, $\alpha_{j,q} = 1$. Constraints 5.2 and 5.21 described in Section 5.4.2 are applied to enforce this property.

Further, for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new variable $\beta_{j,q} \in \{0, 1\}$. The intuitive meaning of a variable $\beta_{j,q}$ is that if h_j has more than $c_j - 2$ assignees better than rank q then $\beta_{j,q}$ may take a value of 0 or 1. Otherwise, $\beta_{j,q} = 1$. Constraints 5.3 and 5.27 described in Section 5.4.2 are applied to enforce this property.

Also, for all j $(1 \leq j \leq n_2)$ define a new variable $\gamma_j \in \{0, 1\}$. The intuitive meaning of a variable γ_j is that if h_j has $c_j - 1$ or more assignees then γ_j may take the value 0 or 1. Otherwise, $\gamma_j = 1$. Constraints 5.4 and 5.23 described in Section 5.4.2 are applied to enforce this property.

Now, for all j $(1 \leq j \leq n_2)$ define a new variable $\delta_j \in \{0, 1\}$. The intuitive meaning of a variable δ_j is that if h_j has c_j assignees then δ_j may take the value of 0 or 1. Otherwise, $\delta_j = 1$. Constraints 5.5 and 5.12 described in Section 5.4.2 are applied to enforce this property.

Finally, for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new variable $\varepsilon_{j,q} \in \{0, 1\}$. The intuitive meaning of a variable $\varepsilon_{j,q}$ is that if h_j has no assignees worse than rank q then $\varepsilon_{j,q}$ may take a value of 0 or 1. Otherwise, $\varepsilon_{j,q} = 1$. Constraints 5.6 and 5.11 described in Section 5.4.2 are applied to enforce this property.

For ease of exposition we define some additional notation. For each j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ let the set $R(h_j, q)$ contain the resident-position pairs (r_i, p) such that r_i is assigned a rank of q $(1 \leq q \leq l(h_j))$ by h_j and h_j is in position p $(1 \leq p \leq l(r_i))$ on r_i's preference list. Hence:

$$R(h_j, q) = \{(r_i, p) \in R \times \mathbb{Z} : rank(h_j, r_i) = q \wedge 1 \leq p \leq l(r_i) \wedge pref(r_i, p) = h_j\}$$

5.4.2 Constraints in the IP model for HRC under BIS stability

The following constraint simply confirms that each variable $x_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$:

$$x_{i,p} \in \{0, 1\} \tag{5.1}$$

The following constraint simply confirms that each variable $\alpha_{j,q}$ must be binary valued for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\alpha_{j,q} \in \{0, 1\} \tag{5.2}$$

The following constraint simply confirms that each variable $\beta_{j,q}$ must be binary valued for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\beta_{j,q} \in \{0, 1\} \tag{5.3}$$

The following constraint simply confirms that each variable γ_j must be binary valued for all j $(1 \leq j \leq n_2)$:

$$\gamma_j \in \{0, 1\} \tag{5.4}$$

The following constraint simply confirms that each variable δ_j must be binary valued for all j $(1 \leq j \leq n_2)$:

$$\delta_j \in \{0, 1\} \tag{5.5}$$

The following constraint simply confirms that each variable $\varepsilon_{j,q}$ must be binary valued for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$:

$$\varepsilon_{j,q} \in \{0, 1\} \tag{5.6}$$

As each resident $r_i \in R$ is either assigned to a single hospital or is unassigned, we introduce the following constraint for all i $(1 \leq i \leq n_1)$:

$$\sum_{p=1}^{l(r_i)+1} x_{i,p} = 1 \tag{5.7}$$

Since a hospital h_j may be assigned at most c_j residents, we constrain that $x_{i,p} = 1$ where $pref(r_i, p) = h_j$ for at most c_j residents. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : pref(r_i, p) = h_j\} \leq c_j \tag{5.8}$$

For each couple (r_{2i-1}, r_{2i}), if resident r_{2i-1} is assigned to the hospital in position p in their projected preference list then r_{2i} must also be assigned to the hospital in position p in their projected preference list. We thus obtain the following constraint for all $1 \leq i \leq c$ and $1 \leq p \leq l(r_{2i-1}) + 1$:

$$x_{2i-1,p} = x_{2i,p} \tag{5.9}$$

Stability 1 - In a stable matching M in I, if a single resident $r_i \in R$ has a worse partner than some hospital $h_j \in H$ where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$ then h_j must be fully subscribed with better partners than r_i. Therefore, either $\sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} = 0$ or h_j is fully subscribed with better partners than r_i and $\sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} = c_j$.

Thus, for each i $(2c + 1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$ and $rank(h_j, r_i) = q$:

$$c_j \sum_{p'=p+1}^{l(r_i)+1} x_{i,p'} \leq \sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\} \tag{5.10}$$

Now, for all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new constraint such that:

$$\varepsilon_{j,q} \geq \frac{\sum_{q'=q+1}^{l(h_j)} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\}}{c_j} \tag{5.11}$$

Thus, if h_j has an assignee worse than rank q then $\varepsilon_{j,q} = 1$. However, if h_j has no assignees worse than rank q then $\varepsilon_{j,q}$ may take a value of zero or one. Further, for all j $(1 \leq j \leq n_2)$ define a new constraint such that:

$$\delta_j \geq 1 - \frac{\sum_{q=1}^{l(h_j)} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q)\}}{c_j} \tag{5.12}$$

Thus, if h_j is undersubscribed then $\delta_j = 1$. However, if h_j has c_j assignees then δ_j may take the value of zero or one.

Stability 2(a) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ prefers hospital pair (h_{j_1}, h_{j_2}), at position p_1 on c_i's preference list, to $(M(r_{2i-1}), M(r_{2i}))$, at position p_2, then, if $h_{j_2} = M(r_{2i})$ then h_{j_1} cannot be undersubscribed or prefer r_{2i-1} to one of its assignees in M. In the special case in which $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_1}$, if $h_{j_1} = h_{j_2} = M(r_{2i})$ then h_{j_1} cannot be undersubscribed or prefer the poorer of r_{2i-1} and r_{2i} according to h_{j_1} to one of its partners in M.

Thus, for the general case, we obtain the following two constraints. For all i $(1 \leq i \leq c)$ and p_1, p_2 $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ such that $pref(r_{2i}, p_1) = pref(r_{2i}, p_2)$ and $rank(h_{j_1}, r_{2i-1}) = q$:

$$x_{2i,p_2} + \delta_{j_1} \leq 1 \tag{5.13}$$

$$x_{2i,p_2} + \epsilon_{j_1,q} \leq 1 \tag{5.14}$$

Intuitively, Constraint 5.13 ensures that h_{j_1} is not undersubscribed and Constraint 5.14 ensures that h_{j_1} does not prefer r_{2i} to any member of $M(h_{j_1})$.

Now, for the special case in which $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_1}$ we obtain the following two constraints. For all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i-1}))$ such that $pref(r_{2i}, p_1) = pref(r_{2i}, p_2)$ where r_{max} is the poorer of r_{2i-1} and r_{2i} according to hospital h_{j_1} with $rank(h_{j_1}, r_{max}) = q_{max}$:

$$x_{2i,p_2} + \delta_{j_1} \leq 1 \tag{5.15}$$

$$x_{2i,p_2} + \epsilon_{j_1,q_{max}} \leq 1 \tag{5.16}$$

Now, intuitively Constraint 5.15 ensures that h_{j_1} is not undersubscribed and Constraint 5.16 ensures that h_{j_1} does not prefer r_{max} to any member of $M(h_{j_1})$.

Stability 2(b) - A similar constraint is required for the odd members of each couple. Thus, for the general case, we obtain the following constraint for all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i}))$ such that $pref(r_{2i-1}, p_1) = pref(r_{2i-1}, p_2)$ and $rank(h_{j_2}, r_{2i}) = q$:

$$x_{2i,p_2} + \delta_{j_2} \leq 1 \tag{5.17}$$

$$x_{2i,p_2} + \epsilon_{j_2,q} \leq 1 \tag{5.18}$$

Intuitively Constraint 5.17 ensures that h_{j_2} is not undersubscribed and Constraint 5.18 ensures that h_{j_2} does not prefer r_{2i-1} to any member of $M(h_{j_2})$.

Again, for the special case in which $pref(r_{2i-1}, p_1) = pref(r_{2i}, p_1) = h_{j_2}$ we obtain the following two constraints. For all i $(1 \leq i \leq c)$ and p_1, p_2 where $(1 \leq p_1 < p_2 \leq l(r_{2i}))$ such that $pref(r_{2i-1}, p_1) = pref(r_{2i-1}, p_2)$ where r_{max} is the poorer of r_{2i-1} and r_{2i} according to hospital h_{j_2} with $rank(h_{j_2}, r_{max}) = q_{max}$:

$$x_{2i,p_2} + \delta_{j_2} \leq 1 \tag{5.19}$$

$$x_{2i,p_2} + \epsilon_{j_2,q_{max}} \leq 1 \tag{5.20}$$

Now, intuitively Constraint 5.19 ensures that h_{j_2} is not undersubscribed and Constraint 5.20 ensures that h_{j_2} does not prefer r_{max} to any member of $M(h_{j_2})$.

Stability 3(a) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than hospital pair (h_{j_1}, h_{j_2}) (where $h_{j_1} \neq h_{j_2}$) it must be the case that for some $t \in \{1, 2\}$, h_{j_t} is full and prefers its worst assignee to r_{2i-2+t}. Now we define the variables $\alpha_{j,q}$ such that if h_j is full with assignees better than rank q then $\alpha_{j,q}$ may take the value zero or one. However, if h_j is not full with assignees better than rank q then $\alpha_{j,q} = 1$. For all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new constraint such that:

$$\alpha_{j,q} \geq 1 - \frac{\sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\}}{c_j} \tag{5.21}$$

Thus we obtain the following constraint for all i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ where $h_{j_1} = pref(r_{2i-1}, p)$, $h_{j_2} = pref(r_{2i}, p)$, $h_{j_1} \neq h_{j_2}$, $rank(h_{j_1}, r_{2i-1}) = q_1$ and $rank(h_{j_2}, r_{2i}) = q_2$:

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \alpha_{j_1,q_1} + \alpha_{j_2,q_2} \leq 2 \tag{5.22}$$

Stability 3(b) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$ then h_j must not have two or more free posts available. For all j $(1 \leq j \leq n_2)$ define a new constraint such that:

$$\gamma_j \geq 1 - \frac{\sum_{q=1}^{l(h_j)} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q)\}}{(c_j - 1)} \tag{5.23}$$

Thus, if h_j has $c_j - 1$ or more assignees then γ_j may take the value zero or one. However, if h_j has less than $c_j - 1$ assignees then $\gamma_j = 1$. Thus we obtain the following constraint for all i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ where $\mathit{pref}((r_{2i-1}, r_{2i}), p) = (h_j, h_j)$:

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \gamma_j \leq 1 \tag{5.24}$$

Stability 3(c) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to an acceptable pair worse than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$ then h_j must not prefer both r_{2i-1} and r_{2i} to some assignee of h_j in M and also have a single free post.

For a given hospital pair (h_j, h_j) acceptable to (r_{2i-1}, r_{2i}), let r_{min} be the better of r_{2i-1} and r_{2i} according to hospital h_j with $\mathit{rank}(h_j, r_{min}) = q_{min}$. Analogously, let r_{max} be the poorer of r_{2i-1} and r_{2i} according to hospital h_j with $\mathit{rank}(h_j, r_{max}) = q_{max}$. Thus we obtain the following constraint for all i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ where $\mathit{pref}((r_{2i-1}, r_{2i}), p) = h_j$:

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \delta_j + \varepsilon_{j,q_{max}} \leq 2 \tag{5.25}$$

Stability 3(d)(i) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$ then h_j must not be undersubscribed and prefer both members of c_i to a resident who is a member of a couple (r_{2k-1}, r_{2k}) for some k $(1 \leq k \leq c, k \neq i)$ and both r_{2k-1} and r_{2k} are in $M(h_j)$. Thus we obtain the following constraint for i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ such that $\mathit{pref}((r_{2i-1}, r_{2i}), p) = (h_j, h_j)$ and $\mathit{pref}((r_{2k-1}, r_{2k}), p'') = (h_j, h_j)$ where $\max\{\mathit{rank}(h_j, r_{2i-1}), \mathit{rank}(h_j, r_{2i})\} < \max\{\mathit{rank}(h_j, r_{2k-1}), \mathit{rank}(h_j, r_{2k})\}$.

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + x_{2k-1,p''} \leq 1. \tag{5.26}$$

Stability 3(d)(ii) - In a stable matching M in I, if a couple $c_i = (r_{2i-1}, r_{2i})$ is assigned to a worse pair than (h_j, h_j) where $M(r_{2i-1}) \neq h_j$ and $M(r_{2i}) \neq h_j$ then h_j must not be undersubscribed and prefer the poorer of r_{2i-1} and r_{2i} to two distinct members of $M(h_j)$. For all j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ define a new constraint such that:

$$\beta_{j,q} \geq 1 - \frac{\sum_{q'=1}^{q-1} \{x_{i',p''} \in X : (r_{i'}, p'') \in R(h_j, q')\}}{(c_j - 1)} \tag{5.27}$$

Thus, if h_j has $c_j - 2$ or fewer assignees better than rank q then $\beta_{j,q} = 1$. However, if h_j has more than $c_j - 2$ assignees better than rank q then $\beta_{j,q}$ may take a value of zero or one. For a given hospital pair (h_j, h_j) acceptable to (r_{2i-1}, r_{2i}), let r_{max} be the poorer of r_{2i-1} and r_{2i} according to hospital h_j with $rank(h_j, r_{max}) = q_{max}$. Thus we obtain the following constraint for i $(1 \leq i \leq c)$ and p $(1 \leq p \leq l(r_{2i-1}))$ such that $pref((r_{2i-1}, r_{2i}), p) = (h_j, h_j)$.

$$\sum_{p'=p+1}^{l(r_{2i-1})+1} x_{2i-1,p'} + \beta_{j,q_{max}} \leq 1. \tag{5.28}$$

Objective Function - A maximum cardinality stable matching M in I is a stable matching in which the maximum number of residents are assigned taken over all of the stable matchings admitted by I. To maximise the size of the stable matching found we apply the following objective function:

$$\max \sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} x_{i,p} \tag{5.29}$$

Chapter 6

Empirical results from the IP models for HRC

6.1 Introduction

In this chapter we present data from an empirical evaluation of an implementation of the IP models for HRC described in Chapters 4 and 5 for finding a maximum cardinality stable matching in arbitrary instances of HRC or reporting that no stable matching exists. We focus on several properties, including the maximum and mean number of stable matchings admitted by an instance; the time taken to find a maximum cardinality stable matching or report that no stable matching exists; the size of a maximum cardinality stable matching admitted by an instance; and the number of stable matchings admitted by an instance. We show how these properties vary as we modify a range of parameters in the constructed instances, including the number of residents in the instance; the percentage of the residents involved in couples; the number of hospitals in the instance; and the lengths of the residents' preference lists. Further, we present data obtained from the application of the IP model for HRCT to real world instances arising from the SFAS application for the years 2010, 2011 and 2012.

In Section 6.2 we present an overview of the experiments performed in this chapter. In Section 6.2.1 we give details of the computational environment in which the experiments were performed. Further, in Section 6.2.2 we describe details of the testing applied to the implementation to attempt to increase confidence in the correctness of the implementation. When creating random instances to solve in our experiments we sought to ensure that the instances created reflected the properties of the instances arising in the SFAS application.

In Sections 6.2.3 and 6.2.4 we describe some of the properties of the SFAS instances that were reflected in the properties of the random instances solved in these experiments. In the SFAS application the individual residents forming part of a couple were each asked to express their own individual preference list over single hospitals. The process by which the preference lists for the couples were derived from these individual lists is described in detail in Section 6.2.3. Further in Section 6.2.4 we consider how best to distribute hospitals and residents amongst each others' preference lists by considering the notion of the popularity of a given hospital to a given resident (or vice versa). Finally in Section 6.2.5 we show how the preference lists of the agents may be reduced by identifying pairs who must be assigned in any stable matching.

In Section 6.3 we present data obtained from applying the IP models described in Sections 4.3 and 5.4 to randomly generated instances of HRC reflecting the properties of the SFAS application. We applied the IP models to 1000 randomly generated instances following the experimental structure used by Biro et al./ in [12].

Finally, in Section 6.4 we examine the output when the IP model for HRCT described in Section 4.5 is applied to real world instances arising from the SFAS application in the years 2010-2012.

6.2 Overview of HRC experiments

In the experiments in Section 6.3 we examine the output of the model as we vary the parameters of the instance under both MM-stability and BIS-stability. We applied the model to randomly generated instances reflecting the properties of the instances arising in the SFAS context and we now present data on the following outputs from the model as we vary the size of the instance, the percentage of the residents involved in couples, the number of hospitals in the instance and the length of the residents' preference lists.

1. the maximum and mean number of stable matchings admitted by an instance;
2. the time taken to find maximum cardinality stable matchings or report that no stable matching exists;
3. the size of a maximum cardinality stable matching admitted by an instance;
4. the number of stable matchings admitted by an instance;

6.2.1 Computational environment for HRC experiments

We ran experiments on a Java implementation of the IP models for HRC under MM-stability and BIS-stability as described in Sections 4.3 and 5.4, applied to randomly-generated instances of HRC reflecting the properties of the instances arising in the SFAS application. All experiments were carried out on a desktop PC with an Intel i5-2400 3.1Ghz processor, with 8Gb of memory running Windows 7. The IP solver used in all cases was CPLEX 12.4 and the models were implemented in Java using CPLEX Concert.

6.2.2 Correctness testing of the implemented model

We implemented stability checkers for HRC under both MM-stability and BIS-stabilility in Java. Every solution output by any of the IP models was tested for stability using these stability checkers. In all cases the solutions output by the IP models were found to be stable by the corresponding stability checker, i.e. no unstable matching was ever found by any model under the appropriate stability definition.

To further test that our implementations correctly output a maximum cardinality stable matching according to the implemented stability checker we used a brute force algorithm. The algorithm recursively generated all feasible matchings admitted by an instance of HRC and selected from amongst the feasible matchings found to be stable by our stability checker a maximum cardinality stable matching, or reported that no stable matching existed if the stability checker found no stable matching. (Due to the inefficiency of this brute force algorithm it may only be realistically applied to relatively small instances.) When solving hundreds of thousands of HRC instances with random properties involving up to fifteen residents, our implementation agreed with the brute force algorithm when reporting whether the instance admitted a stable matching or not. Further, our implementation returned a stable matching of the same size as a maximum cardinality stable matching output by the brute force algorithm in all cases.

6.2.3 Projection of preference lists to mimic the format of the SFAS instances

In the SFAS application, the joint preference list for a couple (r_i, r_j) was derived from the submitted preference lists of the individual residents r_i and r_j. In order to reflect as accurately as possible the properties of the instances arising in the SFAS application we constructed the joint preference lists of the couples in the randomly generated instances in the experiments in the same fashion. Thus, the joint preference lists of the couples in the randomly generated instances were constructed from the preferences of the two individual residents involved in the couple as follows. For a couple (r_i, r_j), let s (respectively t) be the length of the individual preference list of r_i (respectively r_j). For all the instances solved in these experiments, the minimum length of preference list for an individual resident is five and the maximum length is ten.

Now, let a and b be two integers such that $1 \leq a \leq s$ and $1 \leq b \leq t$. The rank pair (a, b) represents the a^{th} hospital on resident r_i's individual preference list and the b^{th} hospital on resident r_j's preference list and s and t represent the length of the individual preference list of a and b respectively. Couple (r_i, r_j) finds acceptable all pairs (h_p, h_q) where r_i finds h_p acceptable and r_j finds h_q acceptable (st pairs in total). These pairs were ordered as follows. Let $L = \max\{s, t\}$. Corresponding to every such acceptable pair (h_p, h_q), create an L-tuple whose i^{th} entry is the number of residents in the couple who obtain their i^{th} choice (when considering their individual lists) in the pair (h_p, h_q). The acceptable pairs on the couple's list are then listed according to a lexicographically increasing order on the reverse of the corresponding L-tuples.

Observe that a lexicographically increasing order on the reverse of the L-tuples is not the same as a lexicographically decreasing order on the L-tuples as demonstrated by the following example. Let r_1 and r_2 be two individual residents who are part of the couple (r_1, r_2). We now demonstrate the process by which the preference lists of are projected to form the joint preference list of (r_1, r_2). Suppose that the residents' individual preferences are as follows:

$$\begin{array}{lllll} r_1 & : & h_3 & h_2 & h_1 \\ r_2 & : & h_2 & h_1 & h_3 \end{array}$$

We construct the preference list for the couple (r_1, r_2) by creating a sequence of (r_1, r_2)'s acceptable pairs as follows. The pair (h_3, h_2) has profile $(2, 0, 0)$ since h_3 is first in r_1's preference list and h_2 is first in r_2's preference list. Since no other hospital pair shares this profile then (h_3, h_2) must be first in the projected joint preference list of (r_1, r_2). Next we consider the two pairs (h_2, h_2) and (h_3, h_1) both of which have profile $(1, 1, 0)$ since each pair has exactly one member in first place and exactly one member in second place on the residents' preference lists. Since these two profiles cannot be strictly ordered lexicographically, we break this tie by randomly selecting one of these pairs to appear before the other in the couples' joint preference list – in this case we place (h_2, h_2) before (h_3, h_1).

Since we wish the pairs to be ordered in a lexicographically increasing order on the reverse of the L-tuples we now consider pairs with profile $(0, 2, 0)$ before those with profile $(1, 0, 1)$. This is different from a lexicographically decreasing order on the profiles in which we would consider tuples with profile $(1, 0, 1)$ before $(0, 2, 0)$. Thus we consider the pair (h_2, h_1) which has the profile $(0, 2, 0)$. Since this is the only pair that has this profile, we place (h_2, h_1) next in the couples' joint preference list. We continue this process, breaking ties where necessary to generate the joint preference list for (r_1, r_2) as follows. The preference lists of the couples in the randomly generated instances in the experiments that follow are constructed in a similar fashion.

$$\begin{array}{lllllllllll} (r_1, r_2) & : & (h_3, h_2) & (h_2, h_2) & (h_3, h_1) & (h_2, h_1) & (h_3, h_3) & (h_1, h_2) & (h_1, h_1) & (h_2, h_3) & (h_1, h_3) \\ & & (2,0,0) & (1,1,0) & (1,1,0) & (0,2,0) & (1,0,1) & (1,0,1) & (0,1,1) & (0,1,1) & (0,0,2) \end{array}$$

6.2.4 The popularity of hospitals

In the data from the SFAS application it is clear that some hospitals receive far more applications than others. We consider that between two hospitals, the one receiving the most applications is the more *popular* hospital. Clearly in most allocation processes hospitals are not of uniform popularity. e.g. hospitals in the major urban areas are more popular than those in more rural areas. So, how best do we distribute hospitals and residents amongst each others' preference lists to reflect this notion of popularity?

Typically, the most popular hospital in the SFAS context would have five to six times as many applicants as the least popular, and the numbers of applicants to the other hospitals were fairly uniformly distributed between the two extremes [38]. With this in mind the instances in these experiments were generated with a skewed preference list distribution on both sides. The process by which the preference lists were generated is explained in detail below.

To generate the preference lists for the residents involved in an instance we use the following process. Let n_2 be the number of hospitals in the instance generated and let x be the hospital popularity ratio. Firstly, the hospitals are assumed to be numbered in popularity order - say Hospital n_2 is the most popular and Hospital 1 the least popular. Suppose a resident is to have a preference list of length p. Then we generate a sequence of p distinct hospitals, and at each step in this process, Hospital n_2 is x times more likely to be chosen than Hospital 1, with the likelihood of the intermediate hospitals being interpolated linearly between the two extremes.

To generate the preference list of a given resident we set up an array of size $n_2(n_2 - 1)(x + 1)/2$. Into this array are placed: $n_2 - 1$ copies of Hospital 1; $n_2 - 1 + (x - 1)$ copies of Hospital 2; $n_2 - 1 + 2(x - 1)$ copies of Hospital 3; $\ldots n_2 - 1 + (n_2 - 1)(x - 1)$ copies of Hospital n_2. To construct an applicants preference list we repeatedly generate a random number s $(0 \leq s < n_2(n_2 - 1)(x + 1)/2)$ and add the hospital h_j $(1 \leq j \leq n_2)$ at position s in the array to the resident's preference list. Subsequently all copies of h_j are removed from the array to ensure that the same hospital may not be chosen twice in an individual resident's preference list. A resident's preference list of length p is formed by selecting p hospitals in this fashion.

To generate the preference lists for the hospitals involved in an instance an adapted version of this technique is applied. Let n_1 be the number of residents, let c be the number of couples and let y be the resident popularity ratio in the instance generated. If we were to assume that the residents are numbered in popularity order where Resident n_1 was the most popular and Resident 1 was the least popular, then by applying the above mechanism Resident n_1 would be more likely to be chosen as the next resident in a hospital's preference list. However, as the instance generator assumes that the first $2c$ residents are involved in couples then this would imply that those residents involved in couples would be less likely to be chosen as the next resident in an arbitrary hospital's preference list as residents later in the list (who are more likely to be single) are more popular.

We used the following adaptation to address this problem. For each instance generated we construct a randomly ordered list of the n_1 distinct residents. We further assume that the residents in this list are in popularity order. Thus the n_1^{th} resident in the list is the most popular and the first resident in the list is the least popular. Suppose q residents have expressed a preference for a hospital. Then the program generates a sequence of the q acceptable residents, and at each step in this process, the resident in position n_1 in the randomly ordered list of residents is y times more likely to be chosen than the resident in position one, with the likelihood of the intermediate residents being interpolated linearly between the two extremes. Since the residents involved in couples are randomly positioned in the array, a resident involved in a couple is no more or less likely to be highly popular than a single resident.

To generate the preference list of a given hospital we set up an array of size $n_1(n_1 - 1)(y + 1)/2$. Into this array are placed: $n_1 - 1$ copies of the resident in first position in the randomly ordered list; $n_1 - 1 + (y - 1)$ copies of the resident in second position in the randomly ordered list; $n_1 - 1 + 2(y - 1)$ copies of the resident in third position in the randomly ordered list; $\ldots n_1 - 1 + (n_1 - 1)(y - 1)$ copies of the resident in n_1^{th} position in the randomly ordered list. For each hospital h_j we remove from the array all positions containing residents that h_j finds unacceptable. Let l be the length of the array which now contains only acceptable residents.

To construct a hospital's preference list, repeatedly generate a random number t $(0 \leq t < l)$ and add the resident at position t in the array to the resident's preference list. Subsequently all copies of the resident at position t are removed from the array to ensure that the same resident may not be chosen twice in a hospital's preference list.

6.2.5 Pre-processing of instances before sending to the solver

Before constructing the model derived from a given instance of HRC we may reduce the instance by removing some entries from the preference lists. We may remove all *fixed assignments* as described in Lemma 3.7.1 in Section 3.7. A fixed assignment in an instance I of HRC is a pair of agents that must be assigned to each other in any stable matching in I. We may iteratively satisfy all fixed assignments in I in linear time, thus reducing the size of the problem that must be delivered to the solver.

6.3 Experimental results with randomly generated instances

6.3.1 HRC Experiment 1

In this first experiment, we report on data obtained as we increased the number of residents while maintaining a constant ratio of couples, hospitals and posts to residents. For various values of x $(100 \leq x \leq 1000)$ in increments of 30, 1000 randomly generated instances were created containing x residents, $0.1x$ couples and $0.1x$ hospitals with x available posts randomly distributed amongst the hospitals. The maximum and mean number of stable matchings admitted by the instances is plotted in Figure 6.1 for all values of x. Figure 6.2 shows the mean size of a maximum cardinality stable matching for all values of x. The mean time taken to find a maximum cardinality stable matching or report that no stable matching existed in each instance is plotted in Figure 6.3 for all values of x. Figure 6.4 displays the percentage of instances encountered that admitted a stable matching.

Figure 6.1 shows that the largest number of stable matchings admitted by the HRC instances did not appear to be correlated with the number of residents involved in the instance. Figure 6.2 also shows that as the number of residents in the instances increased, the mean size of a maximum cardinality stable matching in the instances increased. The data in Figure 6.3 shows that the mean time taken to find a maximum cardinality stable matching or report that no stable matching existed increased as we increased the number of residents in the instance. Figure 6.4 also shows that the percentage of HRC instances admitting a stable matching did not appear to be correlated with the number of residents involved in the instance.

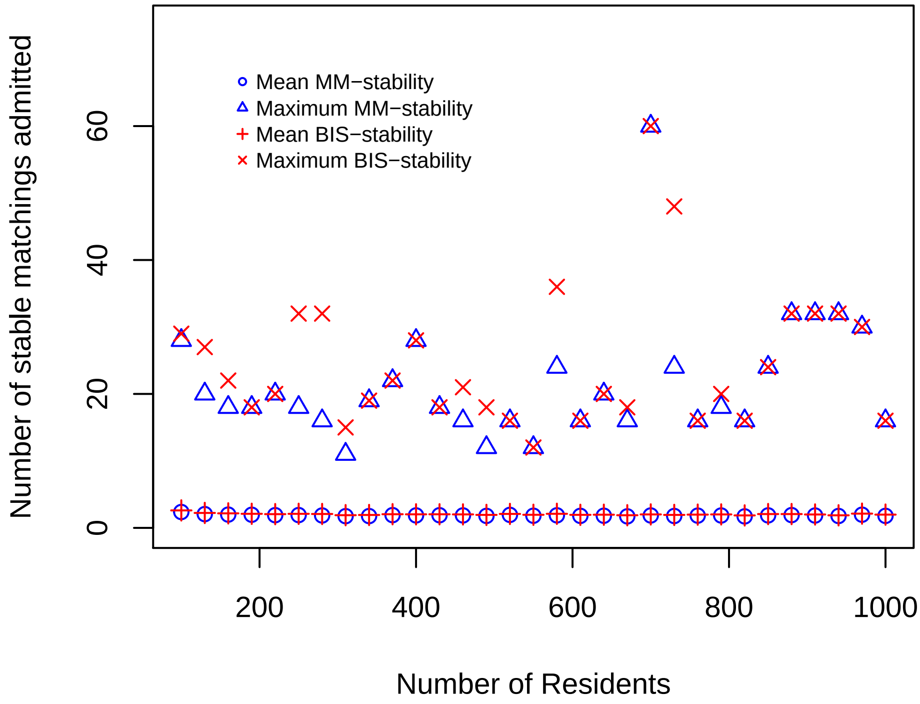

Figure 6.1: HRC Experiment 1 - Number of stable matchings admitted by random instances.

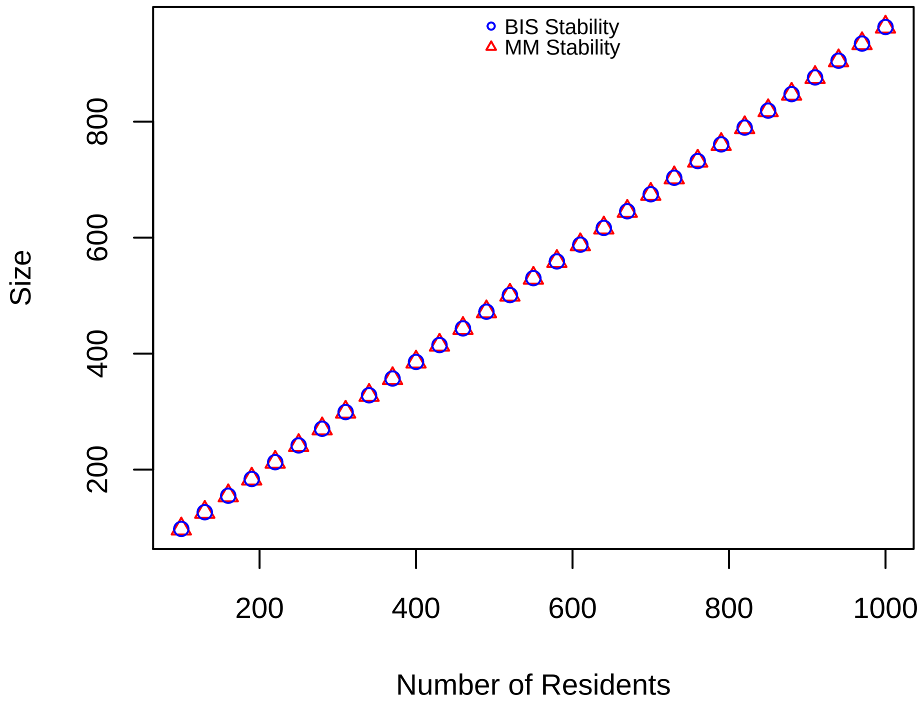

Figure 6.2: HRC Experiment 1 - Mean size of a maximum cardinality stable matching.

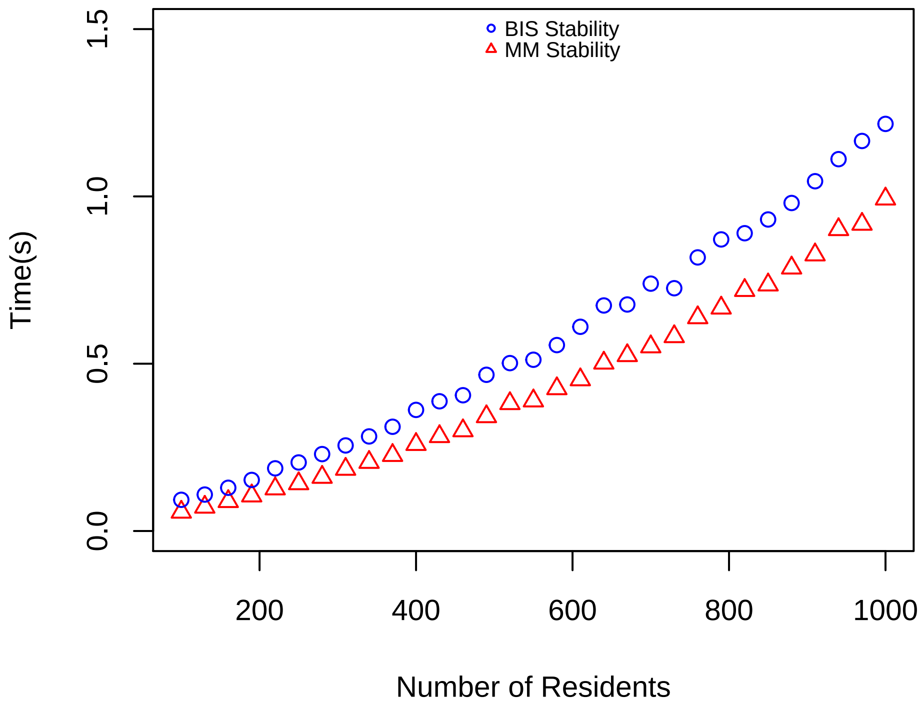

Figure 6.3: HRC Experiment 1 - Mean time to solve to optimality.

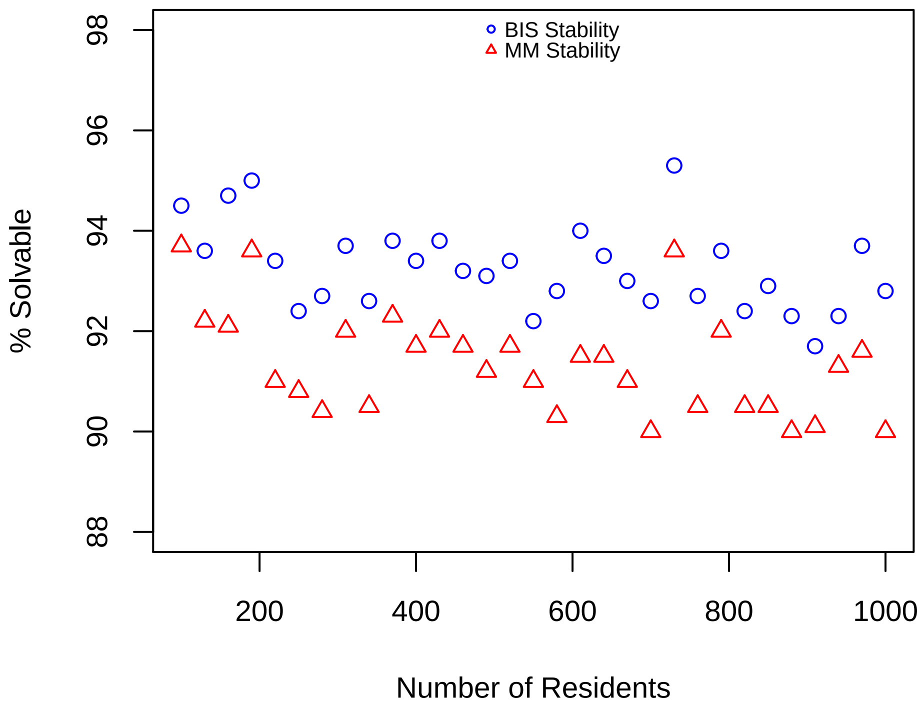

Figure 6.4: HRC Experiment 1 - Percentage of instances admitting a stable matching.

Residents	Mean Number of Stable Matchings		Max Number of Stable Matchings		Mean Time to Optimal Solution		Number of Solvable Instances		Mean Size of a Max Stable Matching		Mean Number of Variables (x1000)		Mean Number of Constraints (x1000)	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
100	2.37	2.61	28	29	0.06	0.09	937	945	97.79	97.79	1.04	1.08	3.66	6.35
130	2.06	2.23	20	27	0.07	0.11	922	936	126.54	126.55	1.31	1.36	4.55	7.86
160	1.99	2.17	18	22	0.09	0.13	921	947	155.03	155.01	1.58	1.64	5.41	9.32
190	1.96	2.10	18	18	0.11	0.15	936	950	183.88	183.87	1.86	1.94	6.36	10.92
220	1.90	2.05	20	20	0.13	0.19	910	934	212.85	212.86	2.14	2.23	7.30	12.53
250	1.92	2.10	18	32	0.14	0.20	908	924	241.76	241.75	2.42	2.52	8.21	14.08
280	1.85	2.08	16	32	0.16	0.23	904	927	270.44	270.46	2.71	2.82	9.17	15.71
310	1.75	1.90	11	15	0.18	0.26	920	937	299.28	299.28	2.99	3.11	10.12	17.33
340	1.76	1.93	19	19	0.21	0.28	905	926	328.24	328.24	3.26	3.40	11.03	18.88
370	1.90	2.04	22	22	0.23	0.31	923	938	357.13	357.11	3.55	3.69	11.97	20.48
400	1.86	2.02	28	28	0.26	0.36	917	934	385.65	385.69	3.84	4.00	12.99	22.24
430	1.89	2.03	18	18	0.28	0.39	920	938	414.82	414.82	4.13	4.30	13.93	23.83
460	1.86	2.01	16	21	0.30	0.41	917	932	443.57	443.56	4.42	4.60	14.91	25.50
490	1.79	1.94	12	18	0.34	0.47	912	931	472.29	472.31	4.69	4.88	15.81	27.04
520	1.94	2.08	16	16	0.38	0.50	917	934	501.01	501.03	4.97	5.18	16.73	28.60
550	1.81	1.96	12	12	0.39	0.51	910	922	530.23	530.20	5.24	5.46	17.61	30.11
580	1.87	2.10	24	36	0.43	0.56	903	928	559.11	559.06	5.54	5.77	18.66	31.90
610	1.79	1.95	16	16	0.45	0.61	915	940	587.93	587.92	5.82	6.07	19.59	33.50
640	1.81	1.97	20	20	0.50	0.67	915	935	616.80	616.79	6.10	6.36	20.53	35.09
670	1.72	1.89	16	18	0.52	0.68	910	930	645.53	645.53	6.38	6.65	21.45	36.67
700	1.85	2.01	60	60	0.55	0.74	900	926	674.84	674.85	6.70	6.98	22.56	38.58
730	1.77	1.94	24	48	0.58	0.73	936	953	703.26	703.29	6.92	7.22	23.24	39.71
760	1.83	1.98	16	16	0.64	0.82	905	927	732.08	732.10	7.25	7.55	24.36	41.63
790	1.85	2.01	18	20	0.67	0.87	920	936	760.86	760.87	7.54	7.85	25.36	43.35
820	1.71	1.86	16	16	0.72	0.89	905	924	789.80	789.77	7.79	8.12	26.17	44.71
850	1.87	2.06	24	24	0.74	0.93	905	929	819.09	819.07	8.11	8.45	27.26	46.58
880	1.91	2.06	32	32	0.79	0.98	900	923	847.49	847.51	8.36	8.71	28.04	47.91
910	1.85	2.00	32	32	0.83	1.05	901	917	876.29	876.28	8.66	9.02	29.06	49.65
940	1.77	1.88	32	32	0.90	1.11	913	923	905.12	905.12	8.96	9.34	30.15	51.52
970	1.97	2.13	30	30	0.92	1.17	916	937	934.63	934.65	9.25	9.64	31.10	53.15
1000	1.79	1.98	16	16	0.99	1.22	900	928	963.24	963.26	9.51	9.91	31.94	54.58

Table 6.1: HRC Experiment 1 - Tabulation of output from HRC Experiment 1.

6.3.2 HRC Experiment 2

In our second experiment, we report on results obtained as we increased the percentage of residents involved in couples while maintaining the same total number of residents, hospitals and posts. For various values of x $(0 \leq x \leq 200)$ in increments of 25, 1000 randomly generated instances were created containing 1000 residents, x couples (and hence $1000 - 2x$ single residents) and 100 hospitals with 1000 available posts that were randomly distributed amongst the hospitals. The maximum and mean number of stable matchings admitted by the instances is plotted in Figure 6.5 for all values of x. Figure 6.6 displays the mean size of a maximum cardinality stable matching in the instances for all values of x. The mean time taken to find a maximum cardinality stable matching or report that no stable matching existed in each instance is plotted in Figure 6.7 for all values of x. Figure 6.8 displays the percentage of instances encountered admitting a stable matching.

Figure 6.5 shows that there does not appear to be a strong correlation between the maximum number of stable matchings admitted in the instances and the number of couples in the instances. However, the mean number of stable matchings admitted across the instances appears to be broadly the same regardless of the number of couples in the instance.

The data in Figure 6.6 shows that as the percentage of the residents involved in couples in the instances increased the mean size of a maximum cardinality stable matching admitted by the instances tended to decrease under both MM-stability and BIS-stability. Further, the mean size of a maximum cardinality stable matching admitted by the instances is very similar under both stability definitions. We conjecture that, under both MM-stability and BIS-stability, as the number of couples increases while the number of residents remains the same, the problem becomes more tightly constrained as more constraints act on the couples in the instance than on the single residents. Thus, there is a reduced likelihood of a given matching being stable and hence the size of a maximum cardinality stable matching might be smaller as shown in Figure 6.6.

The data in Figure 6.7 shows that the mean time taken to find a maximum cardinality stable matching tended to increase as we increased the number of residents in the instances involved in couples. The time taken to find a maximum cardinality stable matching in the instances under each stability definition in instances with less than 150 couples is very similar. However as we reach 175 couples and above the mean time taken to find a maximum cardinality stable matching increases more quickly under BIS-stability than under MM-stability. We conjecture that as the number of couples in the instance increases and thus the number of linked residents for whom a preference list must be projected increases (where a projected preference list is necessarily of greater length than a single resident's preference list), the time taken to find a maximum cardinality stable matching increases.

Further, Figure 6.8 shows that the percentage of HRC instances admitting a stable matching

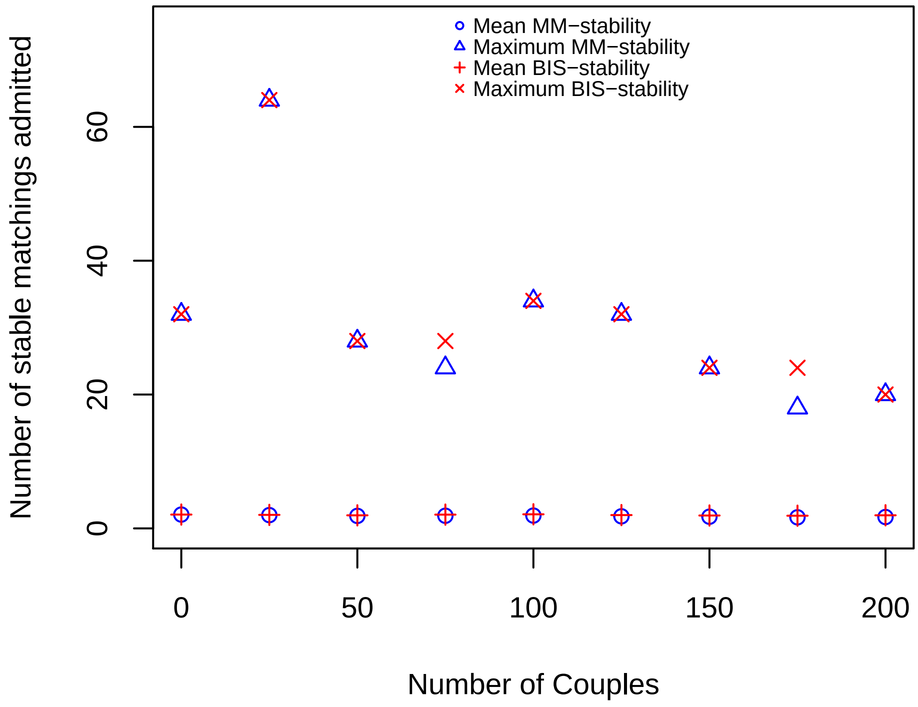

Figure 6.5: HRC Experiment 2 - Number of stable matchings admitted by random instances.

under both stability definitions fell as the percentage of the residents in the instances involved in couples increased under both stability definitions. However, the percentage of instances admitting a stable matching appears to be consistently larger under BIS-stability than under MM-stability for all instance with more than zero couples. As in the discussion of Figure 6.6, we conjecture that, under both MM-stability and BIS-stability, as the number of couples increases while the number of residents remains the same, the problem becomes more tightly constrained as more constraints act on the couples in the instance than on single residents. Thus, there is a reduced likelihood of a given matching being stable.

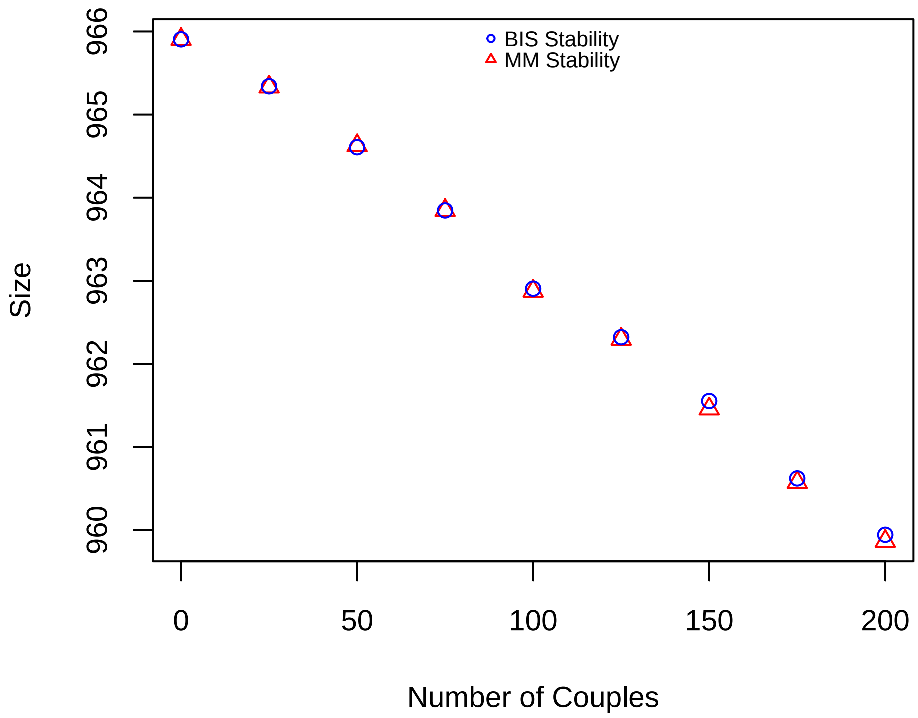

Figure 6.6: HRC Experiment 2 - Mean size of a maximum cardinality stable matching.

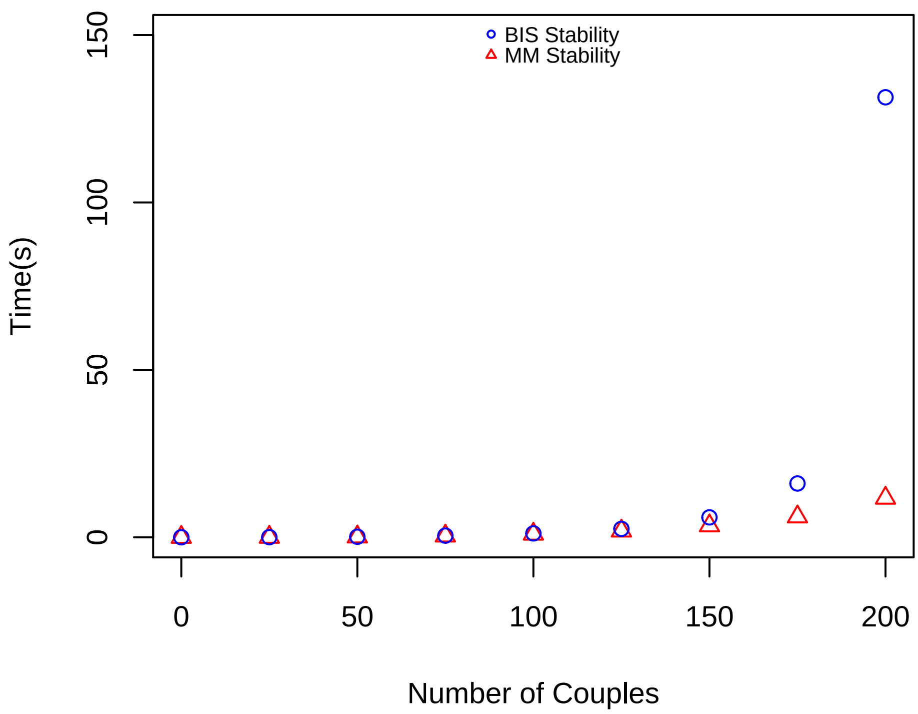

Figure 6.7: HRC Experiment 2 - Mean time to solve to optimality.

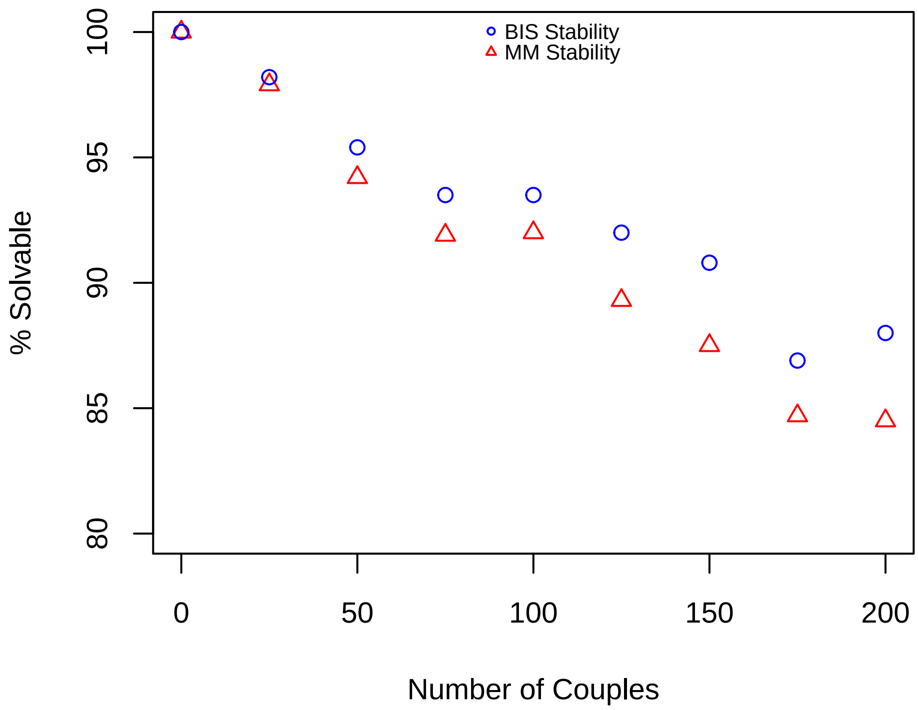

Figure 6.8: HRC Experiment 2 - Percentage of instances admitting a stable matching.

Couples	Mean Number of Stable Matchings		Max Number of Stable Matchings		Mean Time to Optimal Solution		Number of Solvable Instances		Mean Size of a Max Stable Matching		Mean Number of Variables (x1000)		Mean Number of Constraints (x1000)	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
0	2.07	2.07	32	32	0.00	0.01	1000	1000	965.90	965.90	2.17	2.57	4.01	6.14
25	1.98	2.02	64	64	0.02	0.04	979	982	965.33	965.34	3.38	3.78	7.95	12.81
50	1.86	1.95	28	28	0.13	0.16	942	954	964.63	964.61	5.01	5.41	13.82	22.87
75	1.89	2.06	24	28	0.40	0.49	919	935	963.85	963.85	7.08	7.48	21.83	36.82
100	1.91	2.10	34	34	0.96	1.16	920	935	962.88	962.90	9.51	9.91	31.95	54.59
125	1.80	1.99	32	32	1.89	2.51	893	920	962.30	962.32	12.20	12.60	43.66	75.30
150	1.75	1.92	24	24	3.44	5.94	875	908	961.46	961.55	15.12	15.52	56.92	98.90
175	1.65	1.90	18	24	6.11	16.06	847	869	960.57	960.62	18.23	18.63	71.58	125.10
200	1.70	1.96	20	20	11.70	131.41	845	880	959.87	959.94	21.37	21.77	86.66	152.09

Table 6.2: HRC Experiment 2 - Tabulation of output from HRC Experiment 2.

6.3.3 HRC Experiment 3

In our third experiment, we report on data obtained as we increased the number of hospitals in the instance while maintaining the same total number of residents, couples and posts. For various values of x ($50 \leq x \leq 500$) in increments of 25, 1000 randomly generated instances were created consisting of 1000 residents in total, x hospitals, 100 couples (and hence 800 single residents) and 1000 available posts that were randomly distributed amongst the hospitals. The maximum and mean number of stable matchings admitted by the instances is plotted in Figure 6.9 for all values of x. The mean size of a maximum cardinality stable matching for all values of x is displayed in Figure 6.10. The time taken to find a maximum cardinality stable matching or report that no stable matching existed in each instance is plotted in Figure 6.11 for all values of x. Figure 6.12 charts the percentage of instances encountered admitting a stable matching.

Figure 6.9 shows that there did not appear to be a strong correlation between the maximum number of stable matchings admitted by the instances and the number of hospitals in the instances. However, the mean number of stable matchings admitted across the instances appeared to be broadly the same regardless of the number of hospitals in the instance.

The data in Figure 6.10 shows that as the number of hospitals in the instances increased, the mean size of a maximum cardinality stable matching in the instances under both stability definitions tended to decrease. This can be explained by the fact that, as the number of hospitals increased but the residents' preference list lengths and the total number of posts remained constant, the number of posts per hospital decreased. Hence the total number of posts among all hospitals on a resident's preference list decreases and a hospital is less likely to be available to be assigned a given resident.

Figure 6.11 shows that the mean time taken to find a maximum cardinality stable matching tended to decrease under both stability definitions as we increased the number of hospitals in the instances. We conjecture that this is because as the number of hospitals increases while the residents' preference list lengths and the total number of posts remain constant, the length of a hospital's preference list is likely to be shorter. Since every position in a hospital's preference list is constrained with every other position on that hospital's preference list, the number of positions constrained with each other on a given hospital's preference list decreases with the length of the hospital's preference list. Thus, as the hospitals' preference lists become shorter, the model's complexity reduces and the time taken to find a maximum cardinality stable matching reduces. Finding a maximum cardinality stable matching under BIS-stability took longer than finding a maximum cardinality stable matching under MM-stability in all cases, with the difference being more marked when the number of hospitals in the instance was smaller.

The data in Figure 6.12 also shows that the percentage of HRC instances admitting a stable

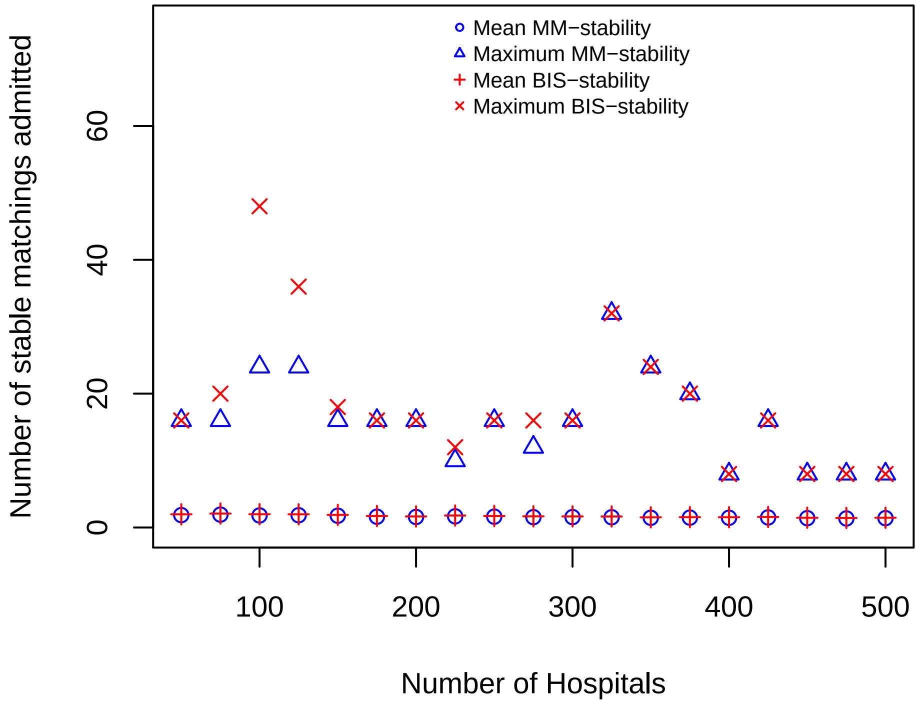

Figure 6.9: HRC Experiment 3 - Number of stable matchings admitted by random instances.

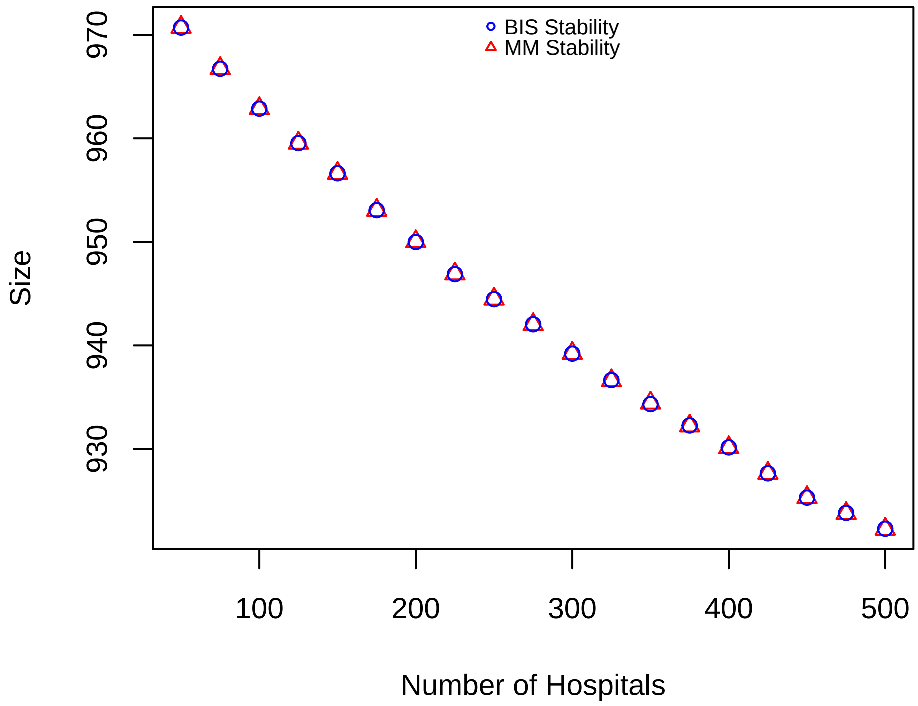

Figure 6.10: HRC Experiment 3 - Mean size of a maximum cardinality stable matching.

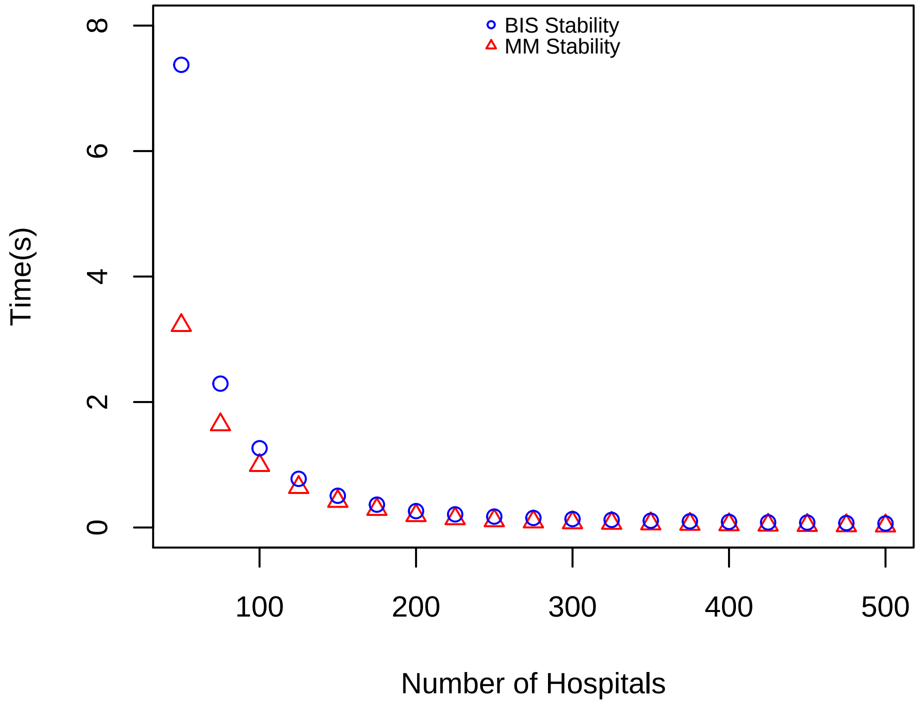

Figure 6.11: HRC Experiment 3 - Mean time to solve to optimality.

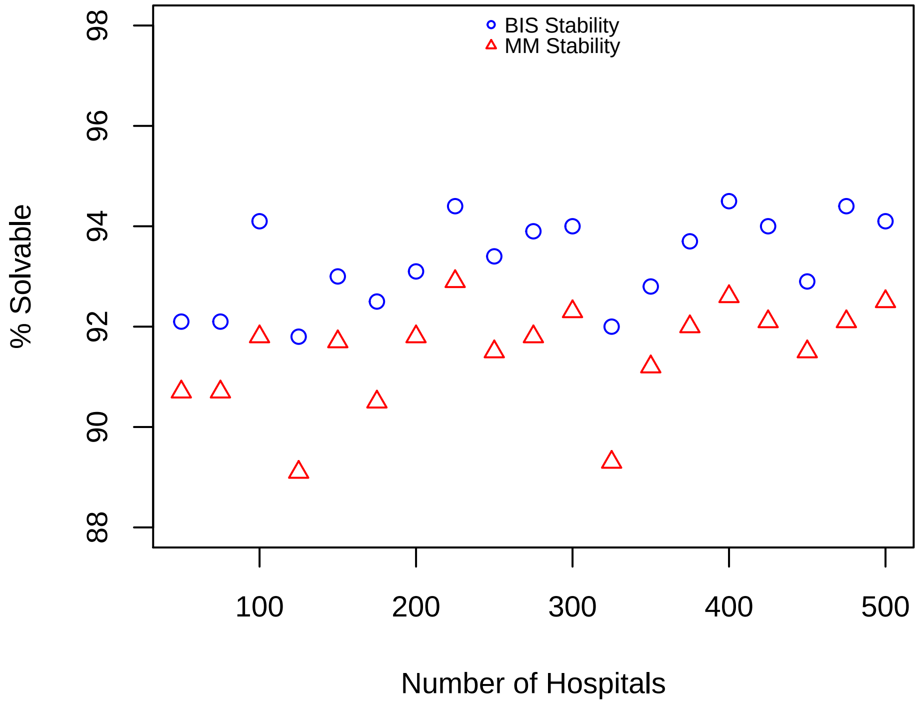

Figure 6.12: HRC Experiment 3 - Percentage of instances admitting a stable matching.

matching under each stability definition did not appear to alter in a significant fashion with the number of hospitals involved in the instance. We conjecture that two effects may influence this outcome. As the number of hospitals increase, each hospital has a smaller number of posts and is thus more likely to become full and hence less likely to be involved in a blocking pair due to being undersubscribed; however, this also has the effect that each hospital is less likely to be available to be assigned a given resident to form a matching. Since there is little change in the number of instances admitting a stable matching across the instances, these two effects seem to offset each other in these experiments. It is however noticeable that the percentage of instances admitting a stable matching was greater under BIS-stability than under MM-stability in all cases.

Hospitals	Mean Number of Stable Matchings		Max Number of Stable Matchings		Mean Time to Optimal Solution		Number of Solvable Instances		Mean Size of a Max Stable Matching		Mean Number of Variables (x1000)		Mean Number of Constraints (x1000)	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
50	1.85	1.98	16	16	3.22	7.37	907	921	970.73	970.70	10.68	10.88	38.06	65.56
75	1.92	2.08	16	20	1.64	2.29	907	921	966.75	966.72	10.03	10.33	34.62	59.36
100	1.80	1.99	24	48	0.99	1.26	918	941	962.88	962.87	9.50	9.90	31.89	54.48
125	1.83	1.98	24	36	0.64	0.78	891	918	959.54	959.54	9.02	9.52	29.44	50.13
150	1.76	1.88	16	18	0.42	0.51	917	930	956.63	956.62	8.65	9.25	27.57	46.83
175	1.61	1.72	16	16	0.29	0.36	905	925	953.06	953.06	8.30	9.00	25.80	43.71
200	1.57	1.66	16	16	0.19	0.26	918	931	950.03	949.99	7.96	8.76	24.09	40.71
225	1.67	1.79	10	12	0.14	0.21	929	944	946.92	946.88	7.73	8.63	22.91	38.64
250	1.62	1.72	16	16	0.11	0.17	915	934	944.48	944.46	7.50	8.50	21.75	36.59
275	1.55	1.67	12	16	0.09	0.15	918	939	942.02	942.04	7.33	8.43	20.81	34.93
300	1.56	1.66	16	16	0.08	0.13	923	940	939.24	939.21	7.15	8.35	19.90	33.32
325	1.53	1.65	32	32	0.07	0.12	893	920	936.59	936.66	7.02	8.32	19.18	32.03
350	1.45	1.53	24	24	0.06	0.11	912	928	934.44	934.33	6.87	8.27	18.35	30.56
375	1.48	1.55	20	20	0.05	0.10	920	937	932.23	932.27	6.78	8.28	17.77	29.50
400	1.46	1.54	8	8	0.05	0.09	926	945	930.13	930.15	6.70	8.30	17.22	28.51
425	1.50	1.58	16	16	0.04	0.08	921	940	927.66	927.64	6.61	8.31	16.68	27.50
450	1.40	1.45	8	8	0.03	0.08	915	929	925.31	925.30	6.54	8.34	16.16	26.53
475	1.35	1.42	8	8	0.03	0.07	921	944	923.77	923.83	6.50	8.40	15.79	25.81
500	1.40	1.46	8	8	0.03	0.06	925	941	922.26	922.30	6.46	8.46	15.42	25.11

Table 6.3: HRC Experiment 3 - Tabulation of output from HRC Experiment 3.

6.3.4 HRC Experiment 4

In our last experiment in this section we report on data obtained as we increased the length of the individual preference lists for the residents in the instance while maintaining the same total number of residents, couples, hospitals and posts. For various values of x $(3 \leq x \leq 11)$ in increments of one, 1000 randomly generated instances were created consisting of 1000 residents in total, 100 hospitals, 100 couples (and hence 800 single residents) and 1000 available posts that were randomly and randomly distributed amongst the hospitals. The maximum and mean number of stable matchings admitted by the instances are plotted in Figure 6.13 for all values of x. Figure 6.14 displays the mean size of a maximum cardinality stable matching for all values of x. The mean time taken to find a maximum cardinality stable matching or report that no stable matching existed in each instance is plotted in Figure 6.15 for all values of x. Figure 6.16 displays the percentage of instances encountered admitting a stable matching.

Figure 6.13 shows that the maximum number of stable matchings admitted across the instances increased as we increased the length of the individual residents' preference lists in the instances. The mean number of stable matchings admitted under both stability definitions also increased slightly. This can perhaps be explained by the fact that as the length of the individual residents' preference lists increased, a resident becomes more likely to find acceptable some hospital that has an available post and so becomes more likely to be able to be assigned.

The data in Figure 6.14 shows that the length of the individual residents' preference lists in the instances increased, the mean size of a maximum cardinality stable matching in the instances under both stability definitions tends to increase. We conjecture that as the length of the individual residents' preference lists increase, a resident becomes more likely to find acceptable a hospital that has an available post and so is more likely to be able to be assigned. However, the rate of increase of this effect in terms of producing larger matchings appears to slow as we approach higher values for the length of the residents' preference lists in the instance. This can be explained by the fact that, as the length of the individual residents' preference lists increase at these higher values, the hospitals added to the end of the residents' preference list become less decisive in determining the size of the resulting maximum cardinality stable matching and so their effect on the size of a maximum cardinality stable matching reduces.

Figure 6.15 shows that the mean time taken to find a maximum cardinality stable matching tended to increase as we increased the length of the individual residents' preference lists in the instances. Finding a maximum cardinality stable matching under BIS-stability took longer than finding a maximum cardinality stable matching under MM-stability in all cases, with the time taken increasing more quickly under BIS-stability then under MM-stability.

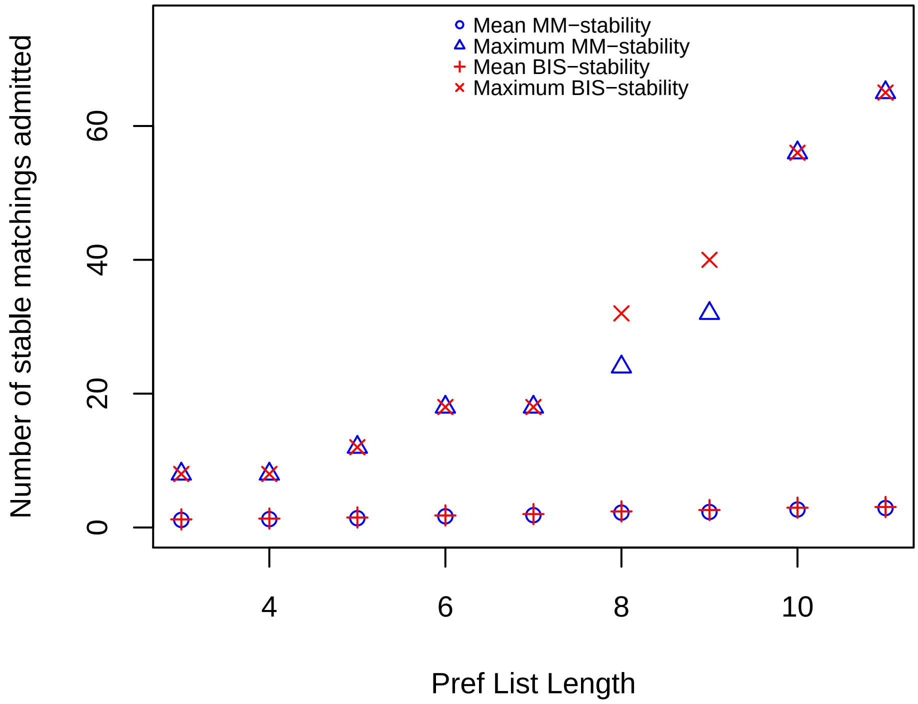

Figure 6.13: HRC Experiment 4 - Number of stable matchings admitted by random instances.

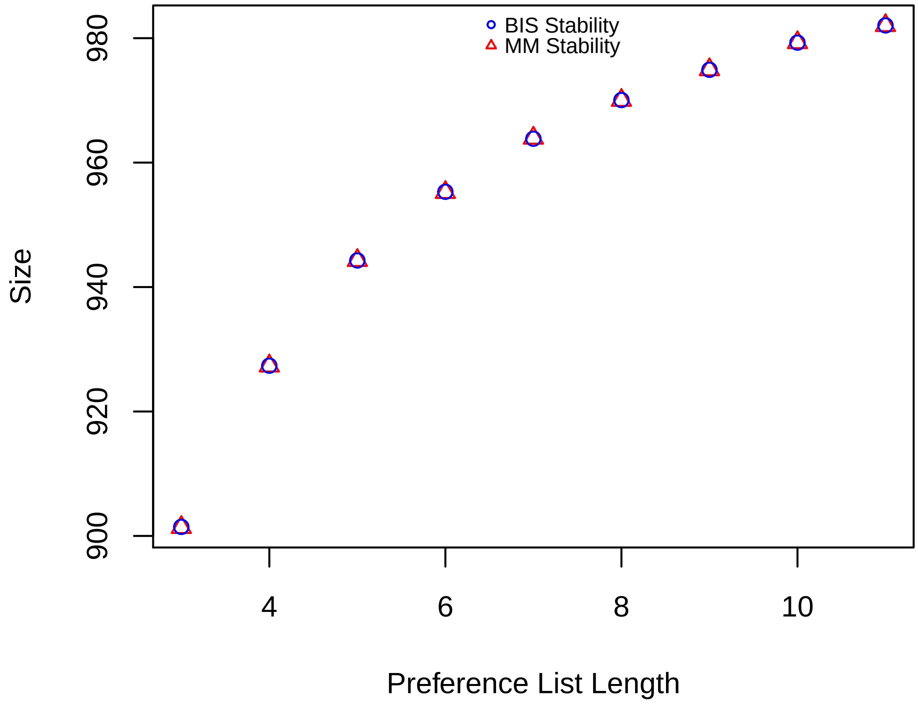

Figure 6.14: HRC Experiment 4 - Mean size of a maximum cardinality stable matching.

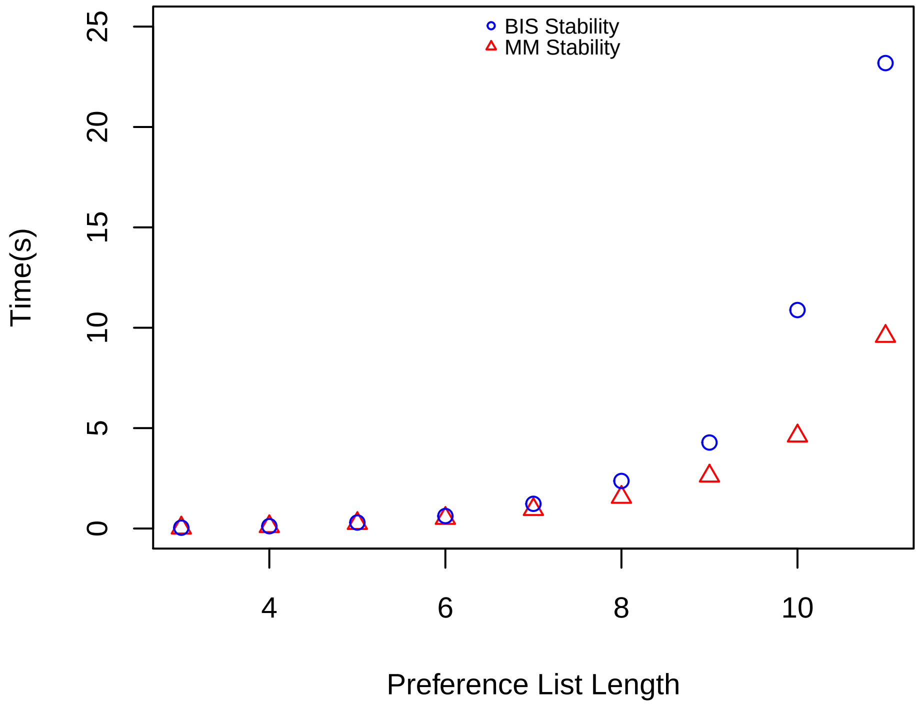

Figure 6.15: HRC Experiment 4 - Mean time to solve to optimality.

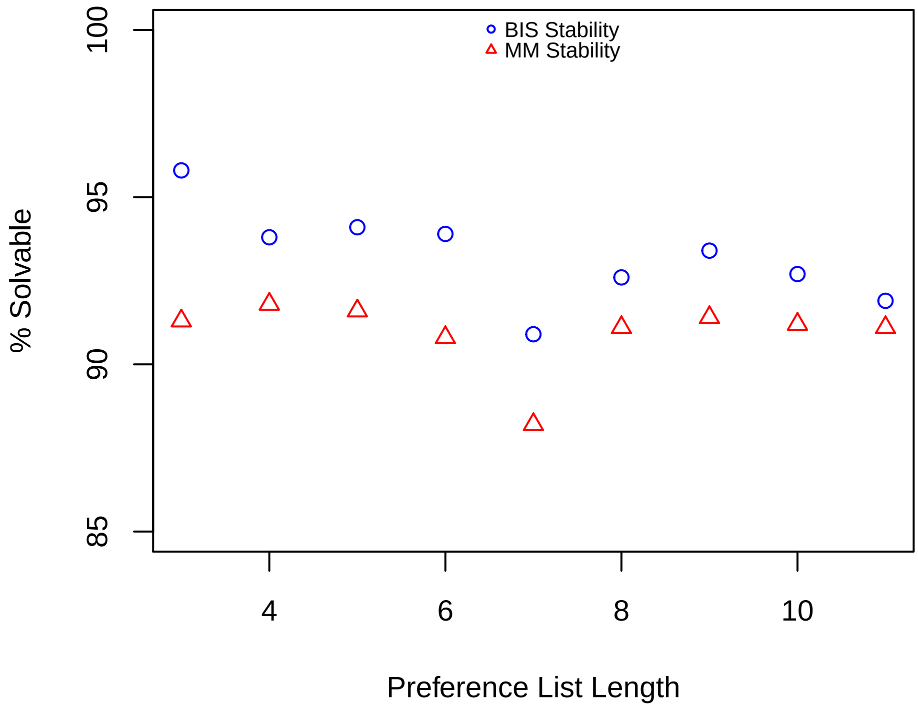

Figure 6.16: HRC Experiment 4 - Percentage of instances admitting a stable matching.

We conjecture that as we increase the length of the residents' preference lists the size of the resulting model increases and hence the time taken to find an optimal solution increases.

The data in Figure 6.16 also shows that the percentage of HRC instances admitting a stable matching under each stability definition does not appear to be correlated with the length of the individual residents' preference lists in the instance. We conjecture that this is because in a maximum cardinality stable matching a great deal of the hospitals will be fully subscribed with residents who rank hospitals early in their preference list. Thus adding additional preferences for hospitals which are likely to already be fully subscribed to the end of the residents' preference lists will not increase the likelihood of a blocking pair. Hence they will not affect the likelihood of the instance admitting a stable matching. It is however noticeable that, as in the previous experiments, the percentage of instances admitting a stable matching was greater under BIS-stability than under MM-stability in all cases.

Pref List Length	Mean Number of Stable Matchings		Max Number of Stable Matchings		Mean Time to Optimal Solution		Number of Solvable Instances		Mean Size of a Max Stable Matching		Mean Number of Variables		Mean Number of Constraints	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
3	1.13	1.22	8	8	0.03	0.04	913	958	901.37	901.46	3.30	3.70	7.12	11.14
4	1.25	1.34	8	8	0.10	0.12	918	938	927.33	927.37	4.28	4.68	10.10	16.12
5	1.39	1.50	12	12	0.25	0.30	916	941	944.29	944.28	5.50	5.90	14.22	23.14
6	1.67	1.81	18	18	0.51	0.62	908	939	955.21	955.32	6.94	7.34	19.68	32.63
7	1.84	2.00	18	18	0.95	1.23	882	909	963.95	963.84	8.62	9.02	26.83	45.25
8	2.22	2.41	24	32	1.56	2.37	911	926	970.02	970.06	10.51	10.91	35.77	61.25
9	2.32	2.61	32	40	2.62	4.28	914	934	974.95	974.93	12.66	13.06	47.06	81.67
10	2.69	2.96	56	56	4.61	10.88	912	927	979.29	979.29	15.02	15.42	60.54	106.29
11	2.91	3.06	65	65	9.57	23.18	911	919	982.04	982.06	17.58	17.98	76.63	135.92

Table 6.4: HRC Experiment 4 - Tabulation of output from HRC Experiment 4.

6.4 HRCT IP models applied to instances arising from the SFAS application

We further extended our implementation in the fashion described in Section 4.5 to find maximum cardinality stable matchings in instances of HRCT and were able to find optimal solutions admitted by the instances arising from the real data obtained from the SFAS application for the years 2010, 2011 and 2012.

In Section 6.4.1 we discuss the difficulties in ensuring that the instances generated from the individual residents' preferences are exactly the same on every occasion. We demonstrate that the non-deterministic nature of the process by which the instances of HRCT are constructed from the individual residents' preference lists makes this very challenging. In Section 6.4.2 we compare the output from the IP models under both MM-stability and BIS-stability with that of the BIS-heuristic as applied in the SFAS application [12]. This heuristic constructed a BIS-stable matching but could not guarantee that the matching was of maximum cardinality.

6.4.1 Comparing solved instances

We cannot guarantee that the instances solved by the BIS-heuristic as part of the SFAS application and the instances solved by the IP models in these experiments correspond to one another directly for any of the three years shown. We demonstrate this by considering the process by which an instance of HRC is constructed from the expressed preferences of the individual residents.

The instance solved by the IP models projects the preference lists of the couples as described in Section 6.2.3 in a process that is intended to mirror that applied in the SFAS application. As described in Section 6.2.3 the couples did not express a joint preference list in the SFAS process, rather we generate a joint preference list for the couples from the individual preference lists that are expressed by the two residents involved in the couples. However, since we generate a strict ordering of hospital pairs in the couples' preference lists by breaking ties in an arbitrary fashion within a lexicographically increasing order on the reverse of the L-tuples as described in Section 6.2.3, the process by which the instance is created is non-deterministic. Thus, any two instances constructed from the same sets of initial individual residents' preferences need not be exactly the same. Hence, any confirmation that the instance solved by the BIS-heuristic at the time in each of the years concerned is one and the same instance as that solved by the IP solver in these experiments seems unlikely.

Further, an additional process was applied in the construction of the SFAS instances at the time that was not applied to the instances solved by the IP models in these experiments. This additional process sought to remove so called *incompatible hospital pairs* from the couples' joint preference lists. *Incompatible hospital pairs* are pairs that the administrator of the scheme had deemed unacceptable for reasons specific to the scheme. These incompatible pairs were removed from the joint preference lists of the couple in the instance solved by the BIS-heuristic at the time. The compatibility matrices for the three years required were not available in these experiments and thus in the instances solved by the IP models these incompatible hospital pairs were not removed from the joint preference lists of the couples. Thus the instances solved by the BIS-heuristic and instances solved by the IP models are not precisely the same.

6.4.2 Results from 2010, 2011 and 2012 SFAS instances

Table 6.5 shows: (i) the size of each maximum cardinality stable matching obtained from the IP models for MM-stability and BIS-stability, (ii) the size of the each BIS-stable matching found by the BIS heuristic, and (iii) the time taken to solve the IP models for each SFAS application dataset in the years 2010, 2011 and 2012.

In 2012 the BIS heuristic found a BIS-stable matching of size 683. The IP model under BIS-stability found that a maximum cardinality BIS-stable matching was of size 682 and the IP model under MM-stability found that a maximum cardinality MM-stable matching was of size 681. Clearly the data for 2012 seems inconsistent – the BIS-stable matching found by the BIS-heuristic is larger in size than the maximum cardinality BIS-stable matching found by the IP model under BIS-stability. However as discussed in Section 6.4.1 this may be due to small differences in the instances solved. Indeed the matching output in 2012 by the BIS-heuristic was found to be BIS-stable in the instance solved by the BIS-heuristic; however, this matching was found to be BIS-unstable in the instance solved by the IP model according to our implemented BIS-stability checker.

Year	No. of Residents	No. of Couples	No. of Hospitals	No. of Posts	Maximum Cardinality MM-stable Matching	Time to Optimal MM-stable Solution	Maximum Cardinality BIS-stable Matching	Time to Optimal BIS-stable Solution	Solution from BIS heuristic
2010	734	20	52	735	681	95.61s	681	171.76s	681
2011	736	12	52	736	688	20.46s	688	11.11s	684
2012	710	17	52	720	681	11.78s	682	17.31s	683

Table 6.5: HRCT IP models applied to instances arising from the SFAS application.

Chapter 7

Empirical results from the IP models for MIN BP HRC

7.1 Introduction

In this chapter we present data from an empirical evaluation of an implementation of the IP model for finding a maximum cardinality 'most stable' matching in an instance of MIN BP HRC under MM-stability described in Chapter 4, and an implementation of the corresponding MIN BP HRC model under BIS-stability derived from the HRC model presented in Chapter 5. The models in these experiments each find a maximum cardinality 'most stable' matching in an arbitrary instance of MIN BP HRC under the corresponding stability definition. We consider the following properties: the time taken to find a maximum cardinality 'most stable' matching; the size of a maximum cardinality 'most stable' matching admitted by an instance; and the number of blocking pairs admitted by a 'most stable' matching. We show how these properties vary as we modify a range of parameters in the constructed instances, including the number of residents in the instance; the percentage of residents involved in couples; the number of hospitals in the instance; and the lengths of the residents' preference lists.

In Section 7.2 we present an overview of the experiments performed in this chapter. In Section 7.3 we give details of the computational environment in which the experiments were performed. Further, in Section 7.2.2 we describe details of the testing applied to the implementation to attempt to increase confidence in the correctness of the implementation.

In the experiments in this chapter, as in the experiments in Chapter 6 we sought to reflect the properties of instances arising in the SFAS application. The randomly generated instances used in these experiments differ little from those used in the experiments in Chapter 6 – these minor differences are detailed in Section 7.2.3. In Section 7.3 we present data from an empirical evaluation of the MIN BP HRC IP models described here to randomly generated instances of HRC reflecting the properties of the SFAS application. As in Chapter 6 we applied the model to 1000 randomly generated instances following the experimental structure used by Biro et al./ in [12].

7.2 Overview of the MIN BP HRC experiments

To find a 'most stable' matching in an instance I of HRC we apply the following procedure. We first use the corresponding HRC IP model to find a maximum cardinality stable matching M in I if such a matching exists. Clearly, if M exists, then M is a maximum cardinality 'most stable' matching. However, if I does not admit a stable matching, then we apply the MIN BP HRC model to I. In this case we apply a lower bound of one on the number of blocking pairs in a 'most stable' matching in I since we know no stable matching exists.

In the experiments that follow we examine the output of the model as we vary the parameters of the instance under both MM-stability and BIS-stability. We applied the model to randomly generated instances reflecting the properties of the instances arising in the SFAS context and we present data on the following outputs from the model as we vary the size of the instance, the percentage of the residents involved in couples, the number of hospitals in the instance and the length of the residents' preference lists:

1. the time taken to find maximum cardinality 'most stable' matchings;
2. the size of a maximum cardinality 'most stable' matching admitted by an instance;
3. the number of blocking pairs admitted by a 'most stable' matching;

7.2.1 Computational environments for MIN BP HRC experiments

We ran experiments on a Java implementation of the IP models as described in Section 4 applied to both randomly-generated and real data. All experiments were carried out on a desktop PC with an Intel i5-2400 3.1Ghz processor, with 8Gb of memory running Windows 7. The IP solver used in all cases was CPLEX 12.4 and the model was implemented in Java using CPLEX Concert.

7.2.2 Correctness testing of the implemented model

We implemented *stability counters* for HRC under both MM-stability and BIS-stabilility in Java. Unlike a stability checker that returns a simple 'yes' or 'no' as to whether a matching is stable, a stability counter returns an integer indicating how many blocking pairs are admitted by the matching. The number of blocking pairs admitted by every solution output by any of the IP models was tested using these stability counters. In all cases the number of blocking pairs returned by a solution to the IP models was found to correspond exactly to the number of blocking pairs reported by the corresponding stability counter.

To further test that our implementations correctly output a maximum cardinality 'most stable' matching according to the implemented stability counter we used a brute force algorithm. The algorithm recursively generated all possible matchings admitted by an instance of MIN BP HRC and selected a matching of maximum cardinality taken over all of the matchings found by our stability counter to admit the minimum number of blocking pairs. Due to the inefficiency of this algorithm it may only be realistically applied to relatively small instances. When solving hundreds of thousands of randomly generated MIN BP HRC instances involving up to fifteen residents, our implementation agreed with the brute force algorithm and always returned a 'most stable' matching of the same size and admitting the same number of blocking pairs as a maximum cardinality 'most stable' matching output by the brute force algorithm.

7.2.3 Construction of random instances MIN BP HRC

The random instances used in the experiments presented in the following sections were generated as described in Sections 6.2.3 and 6.2.4. However, unlike in the HRC experiments described in Section 6 we do not remove any fixed assignments before presenting the instance to the solver leaving open the possibility that a 'most stable' matching may be a matching in which a fixed assignment is not satisfied.

7.3 Experiments with randomly generated instances

7.3.1 MIN BP HRC Experiment 1

In this first experiment, we report on data obtained as we increased the number of residents while maintaining a constant ratio of couples, hospitals and posts to residents. For various values of x $(50 \leq x \leq 150)$ in increments of 20, 1000 randomly generated instances were created containing x residents, $0.1x$ couples (and hence $0.8x$ single residents) and $0.1x$ hospitals with x available posts that were randomly distributed amongst the hospitals. Each resident's preference list contained a minimum of three and a maximum of five hospitals. The mean time taken to find a maximum cardinality 'most stable' matching is displayed in Figure 7.1 for all values of x. Further, the mean time taken to find a maximum cardinality 'most stable' matching for only those instances that did not admit a stable matching is also displayed in Figure 7.1 for all values of x. Figure 7.2 displays the mean size of the maximum cardinality 'most stable' solution for all values of x and also displays the mean size of a maximum cardinality 'most stable' matching taken over only those instances admitting no stable matching for all values of x. Figure 7.3 displays the mean and maximum number of blocking pairs admitted by the instances for all values of x.

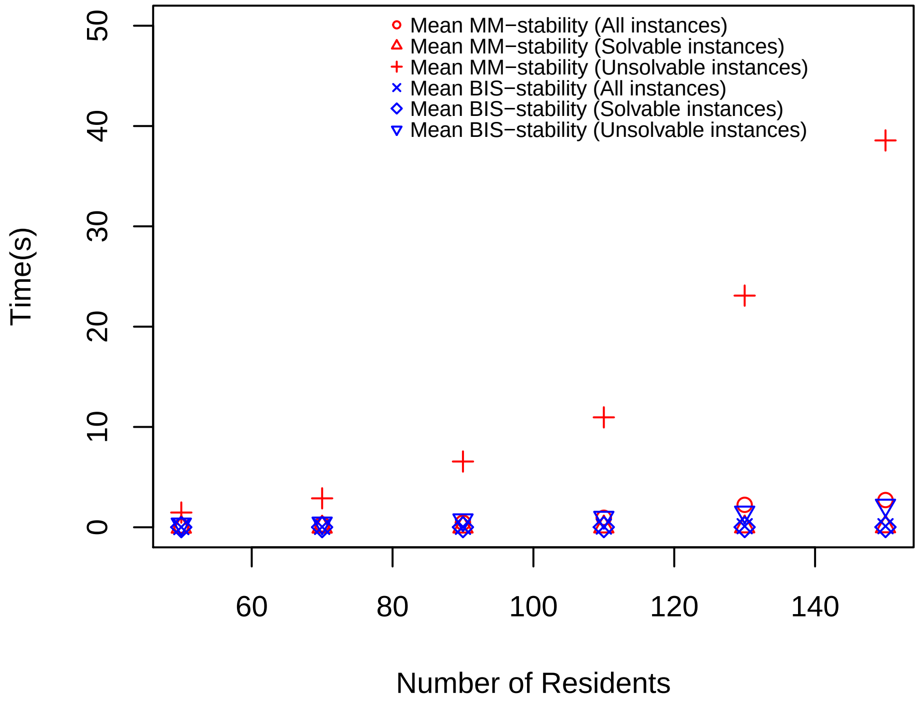

Figure 7.1: MIN BP HRC Experiment 1 - Time taken to find a maximum cardinality 'most stable' matching.

The data in Figure 7.1 show that the mean time taken to find a maximum cardinality 'most stable' matching tends to increase as we increase the number of residents in the instances. The increase is more pronounced for those instances not admitting a stable matching. Instances admitting a stable matching are solved by the HRC IP model and are not considered by the MIN BP HRC model. The instances not admitting a stable matching must be solved by the MIN BP HRC IP model and this model takes much longer to find an optimal solution than the corresponding HRC IP model. The time taken to find a maximum cardinality 'most stable' matching for those instances that do not admit a stable matching increases much more quickly under MM-stability than under BIS-stability.

The data in Figure 7.2 show that as the size of the instance increased the mean size of a maximum cardinality 'most stable' matching admitted by the instances tended to increase under both MM-stability and BIS-stability. However, the mean size of a maximum cardinality 'most stable' matching for those instances not admitting a stable matching is greater

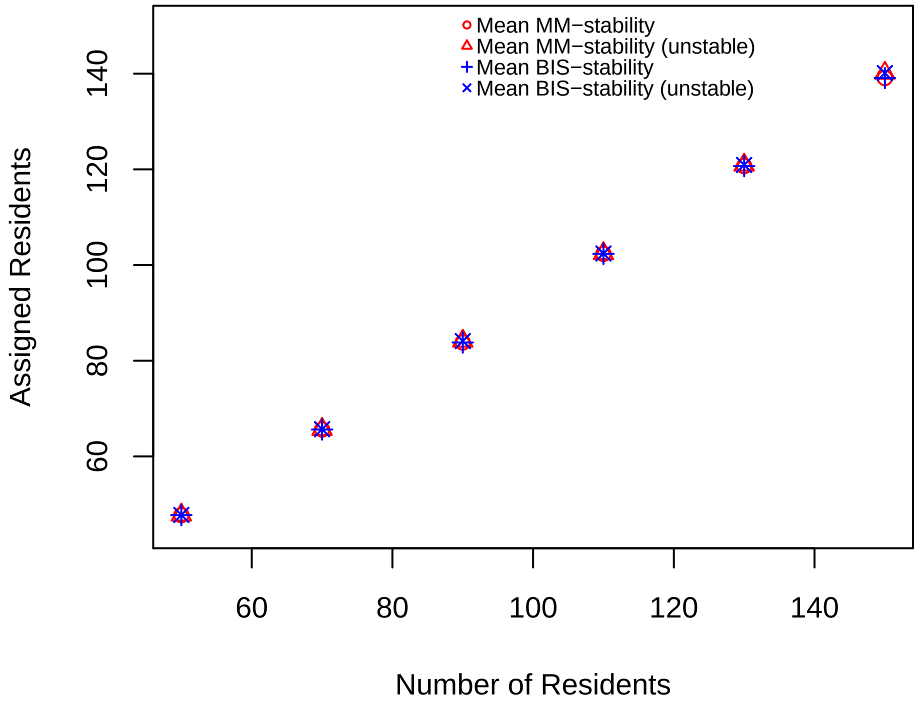

Figure 7.2: MIN BP HRC Experiment 1 - Mean size of a maximum cardinality 'most stable' matching.

in all cases than the mean size of a maximum cardinality 'most stable' matching for those instances that admitting a stable matching. We conjecture that the set of matchings admitting exactly zero blocking pairs is likely to be smaller than the set of matchings admitting more than zero blocking pairs. Hence the latter set is more likely to admit a matching with a larger maximum cardinality 'most stable' matching.

The data in Figure 7.3 show that the maximum number of blocking pairs in a maximum cardinality 'most stable' matching is exactly one for all of the instance sizes considered. The mean number of blocking pairs in a maximum cardinality 'most stable' matching does not appear to alter significantly with the size of the instance. However, the mean number of blocking pairs in a maximum cardinality 'most stable' matching is greater under MM-stability than under BIS-stability. This can be explained by observing that we found in Chapter 6 that the IP model for HRC is slightly more likely to find a stable matching under BIS-stability than under MM-stability.

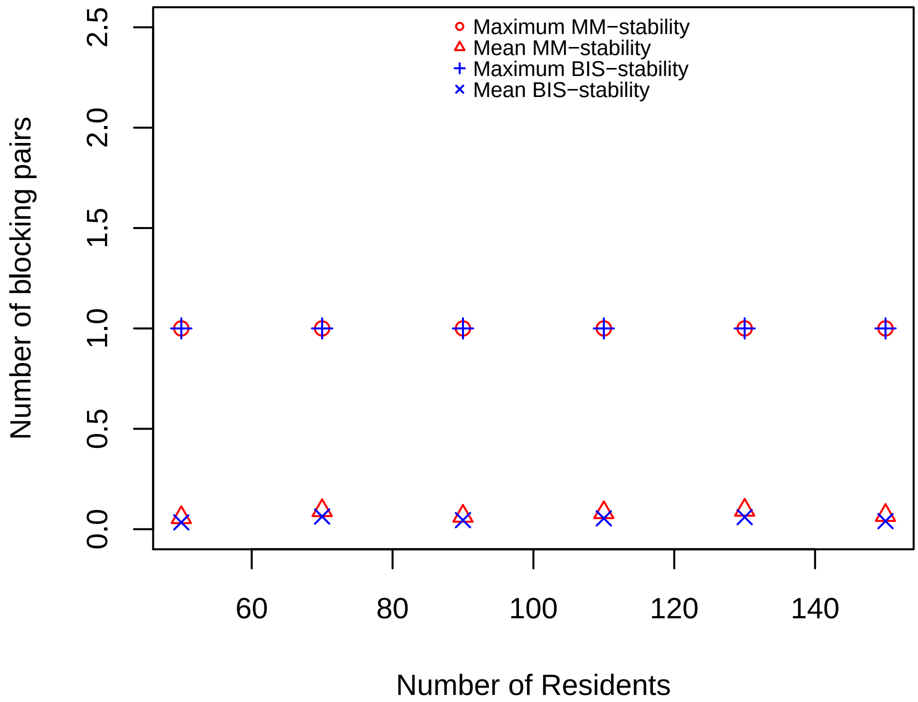

Figure 7.3: MIN BP HRC Experiment 1 - Mean and maximum number of blocking pairs in a 'most stable' matching.

Residents	Mean Time to Optimality (All Instances)		Mean Time to Optimality (Solvable)		Mean Time to Optimality (Unsolvable)		Mean Optimal Matching Size (Solvable)		Mean Optimal Matching Size (Unsolvable)		Mean Number of Blocking Pairs)		Max Number of Blocking Pairs	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
50	0.10	0.02	0.02	0.01	1.47	0.31	47.72	47.72	47.76	47.76	0.06	0.03	1	1
70	0.29	0.04	0.02	0.01	2.88	0.41	65.63	65.63	65.68	65.68	0.09	0.06	1	1
90	0.46	0.04	0.03	0.01	6.56	0.73	83.81	83.81	84.11	84.11	0.06	0.04	1	1
110	0.95	0.07	0.04	0.02	10.96	0.99	102.34	102.34	102.41	102.41	0.08	0.05	1	1
130	2.24	0.11	0.05	0.02	23.09	1.50	120.68	120.68	120.92	120.92	0.10	0.06	1	1
150	2.71	0.11	0.06	0.02	38.56	2.19	139.11	139.11	140.12	140.12	0.07	0.04	1	1

Table 7.1: MIN BP HRC Experiment 1 - Tabulation of output from MIN BP HRC Experiment 1.

7.3.2 MIN BP HRC Experiment 2

In our second experiment, we report on results obtained as we increased the percentage of residents involved in couples while maintaining the same total number of residents, hospitals and posts. For various values of x ($0 \leq x \leq 30$) in increments of 5, 1000 randomly generated instances were created containing 100 residents, x couples (and hence $100 - 2x$ single residents) and 10 hospitals with 1000 available posts that were randomly distributed amongst the hospitals. Each resident's preference list contained a minimum of three and a maximum of five hospitals. The mean time taken to find a maximum cardinality 'most stable' matching is displayed in Figure 7.4 for all values of x alongside the mean time taken to find a maximum cardinality 'most stable' matching for only those instances admitting a stable matching for all values of x. Figure 7.5 displays the mean size of the maximum cardinality 'most stable' matching for all values of x and also displays the mean size of a maximum cardinality 'most stable' matching taken over only those instances that did not admit a stable matching for all values of x. Figure 7.6 displays the mean and maximum number of blocking pairs admitted by the instances for all values of x.

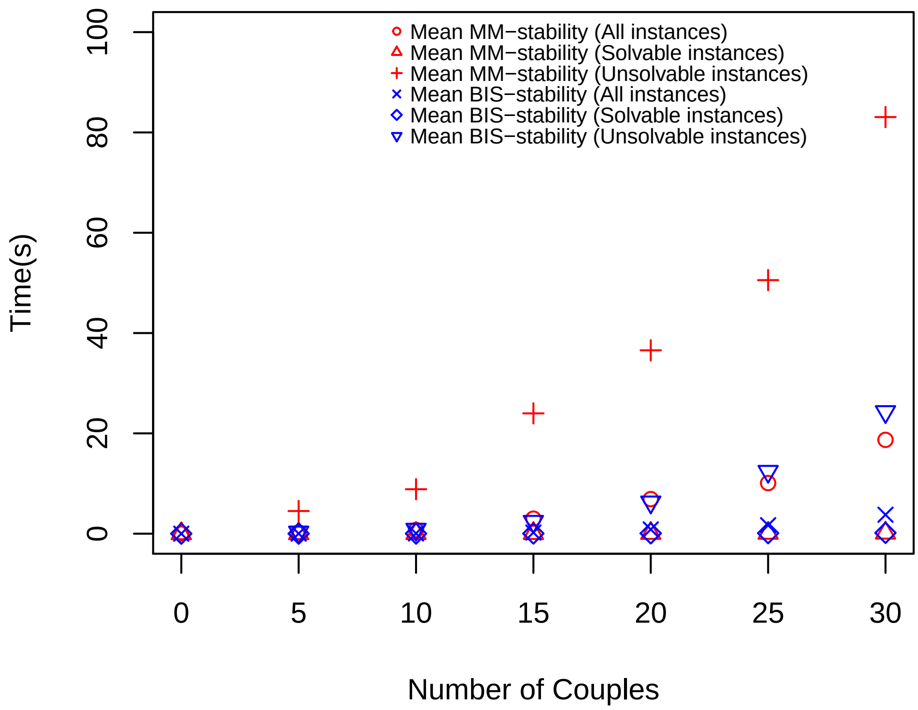

Figure 7.4: MIN BP HRC Experiment 2 - Time taken to find a maximum cardinality 'most stable' matching.

The data in Figure 7.4 shows that the mean time taken to find a maximum cardinality 'most stable' matching tended to increase as we increased the number of residents involved in couples in the instances. Again, the increase is more pronounced for those instances that not

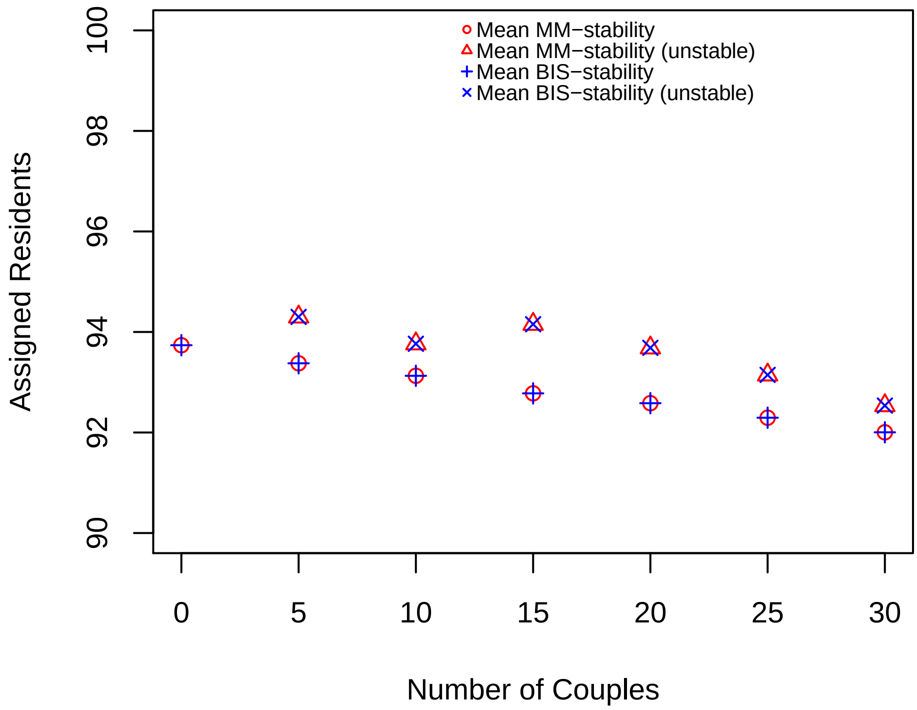

Figure 7.5: MIN BP HRC Experiment 2 - Mean size of a maximum cardinality 'most stable' matching.

admitted a stable matching. This can be understood by observing that instances admitting a stable matching are solved by the HRC IP model and are not considered the MIN BP HRC model and the MIN BP HRC IP model takes much longer to find an optimal solution than the HRC IP model in an arbitrary instance. The time taken to find a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching increased much more quickly under MM-stability than under BIS-stability.

The data in Figure 7.5 shows that as the number of residents involved in couples increased the mean size of a maximum cardinality 'most stable' matching in the instances under both MM-stability and BIS-stability tended to decrease. When the number of couples is non-zero, the mean size of a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching was greater than the mean size of a maximum cardinality 'most stable' matching for those instances that admitted a stable matching. We explain this variance by our conjecture that the set of matchings admitting exactly zero blocking pairs is likely to be smaller than the set of matchings admitting greater than zero blocking pairs. Hence the latter set is more likely to admit a matching with a larger maximum cardinality 'most stable' matching.

The data in Figure 7.6 shows that the maximum number of blocking pairs admitted by a maximum cardinality 'most stable' matching is exactly one for all cases considered other than

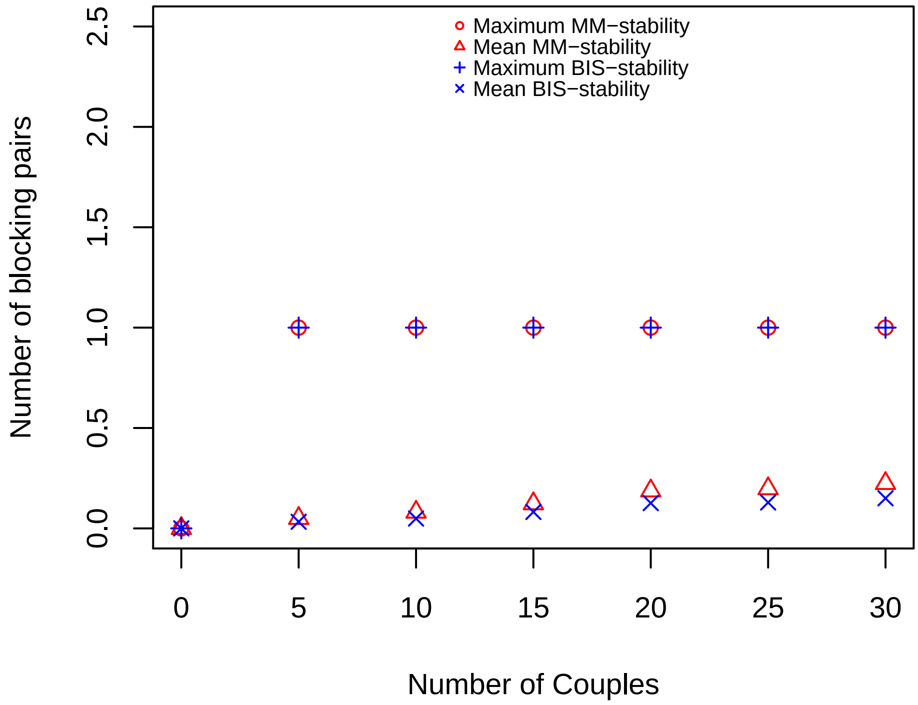

Figure 7.6: MIN BP HRC Experiment 2 - Mean and maximum number of blocking pairs in a 'most stable' matching.

the case in which there are no couples in the instance. Clearly, an instance of HRC with no couples is simply an instance of HR and thus will always admit a stable matching. The mean number of blocking pairs in a maximum cardinality 'most stable' matching increased as the number of residents involved in couples increased. This can be understood by observing that as we increase the number of residents involved in couples in an instance of HRC the instance is less likely to admit a stable matching and thus a maximum cardinality 'most stable' matching is more likely to admit one or more blocking pairs. Again, the mean number of blocking pairs in a maximum cardinality 'most stable' matching is greater under MM-stability than under BIS-stability. This can be explained by observing that an instance of HRC is slightly more likely to admit a stable matching under BIS-stability than under MM-stability.

Couples	Mean Time to Optimality (All Instances)		Mean Time to Optimality (Solvable)		Mean Time to Optimality (Unsolvable)		Mean Optimal Matching Size (Solvable)		Mean Optimal Matching Size (Unsolvable)		Mean Number of Blocking Pairs)		Max Number of Blocking Pairs	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
0	0.01	0.00	0.01	0.00	-	-	93.74	93.74	-	-	0.00	0.00	0	0
5	0.25	0.01	0.02	0.00	4.52	0.30	93.38	93.38	94.30	94.30	0.05	0.03	1	1
10	0.75	0.05	0.04	0.01	8.86	0.86	93.13	93.13	93.77	93.77	0.08	0.05	1	1
15	3.00	0.23	0.05	0.03	23.98	2.43	92.78	92.78	94.15	94.15	0.12	0.08	1	1
20	6.90	0.85	0.07	0.06	36.55	6.29	92.58	92.58	93.68	93.68	0.19	0.13	1	1
25	10.09	1.69	0.10	0.10	50.55	12.40	92.29	92.29	93.15	93.15	0.20	0.13	1	1
30	18.71	3.78	0.13	0.16	83.06	24.31	92.00	92.00	92.54	92.54	0.22	0.15	1	1

Table 7.2: MIN BP HRC Experiment 2 - Tabulation of output from MIN BP HRC Experiment 2.

7.3.3 MIN BP HRC Experiment 3

In our third experiment, we report on data obtained as we increased the number of hospitals in the instance while maintaining the same total number of residents, couples and posts. For various values of x $(10 \leq x \leq 100)$ in increments of 10, 1000 randomly generated instances were created containing 100 residents, 10 couples (and hence 80 single residents) and x hospitals with 1000 available posts that were randomly distributed amongst the hospitals. Each resident's preference list contained a minimum of three and a maximum of five hospitals. The mean time taken to find a maximum cardinality 'most stable' matching is displayed in Figure 7.7 for all values of x alongside the mean time taken to find a maximum cardinality 'most stable' matching for only those instances that did not admit a stable matching for all values of x. Figure 7.8 displays the mean size of the maximum cardinality 'most stable' solution for all values of x and also displays the mean size of a maximum cardinality 'most stable' matching taken over only those instances that did not admit a stable matching for all values of x. Figure 7.9 displays the mean and maximum number of blocking pairs admitted by the instances for all values of x.

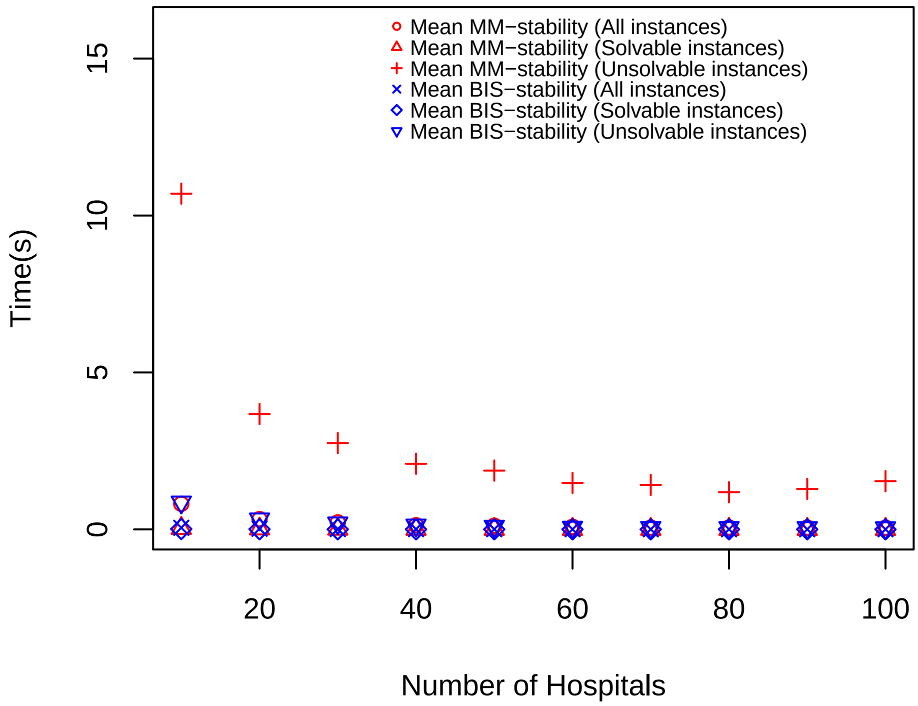

Figure 7.7: MIN BP HRC Experiment 3 - Time taken to find a maximum cardinality 'most stable' matching.

The data in Figure 7.7 shows that the mean time taken to find a maximum cardinality 'most stable' matching tended to decrease as we increased the number of hospitals in the instances. As in the previous MIN BP HRC experiments, the mean time taken to find a 'most stable'

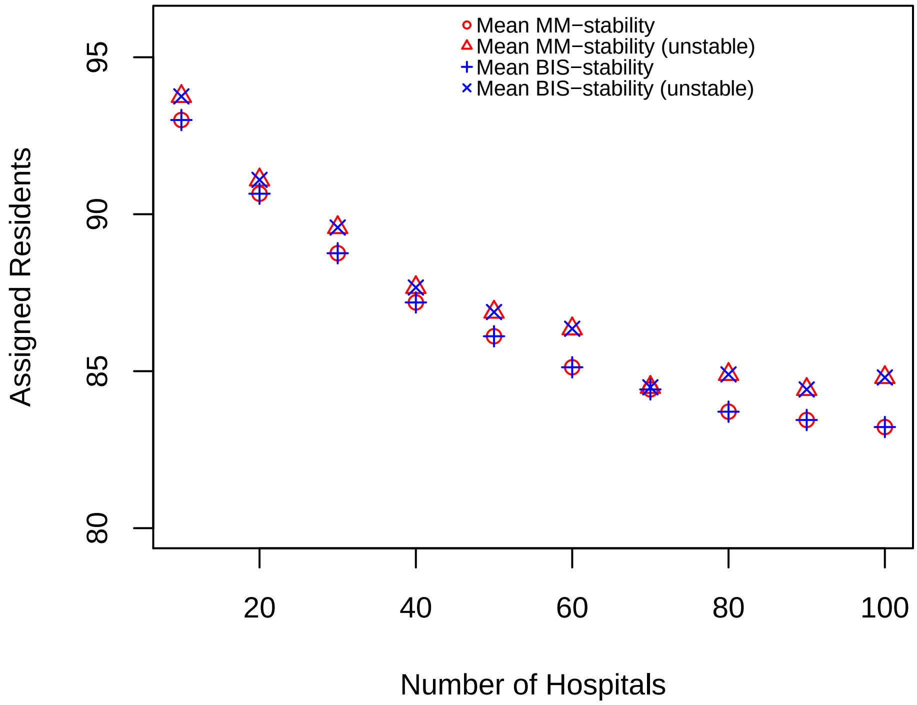

Figure 7.8: MIN BP HRC Experiment 3 - Mean size of a maximum cardinality 'most stable' matching.

matching is greater for instances that did not admit a stable matching as instances that admit a stable matching are solved by the HRC model only and are not considered by the MIN BP HRC model. The mean time taken to find a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching is consistently greater under MM-stability than under BIS-stability.

The data in Figure 7.8 shows that as the number of hospitals in the instance increased the mean size of a maximum cardinality 'most stable' matching in the instances under both MM-stability and BIS-stability tended to decrease. However, the mean size of a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching is greater in all cases than the mean size of a maximum cardinality 'most stable' matching for those instances that did admit a stable matching. As in the previous experiments we explain this variance by our conjecture that the set of matchings admitting exactly zero blocking pairs is likely to be smaller than the set of matchings admitting greater than zero blocking pairs and hence the latter set is more likely to admit a matching with a larger maximum cardinality 'most stable' matching.

The data in Figure 7.9 shows that the maximum number of blocking pairs in a maximum cardinality 'most stable' matching is either one or two for all of the instances considered. This finding differs from the other experiments in this chapter – the maximum number of

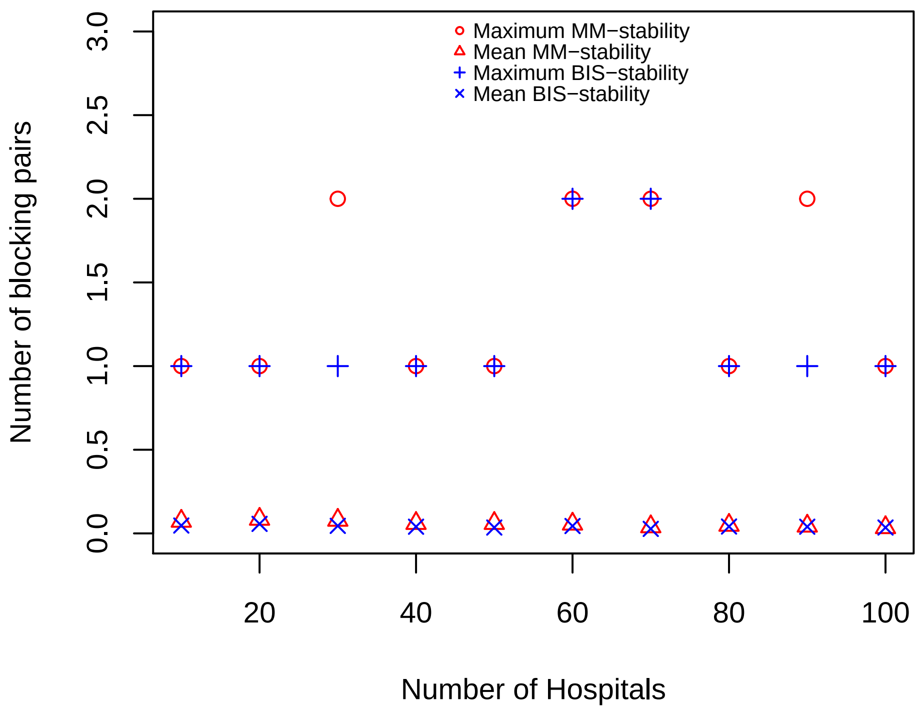

Figure 7.9: MIN BP HRC Experiment 3 - Mean and maximum number of blocking pairs in a 'most stable' matching.

blocking pairs in a maximum cardinality 'most stable' matching was exactly one in every other experiment presented in this chapter. The mean number of blocking pairs in a maximum cardinality 'most stable' matching does not appear to alter as the number of hospitals in the instance increased. Again, the mean number of blocking pairs in a maximum cardinality 'most stable' matching is greater under MM-stability than under BIS-stability. However, this difference is less pronounced where the number of hospitals in the instance is larger.

Hospitals	Mean Time to Optimality (All Instances)		Mean Time to Optimality (Solvable)		Mean Time to Optimality (Unsolvable)		Mean Optimal Matching Size (Solvable)		Mean Optimal Matching Size (Unsolvable)		Mean Number of Blocking Pairs)		Max Number of Blocking Pairs	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
10	0.81	0.05	0.04	0.01	10.70	0.87	93.00	93.00	93.75	93.75	0.07	0.05	1	1
20	0.33	0.02	0.02	0.00	3.68	0.33	90.65	90.65	91.09	91.09	0.08	0.06	1	1
30	0.22	0.01	0.01	0.00	2.75	0.20	88.76	88.76	89.58	89.58	0.08	0.04	2	1
40	0.13	0.01	0.01	0.00	2.09	0.13	87.19	87.19	87.67	87.67	0.06	0.04	1	1
50	0.12	0.01	0.01	0.00	1.87	0.10	86.11	86.11	86.88	86.88	0.06	0.04	1	1
60	0.08	0.00	0.00	0.00	1.48	0.07	85.12	85.12	86.35	86.35	0.06	0.04	2	2
70	0.06	0.00	0.00	0.00	1.42	0.06	84.42	84.42	84.49	84.49	0.04	0.03	2	2
80	0.06	0.00	0.00	0.00	1.18	0.06	83.71	83.71	84.90	84.90	0.05	0.04	1	1
90	0.06	0.00	0.00	0.00	1.29	0.05	83.45	83.45	84.42	84.42	0.04	0.04	2	1
100	0.06	0.00	0.00	0.00	1.53	0.05	83.22	83.22	84.80	84.80	0.04	0.04	1	1

Table 7.3: MIN BP HRC Experiment 3 - Tabulation of output from MIN BP HRC Experiment 3.

7.3.4 MIN BP HRC Experiment 4

In our last experiment, we report on data obtained as we increased the length of the individual preference lists for the residents in the instance while maintaining the same total number of residents, couples, hospitals and posts. For various values of x $(2 \leq x \leq 6)$, 1000 randomly generated instances were created containing 100 residents, 10 couples (and hence 80 single residents) and 10 hospitals with 1000 available posts that were randomly distributed amongst the hospitals. Each resident's preference list contained exactly x hospitals. The mean time taken to find a maximum cardinality 'most stable' matching is displayed in Figure 7.10 for all values of x alongside the mean time taken to find a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching for all values of x. Figure 7.11 displays the mean size of the maximum cardinality 'most stable' matching for all values of x and also displays the mean size of a maximum cardinality 'most stable' matching taken over only those instances that did not admit a stable matching for all values of x. Figure 7.12 displays then mean and maximum number of blocking pairs admitted by the instances for all values of x.

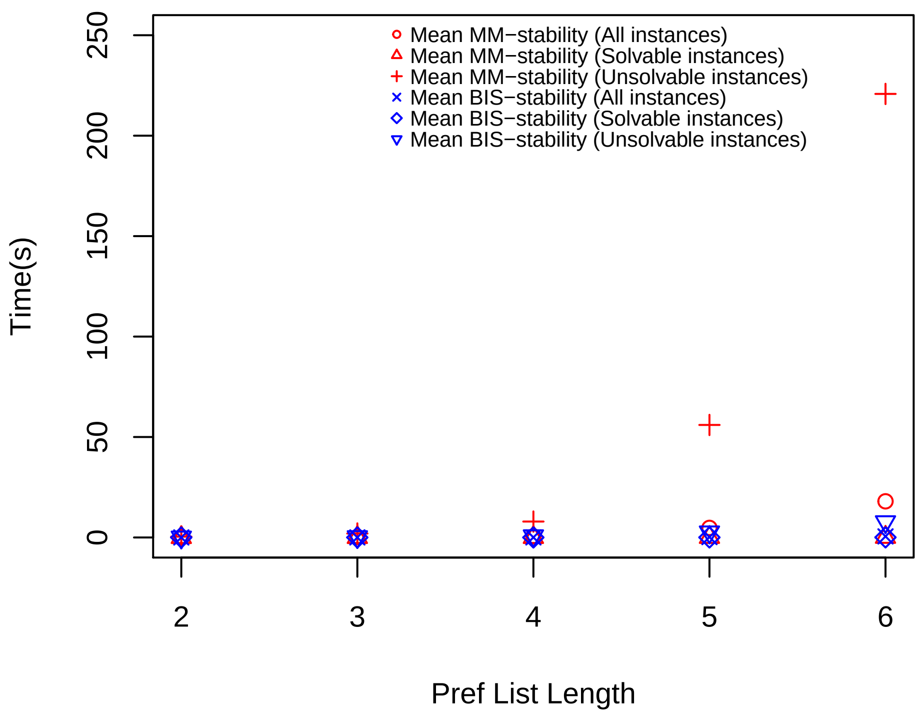

Figure 7.10: MIN BP HRC Experiment 4 - Time taken to find a maximum cardinality 'most stable' matching.

The data in Figure 7.10 shows that the mean time taken to find a maximum cardinality 'most stable' matching tended to increase as we increased the length of the residents' preference lists in the instances. As in the previous MIN BP HRC experiments, the mean time taken to

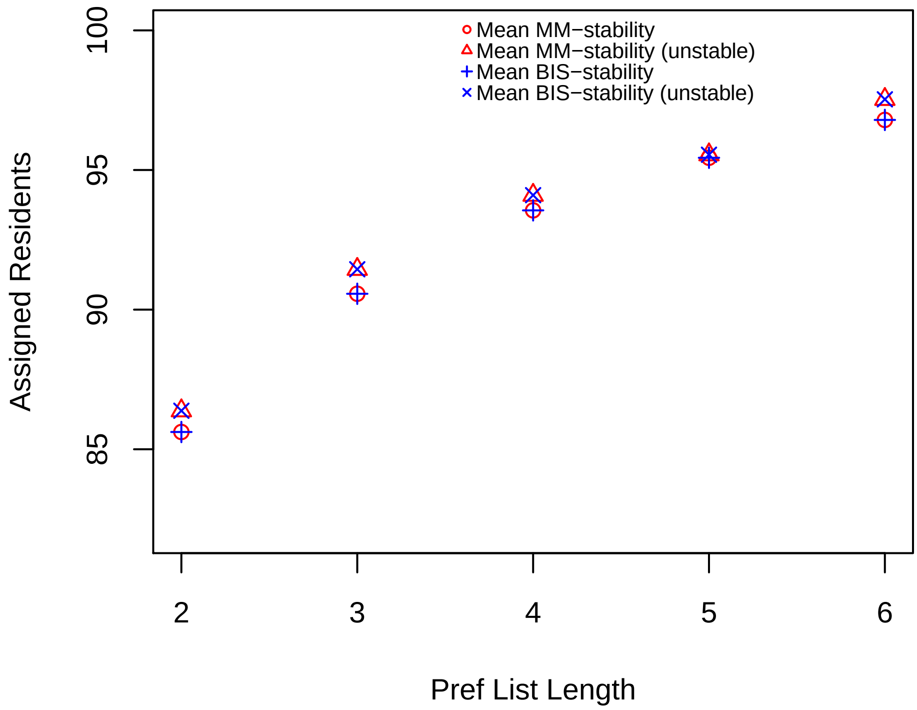

Figure 7.11: MIN BP HRC Experiment 4 - Mean size of a maximum cardinality 'most stable' matching.

find a 'most stable' matching is greater for instances that did not admit a stable matching as instances admitting a stable matching are solved by the HRC IP model only and are not considered by the MIN BP HRC model. The mean time taken to find a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching is consistently greater under MM-stability than under BIS-stability and this difference is more pronounced for longer resident preference lists.

The data in Figure 7.11 shows that as the length of the residents' preference lists increased the mean size of a maximum cardinality 'most stable' matching admitted by the instances under both MM-stability and BIS-stability tended to increase. However, the mean size of a maximum cardinality 'most stable' matching for those instances that did not admit a stable matching is greater in all cases than the mean size of a maximum cardinality 'most stable' matching for those instances that admitted a stable matching. As in the previous experiments we explain this variance by our conjecture that the set of matchings admitting exactly zero blocking pairs is likely to be smaller than the set of matchings admitting greater than zero blocking pairs and thus the latter set is more likely to admit a matching with a larger maximum cardinality 'most stable' matching.

The data in Figure 7.12 shows that the maximum number of blocking pairs in a maximum cardinality 'most stable' matching is exactly one for all of the instance considered. The mean

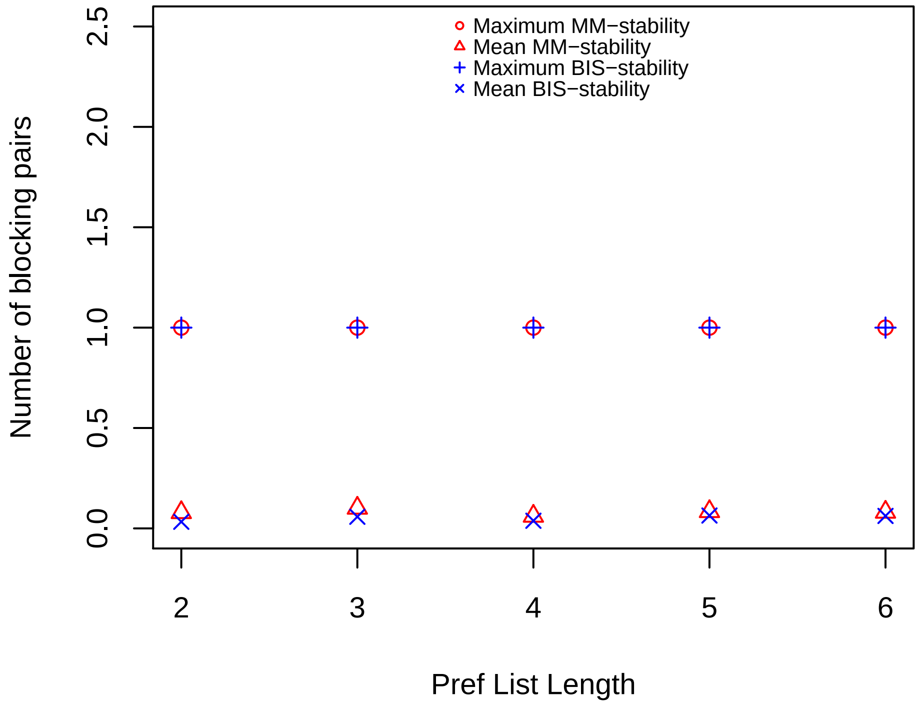

Figure 7.12: MIN BP HRC Experiment 4 - Mean and maximum number of blocking pairs in a ‘most stable’ matching.

number of blocking pairs in a maximum cardinality ‘most stable’ matching does not appear to alter significantly as the length of the residents’ preference lists increased. Again, the mean number of blocking pairs in a maximum cardinality ‘most stable’ matching is consistently greater under MM-stability than under BIS-stability.

Pref List Length	Mean Time to Optimality (All Instances)		Mean Time to Optimality (Solvable)		Mean Time to Optimality (Unsolvable)		Mean Optimal Matching Size (Solvable)		Mean Optimal Matching Size (Unsolvable)		Mean Number of Blocking Pairs)		Max Number of Blocking Pairs	
	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS	MM	BIS
2	0.04	0.01	0.01	0.00	0.42	0.14	85.62	85.62	86.38	86.38	0.08	0.03	1	1
3	0.22	0.03	0.02	0.01	1.98	0.33	90.57	90.57	91.45	91.45	0.10	0.06	1	1
4	0.52	0.04	0.04	0.01	7.92	0.78	93.55	93.55	94.10	94.10	0.06	0.04	1	1
5	4.77	0.20	0.07	0.03	56.01	2.69	95.44	95.44	95.55	95.55	0.08	0.06	1	1
6	17.99	0.53	0.12	0.05	220.78	7.78	96.79	96.79	97.53	97.53	0.08	0.06	1	1

Table 7.4: MIN BP HRC Experiment 4 - Tabulation of output from MIN BP HRC Experiment 4.

Chapter 8

Integer programming models for HR with score limits

8.1 Introduction

In this chapter we describe the first IP models for HR SLT variants with upper and lower quotas and also the HR SLT variant in which residents may express preferences over unordered pairs of hospitals. These matching problems are motivated by applications in the Hungarian Higher Education scheme. In Sections 8.2, 8.3 and 8.4 we present and prove the correctness of three IP models for finding a minimal set of H-stable score limits in instances of HR SLT variants. In Section 8.2 we present and prove the correctness of an IP model for finding a minimal set of H-stable score limits in an instance of HR SLT, in Section 8.3 we present an IP model for finding a minimal set of H-stable score limits in an instance the *Hospitals / Residents problem with Stable Score Limits, Ties and Paired Applications*, and in Section 8.4 we present an IP model for finding a minimal set of H-stable score limits in an instance of the *Hospitals / Residents problem with Stable Score Limits, Ties and Common Upper Quotas*. By construction these models can only output a minimal set of H-stable score limits in an instance of the corresponding problem – a minimal set of H-stable score limits has the property that the sum over the score limits in the set is minimal taken over all of sets of stable score limits admitted by the instance. These models are designated Type A models.

Subsequently, we present and prove the correctness of two models for finding a set of H-stable score limits in an instance of an HR SLT variant that may have desirable properties other than minimality. By amending the structure of the Type A models we can create a new type of model in which we can apply an objective function that allows us to choose some desirable property other than minimality for an optimal set of H-stable score limits output by the model – models of this new type are designated Type B models. We describe each of the models and in each case apply an objective function that seeks to find a set of H-stable score limits $\hat{t}$ in an instance such that the assignment induced from $\hat{t}$ is of maximum cardinality taken over all of the matchings induced from the sets of H-stable score limits admitted by the instance. (By appropriate choice of objective functions these models could be made to output solutions with other desirable properties.) In Section 8.5 we present an IP model for finding a set of H-stable score limits in arbitrary instance of HR SLT such that the assignment induced from the set of H-stable score limits is of maximum cardinality taken over all of the matchings induced from the sets of H-stable score limits in the instance. In Section 8.6 we present an IP model for finding a set of H-stable score limits in an arbitrary instance of the *Hospitals / Residents problem with Ties, Lower Quotas and Stable Score Limits* such that the assignment induced from the set of H-stable score limits is of maximum cardinality taken over all of the matchings induced from the sets of H-stable score limits in the instance.

8.2 A Type A IP model for HR SLT

We describe the variables and constraints in the IP model for finding a minimal set of H-stable score limits in an instance of HR SLT in Sections 8.2.1 and 8.2.2 respectively and in Section 8.2.3 we prove the correctness of the model.

8.2.1 Variables in HR SLT A

Let I be an instance of HR SLT as described in Section 2.7. Let J be the following IP formulation of I. In J, for each i $(1 \le i \le n_1)$ and p $(1 \le p \le l(r_i))$, define a variable $x_{i,p}$ such that:

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \le i \le n_1 \wedge 1 \le p \le l(r_i)\}$. Let $pref(r_i, p)$ denote the hospital at position p of r_i's preference list where $1 \le i \le n_1$ and $1 \le p \le l(r_i)$. Let $Y = \{y_1, y_2 \ldots, y_{n_2}\}$ be a set of variables where intuitively y_j is the score limit of h_j in J for $1 \le j \le n_2$.

8.2.2 Constraints in HR SLT A

The following constraint simply confirms that each variable $x_{i,p}$ must be binary valued for all i $(1 \le i \le n_1)$ and p $(1 \le p \le l(r_i))$:

$$x_{i,p} \in \{0, 1\} \tag{8.1}$$

As each resident $r_i \in R$ is either assigned to a single hospital or is unassigned, we introduce the following constraint for all i $(1 \leq i \leq n_1)$:

$$\sum_{p=1}^{l(r_i)} x_{i,p} \leq 1 \tag{8.2}$$

Since a hospital h_j may be assigned to at most c_j residents, $x_{i,p} = 1$ where $\mathit{pref}(r_i, p) = h_j$ for at most c_j residents. We define the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : \mathit{pref}(r_i, p) = h_j\} \leq c_j \tag{8.3}$$

The following constraint ensures that any r_i assigned to a hospital h_j achieves a score of at least y_j. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we define the following constraint where $\mathit{pref}(r_i, p) = h_j$:

$$y_j \leq (1 - x_{i,p})(\bar{s}_j + 1) + s_{i,j} \tag{8.4}$$

The following constraint ensures that any r_i who is not assigned to a hospital h_j either does not achieve the score limit at h_j (and thus $s_{i,j} < y_j$) or is assigned to a better hospital. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $\mathit{pref}(r_i, p) = h_j$:

$$s_{i,j} + 1 \leq (\bar{s}_j + 1) \sum_{p'=1}^{p} x_{i,p'} + y_j \tag{8.5}$$

Objective Function: A minimal set of H-stable score limits in I is a set of H-stable score limits in I where $\sum_{t \in \hat{t}} t$ is minimal taken over all the sets of H-stable score limits in I. We apply the following objective function to ensure that an optimal solution meets this criterion:

$$\min \sum_{j=1}^{n_2} y_j \tag{8.6}$$

8.2.3 Proof of correctness of the constraints in HR SLT A

We now establish the correctness of the IP model presented in Sections 8.2.1 and 8.2.2.

Theorem 8.2.1. *Given an instance I of HR SLT, let J be the corresponding IP model as defined in Sections 8.2.1 and 8.2.2. A minimal set of H-stable score limits $\hat{t}$ in I and the assignment induced in I from $\hat{t}$ are exactly equivalent to an optimal solution to J.*

Proof. First, let I be an instance of HR SLT. Let $\hat{t}$ be a minimal set of H-stable score limits in I and let M be the assignment in I induced from $\hat{t}$. We form an assignment of values to the variables $\langle$ **x**, **y** $\rangle$ in J as follows. Initially $x_{i,p} = 0$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, then for each $(r_i, h_j) \in M$, $x_{i,p} = 1$ where $h_j = pref(r_i, p)$. For each $h_j \in H$, set $y_j \in Y$ to the score limit at hospital h_j in I.

We now show that $\langle$ **x**, **y** $\rangle$ satisfies all of the constraints in the model. As each resident has a single partner or is unassigned (but not both), for a given i $(1 \leq i \leq n_1)$, it follows that $x_{i,p} = 1$ for at most one value of p in the range $1 \leq p \leq l(r_i)$, and for each other value of p in the same range, $x_{i,p} = 0$. Hence, Constraints 8.1 and 8.2 are satisfied in $\langle$ **x**, **y** $\rangle$. Since each hospital is assigned in M to at most c_j acceptable residents, Constraint 8.3 is also satisfied in $\langle$ **x**, **y** $\rangle$. Since M is the assignment in I induced from $\hat{t}$, each resident is assigned in M to the first hospital on his preference list at which he meets the score limit. Thus Constraints 8.4 and 8.5 must be satisfied in $\langle$ **x**, **y** $\rangle$.

Since $\hat{t}$ is a set of H-stable score limits in I then no hospital in I can reduce its score limit further without exceeding its capacity. Further since $\hat{t}$ is a minimal set of H-stable score limits in I, it follows that $\sum_{t \in \hat{t}} t$ is minimal over the sets of H-stable score limits in I and thus $\sum_{y \in Y} y$ is minimal taken over the feasible solutions in J. Hence the objective function in Constraint 8.6 is satisfied. Thus we have that a minimal set of H-stable score limits $\hat{t}$ in I and the assignment induced from $\hat{t}$ give rise to an optimal solution to J.

Conversely, consider an optimal solution $\langle$ **x**, **y** $\rangle$ to J, it follows that $\sum_{y \in Y} y$ is minimal taken over all feasible solutions to J. From such a solution we form in I a set of pairs, M, and a set of score limits $\hat{t}$ as follows. Initially let $M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, if $x_{i,p} = 1$, then add (r_i, h_j) to M where $h_j = pref(r_i, p)$. Further, for each $y_j \in Y$ set the score limit at hospital h_j in $\hat{t}$ to $y_j \in Y$.

As $\langle$ **x**, **y** $\rangle$ satisfies Constraints 8.1, 8.2 and 8.3, each resident in M must have exactly one partner or be unassigned (but not both) and each hospital h_j in M must have at most c_j assignees. We now show that $\hat{t}$ is a set of H-stable score limits in I. Assume an applicant r_i is assigned to a hospital h_j at which he does not achieve the score limit where $pref(r_i, p) = h_j$. Hence, both $(1 - x_{i,p})\bar{s}_j = 0$ and further $s_{i,j} < y_j$. Thus Constraint 8.4 is not satisfied in $\langle$ **x**, **y** $\rangle$, a contradiction. Now assume that some r_i is not assigned to a hospital h_j at which they achieved the score limit. Further assume that r_i is not assigned to a better hospital than h_j. Hence, $(\bar{s}_j + 1) \sum_{p'=1}^{p} x_{i,p'} = 0$ and $s_{i,j} \geq y_j$ where $pref(r_i, p) = h_j$. Thus Constraint 8.5 is not satisfied in $\langle$ **x**, **y** $\rangle$, a contradiction.

Since the objective function Constraint 8.6 is satisfied in $\langle$ **x**, **y** $\rangle$, no hospital may reduce its score limit without exceeding its capacity and thus $\hat{t}$ is a set of H-stable score limits in I; moreover, M is the assignment in I induced from $\hat{t}$. Further, the objective function ensures that $\hat{t}$ is a minimal set of H-stable score limits in I and the result is proven. □

8.3 A Type A IP model for HR PA SLT

We describe the variables and constraints in the IP model for finding a minimal set of H-stable score limits in an instance of HR PA SLT in Sections 8.3.1 and 8.3.2 respectively and in Section 8.3.3 we prove the correctness of the model.

8.3.1 Variables in HR PA SLT A

Let I be an instance of HR PA SLT as described in Section 2.7.2. Further, let J be the following IP formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that:

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice application} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(r_i)\}$. Further, let $Y = \{y_1, y_2 \ldots, y_{n_2}\}$ be a set of variables where intuitively y_j is the score limit at h_j in J for each $1 \leq j \leq n_2$.

Now, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new variable $\tau_{i,p} \in \{0, 1\}$. The intuitive meaning of $\tau_{i,p}$ is that if r_i's p^{th} application is to a single hospital, then $\tau_{i,p} = 0$. Otherwise $\tau_{i,p}$ may take the value of zero or one. Constraints 8.7 and 8.8 described in Section 8.3.2 are applied to enforce this property.

8.3.2 Constraints in HR PA SLT A

The HR PA SLT A model is constructed by applying Constraints 8.1 and 8.2 from HR SLT A model described in Section 8.2.2 in addition to the constraints described below.

The following constraint simply confirms that each variable $\tau_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$:

$$\tau_{i,p} \in \{0, 1\} \tag{8.7}$$

To ensure that if $|pref(r_i, p)| = 1$, then $\tau_{i,p} = 0$, we apply the following constraint:

$$\tau_{i,p} \leq |pref(r_i, p)| - 1 \tag{8.8}$$

Since a hospital h_j may be assigned to at most c_j residents, $x_{i,p} = 1$ where $h_j \in pref(r_i, p)$ for at most c_j residents. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1}\sum_{p=1}^{l(r_i)}\{x_{i,p} \in X : h_j \in pref(r_i,p)\} \leq c_j \tag{8.9}$$

The following constraint ensures that any r_i assigned to a hospital h_j achieves a score of at least y_j. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define the following constraint where $h_j \in pref(r_i, p)$:

$$y_j \leq (1 - x_{i,p})(\bar{s}_j + 1) + s_{i,j} \tag{8.10}$$

The following constraint ensures that any r_i who is not assigned to the application at position p on his preference list where $pref(r_i, p)$ is a single hospital h_j either does not achieve the score limit at h_j (and thus $s_{i,j} < y_j$) or is assigned to a better application. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$:

$$s_{i,j} + 1 \leq (\bar{s}_j + 1)\sum_{p'=1}^{p} x_{i,p'} + y_j \tag{8.11}$$

The following constraint ensures that any r_i who is not assigned to the application at position p on his preference list where $pref(r_i, p)$ is a pair of hospitals $\{h_j, h_k\}$ either does not achieve the score limit of at least one of h_j or h_k or r_i is assigned to a better application. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = \{h_j, h_k\}$:

$$s_{i,j} + 1 \leq (\bar{s}_j + 1)\left(\sum_{p'=1}^{p} x_{i,p'} + \tau_{i,p}\right) + y_j \tag{8.12}$$

$$s_{i,k} + 1 \leq (\bar{s}_k + 1)\left(\sum_{p'=1}^{p} x_{i,p'} + (1 - \tau_{i,p})\right) + y_k \tag{8.13}$$

Objective Function: A minimal set of H-stable score limits in I is a set of H-stable score limits in I where $\sum_{t\in\hat{t}} t$ is minimal taken over all of the sets of H-stable score limits in I. We apply the following objective function to ensure that an optimal solution meets this criterion:

$$\min \sum_{j=1}^{n_2} y_j \tag{8.14}$$

8.3.3 Proof of correctness of the constraints in HR PA SLT A

We now establish the correctness of the IP model presented in Sections 8.3.1 and 8.3.2.

Theorem 8.3.1. *Given an instance I of* HR PA SLT, *let J be the corresponding IP model as defined in Sections 8.3.1 and 8.3.2. A minimal set of H-stable score limits $\hat{t}$ in I and the assignment in I induced from $\hat{t}$ are exactly equivalent to an optimal solution to J.*

Proof. Let I be an instance of HR PA SLT. Let $\hat{t}$ be a minimal set of H-stable score limits in I and let M be the assignment induced in I from $\hat{t}$. We form an assignment of values to the variables $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$ in J as follows. Initially, $x_{i,p} = 0$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$. For each $(r_i, h_j) \in M$ where r_i has a single partner set $x_{i,p} = 1$, where $h_j = \mathit{pref}(r_i, p)$. Further, for each r_i in I where $|M(r_i)|= 2$, and thus r_i is assigned to a paired application, let $h_j \in H$ and $h_k \in H$ be the two distinct partners of r_i, $x_{i,p} = 1$, where $\mathit{pref}(r_i, p) = \{h_j, h_k\}$.

For each $h_j \in H$ set $y_j \in Y$ to the score limit at hospital h_j in I. Further, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ if $|\mathit{pref}(r_i, p)|= 1$, then $\tau_{i,p} = 0$. Otherwise $|\mathit{pref}(r_i, p)|= 2$. Let $\mathit{pref}(r_i, p) = \{h_j, h_k\}$. Now, if either $s_{i,j} \geq y_j$ or $s_{i,k} \geq y_k$, then set $\tau_{i,p} = 1$ otherwise set $\tau_{i,p} = 0$.

We now show that $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$ satisfies all of the constraints in the model. As each resident is assigned to at most one of his acceptable applications or is unassigned (but not both), for a given i $(1 \leq i \leq n_1)$, it follows that $x_{i,p} = 1$ for at most one value of p in the range $1 \leq p \leq l(r_i)$, and for each other value of p in the same range, $x_{i,p} = 0$. Hence, Constraints 8.1 and 8.2 are satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$. Since each hospital is assigned in M to at most c_j acceptable residents, Constraint 8.9 is also satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$.

Since M is the assignment induced in I from $\hat{t}$, each resident is assigned to the first application on his preference list at which he meets the score limit of each hospital involved in the application at that position. Thus Constraints 8.10, 8.11, 8.12 and 8.13 must be satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$.

Since $\hat{t}$ is a set of H-stable score limits in I it follows that no hospital in I can reduce its score limit further without exceeding its capacity. Further since $\hat{t}$ is a minimal set of H-stable score limits in I then $\sum_{t \in \hat{t}} t$ is minimal taken over the all of the sets of H-stable score limits in I and thus $\sum_{y \in Y} y$ is minimal taken over the feasible solutions in J. Hence the objective function in Constraint 8.14 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$. Thus we have that a minimal set of H-stable score limits $\hat{t}$ in I and the assignment induced in I from $\hat{t}$ give rise to an optimal solution to J.

Conversely, consider an optimal solution, $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$, to J. From such a solution we form in I a set M of pairs and a set $\hat{t}$ of score limits as follows. Initially let $M = \emptyset$. For each

i ($1 \le i \le n_1$) and p ($1 \le p \le l(r_i)$) either $pref(r_i, p) = h_j$ for some $h_j \in H$ or $pref(r_i, p) = \{h_j, h_k\}$ for some h_j, h_k where $h_j \neq h_k$. If $x_{i,p} = 1$ and $pref(r_i, p) = h_j$, then add (r_i, h_j) to M, otherwise if $x_{i,p} = 1$ and $pref(r_i, p) = \{h_j, h_k\}$, add both (r_i, h_j) and (r_i, h_k) to M. Further, for each $y_j \in Y$ set the score limit at hospital h_j in $\hat{t}$ to be $y_j \in Y$.

As $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$ satisfies Constraints 8.1, 8.2 and 8.9 each resident in M must be assigned to exactly one application or be unassigned (but not both) and each hospital h_j in M must have at most c_j assignees.

We now show that $\hat{t}$ is a set of H-stable score limits in I. Assume an applicant r_i is assigned in M to an application involving a hospital h_j at which he does not achieve the score limit. Hence, $(1-x_{i,p})\bar{s}_j = 0$ where $h_j \in pref(r_i, p)$ and further $s_{i,j} < y_j$ and thus Constraint 8.10 is not satisfied in J, a contradiction. Now assume that some r_i is not assigned to a hospital h_j where $pref(r_i, p)$ is a single hospital h_j at which they achieved the score limit and further r_i is not assigned to h_j or to a better hospital. Hence, $(\bar{s}_j + 1)\sum_{p'=1}^{p} x_{i,p'} = 0$ and $s_{i,j} \ge y_j$, and thus Constraint 8.11 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$, a contradiction.

Now assume that some r_i is not assigned to an application at position p or better on his preference list, where $pref(r_i, p) = \{h_j, h_k\}$. Further assume that r_i achieves the score limit at both h_j and h_k. Since r_i is not assigned to the application at position p or to a better application we have that in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$, $(\bar{s}_j + 1)\sum_{p'=1}^{p} x_{i,p'} = 0$ and since r_i meets the score limit at h_j, $s_{i,j} \ge y_j$. Now assume $\tau_{i,p} = 0$. Then Constraint 8.12 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$, a contradiction. Thus $\tau_{i,p} = 1$. However, we now have that $(\bar{s}_k + 1)\sum_{p'=1}^{p} x_{i,p'} = 0$, $s_{i,k} \ge y_k$ and $(1 - \tau_{i,p}) = 0$ and Constraint 8.13 is not satisfied in J, a contradiction.

Since the objective function Constraint 8.14 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\tau}$ $\rangle$, no hospital may reduce its score limit without exceeding its capacity and thus $\hat{t}$ is a set of H-stable score limits in I and moreover, M is the assignment induced in I from $\hat{t}$. Further, the objective function ensures that $\hat{t}$ is a minimal set of H-stable score limits in I and the result is proven. □

8.4 A Type A IP model for HR CQ SLT

We now present a model that extends the one described in Section 8.2 to allow for the possibility that coalitions of hospitals may share common upper quotas. We describe the variables and constraints in the IP model for finding a minimal set of H-stable score limits in an instance of HR CQ SLT in Sections 8.4.1 and 8.4.2 respectively and in Section 8.4.3 we prove the correctness of the model.

8.4.1 Variables in HR CQ SLT A

Let I be an instance of HR CQ SLT as described in Section 2.7.4. Further, let J be the following Integer Programming (IP) formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that:

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(r_i)\}$. Let $Y = \{y_1, y_2 \ldots, y_{n_3}\}$ be a set of variables where intuitively y_k is the score limit at H_k in J for $1 \leq k \leq n_3$.

Now, for all i $(1 \leq i \leq n_1)$ and k $(1 \leq k \leq n_3)$ define a new variable $\theta_{i,k} \in \{0, 1\}$. The intuitive meaning of $\theta_{i,k}$ is as follows. If r_i does not meet the score limit at some $h_j \in H_k$, then $\theta_{i,k} = 0$, otherwise $\theta_{i,k} \in \{0, 1\}$. However for the coalition $H_k = \{h_j\}$, $\theta_{i,k} = 0$ if r_i is admitted to h_j or to a better partner. Constraints 8.18 and 8.19 described in the Section 8.4.2 enforce the properties required of $\theta_{i,k}$.

8.4.2 Constraints in HR CQ SLT A

The HR CQ SLT A model is constructed by applying Constraints 8.1 and 8.2 from the HR SLT A model described in Section 8.2.2 in addition to the constraints described below.

The following constraint simply confirms that each variable $\theta_{i,k}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and k $(1 \leq k \leq n_3)$:

$$\theta_{i,k} \in \{0, 1\} \tag{8.15}$$

Since the hospitals in any coalition in $H_k \in H^*(1 \leq k \leq n_3)$ may have at most u_k assignees, $x_{i,p} = 1$ where $pref(r_i, p) = h_j$ and $h_j \in H_k$ for at most u_k residents. We thus obtain the following constraint for all k $(1 \leq k \leq n_3)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : pref(r_i, p) = h_j \wedge h_j \in H_k\} \leq u_k \tag{8.16}$$

The following constraint ensures that any r_i assigned to a hospital h_j where $h_j \in H_k$ $(1 \leq k \leq n_3)$ achieves a score of at least y_k at H_k. For all i $(1 \leq i \leq n_1)$ and for all p $(1 \leq p \leq l(r_i))$, let $pref(r_i, p) = h_j$. Then for all k $(1 \leq k \leq n_3)$ such that $h_j \in H_k$ we obtain the following constraint:

$$y_k \leq (1 - x_{i,p})(\bar{s}_k + 1) + S_{i,k} \tag{8.17}$$

For each $h_j \in H$ let $N_j \subseteq H^*$ be the set of coalitions involving h_j. We wish to enforce that $\theta_{i,k} = 0$ if r_i achieves a score of less than y_k for some coalition $H_k \in H^*$, and $\theta_{i,k} = \{0, 1\}$ otherwise. However for the coalition $H_k = \{h_j\}$, $\theta_{i,k} = 0$ if r_i is admitted to h_j or to a better hospital. The following two constraints are applied to ensure this property. Thus, for all i $(1 \leq i \leq n_1)$ and for all p $(1 \leq p \leq l(r_i))$, we obtain the following constraint where $pref(r_i, p) = h_j$:

$$\sum_{\substack{1 \leq k \leq n_3 \\ h_j \in H_k}} \theta_{i,k} \leq |N_j| - 1 \tag{8.18}$$

The following constraint ensures that any r_i who is not assigned to a hospital h_j either did not achieve the score limit of a coalition H_k containing h_j or is assigned to a better hospital. For all i $(1 \leq i \leq n_1)$ and for all p $(1 \leq p \leq l(r_i))$, let $pref(r_i, p) = h_j$. Then for all k $(1 \leq k \leq n_3)$ such that $h_j \in H_k$ we obtain the following constraint:

$$S_{i,k} + 1 \leq (\bar{S}_k + 1)\left(\sum_{p'=1}^{p} x_{i,p'} + \theta_{i,k}\right) + y_k \tag{8.19}$$

Objective Function: A minimal set of H-stable score limits in I is a set $\hat{t}$ of H-stable score limits in I where $\sum_{t \in \hat{t}} t$ is minimal over all sets of H-stable score limits in I. We apply the following objective function to ensure that an optimal solution meets this criterion:

$$\min \sum_{k=1}^{n_3} y_k \tag{8.20}$$

8.4.3 Proof of correctness of the constraints in HR CQ SLT A

Theorem 8.4.1. *Given an instance I of HR CQ SLT, let J be the corresponding IP model as defined in Sections 8.4.1 and 8.4.2. A minimal set of H-stable score limits $\hat{t}$ in I and the assignment induced in I from $\hat{t}$ are exactly equivalent to an optimal solution to J.*

Proof. Let I be an instance of HR CQ SLT. Let $\hat{t}$ be a minimal set of H-stable score limits in I and let M be the assignment induced in I from $\hat{t}$. We form an assignment of values $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$ to the variables in J as follows. Initially set $x_{i,p} = 0$ for all i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$. Then for each $(r_i, h_j) \in M$ set $x_{i,p} = 1$, where $pref(r_i, p) = h_j$. For each $H_k \in H^*$ set $y_k \in Y$ to the score limit of coalition H_k in I. For a given hospital $h_j \in H$, let $N_j \subseteq H^*$ be the set of coalitions involving h_j.

For each $H_k \in N_j$, if r_i does not achieve the score limit at H_k, set $\theta_{i,k} = 0$. If $H_k = \{h_j\}$, r_i achieves the score limit at H_k and r_i is assigned to h_j or to a better partner, set $\theta_{i,k} = 0$. Otherwise set $\theta_{i,k} = 1$

We now show that $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$ satisfies all of the constraints in the model. As each resident is assigned exactly once or is unassigned (but not both), for a given i $(1 \leq i \leq n_1)$, it follows that $x_{i,p} = 1$ for at most one value of p in the range $1 \leq p \leq l(r(i))$, and for each other value of p in the same range, $x_{i,p} = 0$. Hence, Constraints 8.1 and 8.2 are satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$. Since each coalition of hospitals $H_k \in H^*$ is assigned at most u_k acceptable residents in M, Constraint 8.16 is also satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$.

Since M is the assignment induced in I from $\hat{t}$, each resident is assigned in M to the first hospital on his preference list at which he meets the score limit at every coalition containing that hospital and thus Constraint 8.17 must be satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$.

We now demonstrate that Constraint 8.18 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$. Let i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ be given. Let $pref(r_i, p) = h_j$. Firstly suppose r_i is assigned to h_j or to a better hospital. Let $H_k = \{h_j\}$. Then $\theta_{i,k} = 0$ by construction of θ. Thus Constraint 8.18 is satisfied. Now suppose r_i is unassigned or assigned to a worse hospital than h_j. Then r_i does not meet the score limit at some coalition H_k containing h_j, by construction of M from $\hat{t}$. Thus, $\theta_{i,k} = 0$, by construction of θ. Hence Constraint 8.18 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$.

By the stability of $\hat{t}$ any resident who is not assigned to some $h_j \in H$ is either assigned to a better hospital or does not meet the score limit of some coalition containing h_j. Let i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$ be given. Let $pref(r_i, p) = h_j$ and let H_k be any coalition containing h_j. Now, if r_i is assigned to h_j or to a better hospital, then $\sum_{p'=1}^{p} x_{i,p'} = 1$ and thus $(\bar{S}_k + 1)(\sum_{p'=1}^{p} x_{i,p'} + \theta_{i,k}) = (\bar{S}_k + 1)$ and Constraint 8.19 is satisfied in J.

Otherwise, if r_i is unassigned or assigned to a worse hospital than h_j, then r_i fails to achieve the score limit at some H_k containing h_j. Hence $\theta_{i,k} = 0$ and moreover $(\bar{S}_k + 1)(\sum_{p'=1}^{p} x_{i,p'} + \theta_{i,k}) = 0$. Now we have that $S_{i,k} + 1 \leq y_k$ and Constraint 8.19 is satisfied in J.

Since $\hat{t}$ is a set of H-stable score limits in I then no coalition in I can reduce its score limit further without exceeding its capacity. Further since $\hat{t}$ is a minimal set of H-stable score limits in I then $\sum_{t \in \hat{t}} t$ is minimal over the sets of H-stable score limits in I and thus $\sum_{y \in Y} y$ is minimal over the feasible solutions in J. Hence the objective function in Expression 8.20 is satisfied. Thus we have that a minimal set of H-stable score limits $\hat{t}$ in I and the assignment induced in I from $\hat{t}$ are equivalent to an optimal solution to J

Conversely, consider an optimal solution $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$, to J. Thus in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$, $\sum_{y \in Y} y$ is minimal over all feasible solutions to J. From such a solution we form in I a set of pairs, M, and a set of score limits $\hat{t}$ as follows. Initially let $M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and

p $(1 \leq p \leq l(r_i))$ if $x_{i,p} = 1$, then add (r_i, h_j) to M where $h_j = \mathit{pref}(r_i, p)$. Further, for each $H_k \in H^*$ set the score limit of coalition H_k in $\hat{t}$ to be $y_k \in Y$.

As $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$ satisfies Constraints 8.1, 8.2 and 8.16, each resident in M must have exactly one partner or be unassigned (but not both) and each coalition H_k in M must have at most u_k partners.

We now show that $\hat{t}$ is a set of H-stable score limits in I. Assume an applicant r_i is assigned to a hospital h_j where $\mathit{pref}(r_i, p) = h_j$. Further assume that r_i does not achieve the score limit at some coalition H_k containing h_j. Hence, both $(1 - x_{i,p})(\bar{S}_k + 1) = 0$ and $S_{i,k} < y_k$ and thus Constraint 8.17 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$, a contradiction.

Now assume that some r_i is unassigned or assigned to a worse hospital than h_j and moreover that r_i achieved the score limit at each $H_k \in H^*$ where $h_j \in H_k$. Thus $\sum_{p'=1}^{p} x_{i,p'} = 0$, and for each $H_k \in H^*$ such that $h_j \in H_k$, $S_{i,k} \geq y_k$ and hence $\theta_{i,k} = 1$ by Constraint 8.19. Hence, Constraint 8.18 is violated, a contradiction. Hence, r_i does not achieve the score limit at some $H_k \in H^*$ such that $h_j \in H_k$.

Since the objective function Expression 8.20 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\theta}$ $\rangle$, no coalition may reduce its score limit without exceeding its capacity and thus $\hat{t}$ is a set of H-stable score limits in M and moreover, M is the assignment induced in I from $\hat{t}$. Further, the objective function ensures that $\hat{t}$ is a minimal set of H-stable score limits in I and the result is proven. □

8.5 A Type B IP model for HR SLT with a free objective function (HR SLT B)

In the models presented in Sections 8.2, 8.3 and 8.4, the objective function is required to ensure the stability of the set of score limits output as a solution. Thus the objective function cannot be applied to select some other optimality criterion from the feasible solution other than minimality. In this section we present an updated model in which the objective function is not necessary to ensure the stability of the score limits output as a solution. Thus we may apply the objective function to select a set of score limits which produces a solution that is optimal according to some other criteria. In this case the objective function is applied to ensure that a maximum cardinality induced assignment is the solution output. A maximum cardinality induced assignment is only one possible objective; by choosing other objective functions we might obtain other optimal solutions, e.g., a rank-based maximum weight solution or a college-optimal solution.

In Sections 8.5.1 and 8.5.2 respectively we describe the variables and constraints in the IP model for finding a set of H-stable score limits $\hat{t}$ in an instance of HR SLT such that the assignment induced from $\hat{t}$ is of maximum cardinality taken over all of the sets of H-stable score limits admitted by the instance. In Section 8.5.3 we prove the correctness of the model.

8.5.1 Variables in HR SLT B

Let I be an instance of HR SLT as described in Section 2.7. Let J be the following Integer Programming (IP) formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that:

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(r_i)\}$. Let $Y = \{y_1, y_2 \ldots, y_{n_2}\}$ be a set of variables where intuitively y_j is the score limit of h_j in J for $1 \leq j \leq n_2$.

Now, for all j $(1 \leq j \leq n_2)$ define a new variable $\gamma_j \in \{0, 1\}$. The intuitive meaning of γ_j is that if h_j is has a score limit of greater than zero, then $\gamma_j = 1$. However, if h_j has a score limit of zero, then γ_j may take the value of 0 or 1, Constraint 8.23 described in full in Section 8.5.2 enforces this property.

Now, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new variable $\delta_{i,p} \in \{0, 1\}$. The intuitive meaning of $\delta_{i,p}$ is that if r_i has an acceptable partner h_j preferable to their current assignment, to whom they would be assigned were h_j to reduce its score limit by 1 then $\delta_{i,p} \in \{0, 1\}$. Otherwise, $\delta_{i,p} = 0$. Constraints 8.24 and 8.25 described in full in Section 8.5.2 enforce this property.

For ease of exposition we define some additional notation. For an acceptable resident-hospital pair (r_i, h_j), let $rank(h_j, r_i) = q$ denote the rank that hospital h_j assigns resident r_i, where $1 \leq j \leq n_2$, $1 \leq i \leq n_1$ and $1 \leq q \leq l(h_j)$. Further, for each j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(h_j))$ let the set $R(h_j, q)$ contain the resident-position pairs (r_i, p) such that r_i is assigned a rank of q $(1 \leq q \leq l(h_j))$ by h_j and h_j is in position p $(1 \leq p \leq l(r_i))$ on r_i's preference list. Hence:

$$R(h_j, q) = \{(r_i, p) \in R \times \mathbb{Z} : rank(h_j, r_i) = q \wedge 1 \leq p \leq l(r_i) \wedge pref(r_i, p) = h_j\}$$

8.5.2 Constraints in HR SLT B

This IP formulation for HR SLT is constructed by applying Constraints 8.1, 8.2, 8.3, 8.4 and 8.5 from the model described in Section 8.2.2 in addition to the constraints described below.

The following constraint simply ensures that each variable γ_j must be binary valued for all j $(1 \leq j \leq n_2)$:

$$\gamma_j \in \{0, 1\} \tag{8.21}$$

The following constraint simply ensures that each variable $\delta_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$:

$$\delta_{i,p} \in \{0, 1\} \tag{8.22}$$

We now apply constraints to enforce that the variable γ_j has the following properties. If h_j is has a score limit of greater than zero, then $\gamma_j = 1$. Otherwise, h_j has a score limit of zero and γ_j may take the value of 0 or 1. Thus, for all j $(1 \leq j \leq n_2)$ define a new constraint such that:

$$\gamma_j \geq \frac{y_j}{\bar{s}_j + 1} \tag{8.23}$$

We now apply constraints to enforce that the variable $\delta_{i,p}$ has the following properties. If r_i has an acceptable partner h_j preferable to their current assignment, to whom they would be assigned were h_j to reduce its score limit by 1 then $\delta_{i,p} \in \{0, 1\}$. Otherwise, $\delta_{i,p} = 0$. First we apply a constraint to ensure that $\delta_{i,p'} = 0$ where $M(r_i)$, the assigned partner of r_i in the assignment induced from a set of score limits, is $pref(r_i, p)$ and $p \leq p' \leq l(r_i)$. Thus, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new constraint such that:

$$\sum_{p'=p}^{l(r_i)} \delta_{i,p'} \leq (1 - x_{i,p})l(r_i) \tag{8.24}$$

Now we apply a constraint to ensure that it can only be the case that $\delta_{i,p} = 1$ if r_i has an acceptable hospital h_j to whom they could be assigned if h_j reduced their score limit by 1. Thus, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new constraint such that:

$$y_j - 1 \leq (1 - \delta_{i,p})\bar{s}_j + s_{i,j} \tag{8.25}$$

Now we apply a constraint to ensure that the set of score limits $\hat{t}$ obtained from the model is such that if some hospital h_j reduced its score limit below its value in $\hat{t}$ to produce a new set of score limits $\hat{t}^j$, then h_j would be oversubscribed in the assignment induced in I from $\hat{t}^j$. Thus, for all j $(1 \leq j \leq n_2)$ we define a new constraint such that:

$$(c_j + 1)(1 - \gamma_j) + \sum_{q'=1}^{l(h_j)} \{x_{i',p'} + \delta_{i',p'} : x_{i',p'} \in X \wedge (r_{i'}, p') \in R(h_j, q')\} \geq c_j + 1 \tag{8.26}$$

We seek a set of H-stable score limits $\hat{t}$ such that the assignment induced in I from $\hat{t}$ is of maximum cardinality taken over all of the assignments induced from all of the sets of H-stable score limits in I. To ensure this we apply the following objective function:

$$\max \sum_{i=1}^{n} \sum_{p=1}^{l(r_i)} x_{i,p} \tag{8.27}$$

8.5.3 Proof of correctness of the constraints in HR SLT B

We now establish the correctness of the IP model presented in Sections 8.5.1 and 8.5.2.

Theorem 8.5.1. *Given an instance I of HR SLT, let J be the corresponding* HR SLT B *IP model as defined in Sections 8.5.1 and 8.5.2. A set of H-stable score limits $\hat{t}$ in I and the assignment induced in I from $\hat{t}$ are exactly equivalent to a feasible solution to J.*

Proof. Let I be an instance of HR SLT. Let $\hat{t}$ be a set of H-stable score limits in I and let M be the assignment induced in I from $\hat{t}$. Let $\hat{t^j} = \{t_1^j, t_2^j, \ldots, t_{n_2}^j\}$ be the set of score limits obtained by reducing the score limit at one $h_j \in H$ by exactly one and leaving all other score limits unchanged. Thus $t_k^j = t_k$ for all k $(1 \leq k \leq n_2 \wedge j \neq k)$ and $t_j^j = t_j - 1$.

We form an assignment of values J to the variables $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$ as follows. Initially set $x_{i,p} = 0$ for all i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$. Then for each $(r_i, h_j) \in M$ set $x_{i,p} = 1$, where $h_j = \mathit{pref}(r_i, p)$. For each $h_j \in H$ set $y_j \in Y$ to the score limit at hospital h_j in I. Further, for each i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$ where $\mathit{pref}(r_i, p) = h_j$, if r_i has a hospital h_j preferable to $M(r_i)$ to which they would be assigned in the assignment induced in I from $\hat{t^j}$, then $\delta_{i,p} = 1$. Otherwise $\delta_{i,p} = 0$. For each $h_j \in H$, if h_j has a score limit of greater than zero, then we set $\gamma_j = 1$. Otherwise set $\gamma_j = 0$.

We now show that $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$ satisfies all of the constraints in the model. As each resident has a single partner or is unassigned (but not both), for a given i $(1 \leq i \leq n_1)$, it follows that $x_{i,p} = 1$ for at most one value of p in the range $1 \leq p \leq l(r(i))$, and for each other value of p in the same range, $x_{i,p} = 0$. Hence, Constraints 8.1 and 8.2 are satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. Since each hospital is assigned in M to at most c_j acceptable residents, Constraint 8.3 is also satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. Since M is the assignment induced in I from $\hat{t}$ each resident is assigned in M to the first hospital on his preference list at which he meets the score limit. Thus Constraints 8.4 and 8.5 must be satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. Constraint 8.23 is trivially satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$.

If r_i prefers $M(r_i)$ to h_j where $\mathit{pref}(r_i, p) = h_j$, then r_i could not be assigned to h_j in the assignment induced in I from $\hat{t^j}$ and thus $\delta_{i,p} = 0$. Hence, Constraint 8.24 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. Further, if r_i does not achieve a score of at least $y_j - 1$, then r_i cannot be assigned to h_j in the assignment induced from $\hat{t}^j$ and $\delta_{i,p} = 0$. Hence, Constraint 8.25 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$.

Now we consider whether Constraint 8.26 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. Since Constraint 8.23 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, either $\gamma_j = 0$ or $\gamma_j = 1$. If $\gamma_j = 1$, then $(c_j+1)(1-\gamma_j) = 0$. Further, since $\hat{t}$ is a set of H-stable score limits in I, h_j must be oversubscribed in the assignment induced in I from $\hat{t}^j$ and $\sum_{q'=1}^{l(h_j)}\{x_{i',p''} + \delta_{i',p''} : x_{i',p''} \in X \wedge (r_{i'}, p'') \in R(h_j, q')\} \geq c_j + 1$ in J. Hence Constraint 8.26 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. If h_j has a score limit of zero in $\hat{t}$, then $\gamma_j = 0$ and thus $(c_j + 1)(1 - \gamma_j) = (c_j + 1)$ and Constraint 8.26 is trivially satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$. Hence a set $\hat{t}$ of H-stable score limits in I, $\hat{t}$, and the assignment induced in I from $\hat{t}$ give rise to a feasible solution in J.

Conversely, consider a feasible solution, $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, to J. From such a solution we form in I a set of pairs, M, and a set of score limits $\hat{t}$ as follows. Initially let $M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ if $x_{i,p} = 1$, then add (r_i, h_j) to M where $h_j = \mathit{pref}(r_i, p)$. Further, for each $y_j \in Y$ set the score limit at hospital h_j in $\hat{t}$ to be $y_j \in Y$.

As $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$ satisfies Constraints 8.1, 8.2 and 8.3 each resident in I must have exactly one partner or be unassigned (but not both) and each hospital h_j in M must have at most c_j partners. We now show that $\hat{t}$ is a set of H-stable score limits in I. Assume an applicant r_i is assigned to a hospital h_j at which he does not achieve the score limit. Assume that $\mathit{pref}(r_i, p) = h_j$. Hence, both $(1 - x_{i,p})\bar{s}_j = 0$ and $s_{i,j} < y_j$. Hence Constraint 8.4 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, a contradiction. Now assume that some r_i is not assigned to a hospital h_j at which they achieved the score limit and moreover r_i is not assigned to a better hospital than h_j. Hence, $(\bar{s}_j + 1)\sum_{p'=1}^{p} x_{i,p'} = 0$ and $s_{i,j} \geq y_j$, and thus Constraint 8.5 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, a contradiction.

Assume that a hospital h_j has a score limit of greater than zero. Further assume that $\gamma_j = 0$. Now since $y_j > 0$, we have that $\dfrac{y_j}{\bar{s}_j + 1} > 0$ and Constraint 8.23 is not satisfied in J, a contradiction. Hence, if a hospital h_j has a score limit of greater than zero, then $\gamma_j = 1$.

Let i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ be given such that $x_{i,p} = 1$ and let $\mathit{pref}(r_i, p) = h_j$. Assume that $\delta_{i,p'} = 1$ for some $x_{i,p'}$ where $p \leq p' \leq l(r_i)$. Now we have that $\sum_{p'=p}^{l(r_i)} \delta_{i,p'} > 0$ and $(1 - x_{i,p})l(r_i) = 0$ and Constraint 8.24 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, a contradiction. Thus $\delta_{i,p'} = 0$ for all $p \leq p' \leq l(r_i)$. Now, assume that $\delta_{i,p} = 1$ and r_i has a score of less than $y_j - 1$ at h_j. Thus, we have that $(1 - \delta_{i,p})\bar{s}_j + s_{i,j} = s_{i,j}$ and Constraint 8.25 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, a contradiction.

Let $\hat{t}^j$ be the set of score limits obtained from $\hat{t}$ by reducing the score limit at h_j by one and leaving all other score limits unchanged. Since $\hat{t}$ is a set of H-stable score limits h_j must be over-subscribed in the assignment induced in I from $\hat{t}^j$. Assume that h_j is not oversubscribed in the assignment induced in I from $\hat{t}^j$. Then $\sum_{q'=1}^{l(h_j)}\{x_{i',p''} + \delta_{i',p''} : x_{i',p''} \in X \wedge (r_{i'}, p'') \in$

$R(h_j, q')\} \leq c_j$. Moreover, h_j must have a score limit of one in $\hat{t}$, otherwise it could not reduce its score limit. Hence $\gamma_j \geq 1$. Now we have that $(c_j + 1)(1 - \gamma_j) = 0$ and thus Constraint 8.26 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ $\rangle$, a contradiction. Hence, a feasible solution in J gives rise to a set of H-stable score limits $\hat{t}$ and the assignment induced in I from $\hat{t}$ and the result is proven. □

The existence of the objective function (given by Expression 8.27) immediately leads to the following corollary.

Corollary 8.5.2. *Given an instance I of* HR SLT*, let J be the corresponding* HR SLT B *IP model as defined in Sections 8.5.1 and 8.5.2. A set of H-stable score limits $\hat{t}$ in I such that the assignment induced in I from $\hat{t}$ assigns the maximum number of residents taken over all sets of H-stable score limits in I is exactly equivalent to an optimal solution to J.*

8.6 A Type B IP model for HR LQ SLT with a free objective function (HR LQ SLT B)

We now show how the model with free objective function presented in Section 8.5 may be adapted to find a set of H-stable score limits in the HR LQ SLT context. Moreover, since the objective function in this model is no longer a necessary part of ensuring the stability of the solutions found we can apply an objective function to ensure that an optimal solution is equivalent to a maximum cardinality induced assignment taken over all of the sets of H-stable score limits.

In Sections 8.6.1 and 8.6.2 respectively we describe the variables and constraints in the IP model for finding a set of H-stable score limits $\hat{t}$ in an instance of HR LQ SLT such that the assignment induced from $\hat{t}$ is of maximum cardinality taken over all of the sets of H-stable score limits admitted by the instance. In Section 8.6.3 we prove the correctness of the model.

8.6.1 Variables in HR LQ SLT B

Let I be an instance of HR LQ SLT as described in Section 2.7.3. Let J be the following Integer Programming (IP) formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$, define a variable $x_{i,p}$ such that:

$$x_{i,p} = \begin{cases} 1 & \text{if } r_i \text{ is assigned to their } p^{th} \text{ choice hospital} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(r_i)\}$. Let $pref(r_i, p)$ denote the hospital at position p of r_i's preference list where $1 \leq i \leq n_1$ and $1 \leq p \leq l(r_i)$. Let $Y = \{y_1, y_2 \ldots, y_{n_2}\}$ be a set of variables where intuitively y_j is the score limit of h_j in J for $1 \leq j \leq n_2$.

For all j $(1 \leq j \leq n_2)$ define a new variable $\beta_j \in \{0, 1\}$. The intuitive meaning of β_j is that if h_j is closed and $c_j^- \geq 1$, then $\beta_j = 0$. Otherwise, h_j has at least $\min\{1, c_j^-\}$ or greater assignees and thus h_j is open and $\beta_j = 1$. Constraints 8.33 and 8.34 described in full in Section 8.6.2 enforce this property.

Further, for all j $(1 \leq j \leq n_2)$ define a new variable $\gamma_j \in \{0, 1\}$. The intuitive meaning of γ_j is that if h_j has a score limit of greater than zero, then $\gamma_j = 1$. However, if h_j has a score limit of zero, then γ_j may take the value of 0 or one. Constraint 8.23 described in full in Section 8.5.2 enforces this property.

For all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new variable $\delta_{i,p} \in \{0, 1\}$. The intuitive meaning of $\delta_{i,p}$ is that if r_i has an acceptable partner h_j to whom they would be assigned if h_j reduced their score limit by one, where $h_j = pref(r_i, p)$, such that $M(r_i) = pref(r_i, p^1)$ and $p < p^1$ or such that r_i is unassigned, then $\delta_{i,p} = \{0, 1\}$. Otherwise, $\delta_{i,p} = 0$. Constraints 8.24 and 8.25 described in full in Section 8.5.2 enforce this property.

Further, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new variable $\omega_{i,p} \in \{0, 1\}$. The intuitive meaning of $\omega_{i,p}$ is that if r_i meets the score limit at h_j, then $\omega_{i,p} = 1$. Otherwise, $\omega_{i,p}$ may take a value of zero or one. Constraint 8.36 described in full in Section 8.5.2 enforces this property.

Further, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ define a new variable $\phi_{i,p} \in \{0, 1\}$. The intuitive meaning of $\phi_{i,p}$ is that if r_i meets the score limit at h_j *and* r_i is not assigned to a better hospital than h_j, then $\phi_{i,p} = 1$. Otherwise, $\phi_{i,p} = 0$. Constraints 8.37, 8.38 and 8.39 described in full in Section 8.5.2 enforce this property.

For all j $(1 \leq i \leq n_2)$ define a new variable $\lambda_j^- \in \{0, 1\}$. The intuitive meaning of λ_j^- is that if h_j has less than c_j^- acceptable partners who achieve the score limit at h_j and prefer h_j to their current partner then $\lambda_j^- = 1$. Otherwise $\lambda_j^- \in \{0, 1\}$. Further, for all j $(1 \leq i \leq n_2)$ define a new variable $\lambda_j^+ \in \{0, 1\}$. The intuitive meaning of λ_j^+ is that if h_j has more than c_j^+ acceptable partners who achieve the score limit at h_j and prefer h_j to their current partner, then $\lambda_j^+ = 1$. Otherwise $\lambda_j^+ \in \{0, 1\}$.

8.6.2 Constraints in HR LQ SLT B

The HR LQ SLT model is constructed by applying Constraints 8.1, 8.21, 8.22, 8.2, 8.4, 8.23, 8.24, 8.25 and 8.26 from the model described in Section 8.5.2 in addition to the constraints described below.

The following constraint simply ensures that each variable β_j must be binary valued for all j $(1 \leq j \leq n_2)$:

$$\beta_j \in \{0, 1\} \tag{8.28}$$

The following constraint simply ensures that each variable $\omega_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$:

$$\omega_{i,p} \in \{0, 1\} \tag{8.29}$$

The following constraint simply ensures that each variable $\phi_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$:

$$\phi_{i,p} \in \{0, 1\} \tag{8.30}$$

The following constraint simply ensures that each variable λ_j^- must be binary valued for all j $(1 \leq j \leq n_2)$:

$$\lambda_j^- \in \{0, 1\} \tag{8.31}$$

The following constraint simply ensures that each variable λ_j^+ must be binary valued for all j $(1 \leq j \leq n_2)$:

$$\lambda_j^+ \in \{0, 1\} \tag{8.32}$$

Constraint 8.3 from the HR SLT model in Section 8.5.2 is no longer sufficient to ensure the hospital capacity constraints in the HR LQ SLT context. Now we apply both Constraint 8.33, to confirm that no hospital exceeds its capacity, and Constraint 8.34 to confirm that no open hospital has fewer than c_j^- assignees in an induced assignment derived from a feasible solution to the model. Since a hospital h_j may be assigned to at most c_j^+ residents, $x_{i,p} = 1$ where $pref(r_i, p) = h_j$ for at most c_j^+ residents. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : pref(r_i, p) = h_j\} \leq \beta_j c_j^+ \tag{8.33}$$

Since an open hospital h_j must be assigned at least c_j^- residents, $x_{i,p} = 1$ where $pref(r_i, p) = h_j$ for at least c_j^- residents. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : pref(r_i, p) = h_j\} \geq \beta_j c_j^- \tag{8.34}$$

In the HR LQ SLT context we must adapt Constraint 8.5 from the HR SLT model in Section 8.5.2 to consider the circumstance where a hospital might be closed. The following constraint ensures that any r_i who is not assigned to an open hospital h_j either did not achieve the score limit of h_j or is assigned to a better hospital. Thus, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ we obtain the following constraint where $pref(r_i, p) = h_j$:

$$\beta_j(s_{i,j} + 1) \leq (\bar{s}_j + 1) \sum_{p'=1}^{p} x_{i,p'} + y_j \tag{8.35}$$

We apply the following group of constraints to ensure that the set of H-stable score limits derived from any feasible solution to J meets the group stability condition described in Section 2.6.1. We apply constraints to enforce that the variable $\omega_{i,p}$ has the following properties where $pref(r_i, p) = h_j$. If r_i meets the score limit at h_j, then $\omega_{i,p} = 1$. Otherwise, $\omega_{i,p}$ may take a value of zero or one. Thus, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ and $\varepsilon = 1/2$ define the following constraint where $pref(r_i, p) = h_j$:

$$\omega_{i,p} \geq \frac{s_{i,j} - y_j + \varepsilon}{\bar{s}_j} \tag{8.36}$$

We now apply constraints to enforce that the variable $\phi_{i,p}$ has the following properties where $pref(r_i, p) = h_j$. If r_i meets the score limit at hospital h_j *and* r_i is not assigned to a better hospital than h_j, then $\phi_{i,p} = 1$. Otherwise, $\phi_{i,p} = 0$. Thus, for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ where $pref(r_i, p) = h_j$ define the following constraints:

$$\phi_{i,p} \geq \omega_{i,p} + \left(1 - \sum_{p'=1}^{p-1} x_{i,p'}\right) - 1 \tag{8.37}$$

$$\phi_{i,p} \leq \omega_{i,p} \tag{8.38}$$

$$\phi_{i,p} \leq 1 - \sum_{p'=1}^{p-1} x_{i,p'} \tag{8.39}$$

We apply the following constraint to ensure that if h_j has less than c_j^- acceptable partners who achieve the score limit at h_j and prefer h_j to their current partner, then $\lambda_j^- = 1$ and $\lambda_j^- \in \{0, 1\}$ otherwise. For each j $(1 \leq j \leq n_2)$ we obtain the following constraint:

$$\lambda_j^- \geq \frac{c_j^- - \sum_{q=1}^{l(h_j)} \{\phi_{i,p} : (r_i, p) \in R(h_j, q)\}}{n_1} \tag{8.40}$$

We apply the following constraint to ensure that if h_j has more than c_j^+ acceptable partners who achieve the score limit at h_j and prefer h_j to their current partner, then $\lambda_j^+ = 1$ and $\lambda_j^+ \in \{0, 1\}$ otherwise. For each j $(1 \leq j \leq n_2)$ we obtain the following constraint:

$$\lambda_j^+ \geq \frac{\sum_{q=1}^{l(h_j)} \{\phi_{i,p} : (r_i, p) \in R(h_j, q)\} - c_j^+}{n_1} \tag{8.41}$$

We apply the following constraint to ensure that for any closed hospital h_j any coalition of residents who achieve the score limit at h_j and would prefer h_j to their current partner must be of size less c_j^- or of size greater than c_j^+. Thus, For each j $(1 \leq j \leq n_2)$ we obtain the following constraint:

$$\lambda_j^- + \lambda_j^+ \geq 1 - \beta_j \tag{8.42}$$

Objective Function - Recall that the instance shown in Figure 2.6 demonstrates that an instance of HR LQ may admit stable matchings of differing sizes. We seek a solution that maximises the number of residents assigned in the induced assignment from the set of H-stable score limits obtained. To maximise the number of residents assigned we apply the following objective function:

$$\max \sum_{i=1}^{n} \sum_{p=1}^{l(r_i)} x_{i,p} \tag{8.43}$$

8.6.3 Proof of correctness of the constraints in HR LQ SLT B

We now establish the correctness of the IP model presented in Sections 8.6.1 and 8.6.2.

Theorem 8.6.1. *Given an instance I of HR LQ SLT, let J be the corresponding IP model as defined in Sections 8.6.1 and 8.6.2. A set of H-stable score limits $\hat{t}$ in I and the assignment induced in I from $\hat{t}$ are exactly equivalent to a feasible solution to J.*

Proof. Let I be an instance of HR LQ SLT. Let $\hat{t}$ be a set of H-stable score limits in I and let M be the assignment induced in I from $\hat{t}$ in I. Let $\hat{t}^j = \{t_1^j, t_2^j, \ldots, t_{n_2}^j\}$ be the set of score limits obtained from $\hat{t}$ by reducing the score limit at one $h_j \in H$ by exactly one and leaving all other score limits unchanged. Thus $t^j_k = t_k$ for all k $(1 \leq k \leq n_2 \wedge j \neq k)$ and $t^j_j = t_j - 1$.

We form an assignment of values J to the variables $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$ as follows. Initially set $x_{i,p} = 0$ for all i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$. Then for each $(r_i, h_j) \in M$ set $x_{i,p} = 1$, where $h_j = \mathit{pref}(r_i, p)$. For each $h_j \in H$ set $y_j \in Y$ to the score limit at hospital h_j in I. Further, if h_j is closed in M, then set $\beta_j = 0$ otherwise $\beta_j = 1$. For each i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$ where $\mathit{pref}(r_i, p) = h_j$, if r_i has a hospital h_j to which they would be assigned in the assignment induced in I from $\hat{t}^j$, then $\delta_{i,p} = 1$. Otherwise $\delta_{i,p} = 0$. For each $h_j \in H$, if h_j has a score limit of greater than zero, then we set $\gamma_j = 1$. Otherwise set $\gamma_j = 0$. For each i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$ where $\mathit{pref}(r_i, p) = h_j$, if r_i meets the score limit at h_j, then set $\omega_{i,p} = 1$. Otherwise set $\omega_{i,p} = 0$. For each i and p $(1 \leq i \leq n_1, 1 \leq p \leq l(r_i))$ where $\mathit{pref}(r_i, p) = h_j$, if r_i meets the score limit at h_j and r_i is not assigned to a better hospital, then set $\phi_{i,p} = 1$. Otherwise, set $\phi_{i,p} = 0$.

For a given hospital h_j, if a coalition of h_j's acceptable residents of size less than c^-_j prefer h_j to their assigned partner, then set $\lambda^-_j = 1$. Otherwise $\lambda^-_j = 0$. Further, if a coalition of h_j's acceptable residents of size greater than c^+_j prefer h_j to their assigned partner, then set $\lambda^+_j = 1$. Otherwise $\lambda^+_j = 0$.

Theorem 8.5.1 proves that $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$ satisfies Constraints 8.1, 8.21, 8.22, 8.2, 8.4, 8.23, 8.24, 8.25 and 8.26. We now prove that $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$ satisfies the remaining constraints in the HR LQ SLT B model.

Now, every $h_j \in H$ is either open or closed. If h_j is open, then $\beta_j = 1$ and since $\hat{t}$ is a set of H-stable score limits in I and thus each hospital is assigned at most c^+_j acceptable residents in M then $\sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : \mathit{pref}(r_i, p) = h_j\} \leq c^+_j$ and Constraint 8.33 is satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$. Otherwise, h_j is closed (and thus has exactly zero assignees) and $\beta_j = 0$ and Constraint 8.33 is satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$. (Although this constraint still holds whether $\beta_j = 1$ or $\beta_j = 0$ if h_j has zero assignees).

Now, for every $h_j \in H$ such that $c^-_j = 0$ Constraint 8.34 is trivially satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$. For each $h_j \in H$ such that $c^-_j > 0$, h_j is either open or closed. If h_j is open (and thus has more than zero partners), then $\beta_j = 1$ and since h_j must have at least c^-_j partners in M, $\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : \mathit{pref}(r_i, p) = h_j\} \geq c^-_j$. Hence Constraint 8.34 is satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$. If h_j is closed and thus h_j has no assignees in M, then $\beta_j = 0$ and Constraint 8.34 is satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$.

Now, if h_j is closed, then $\beta_j = 0$ and Constraint 8.35 is trivially satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$. Otherwise h_j is open and $\beta_j = 1$. Now we have that, for some i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ where $\mathit{pref}(r_i, p) = h_j$, if r_i is assigned to h_j or to a better hospital, then $(\bar{s}_j + 1) \sum_{p'=1}^{p} x_{i,p'} = (\bar{s}_j + 1)$ and Constraint 8.35 is satisfied in $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$.

Now, for some i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ where $pref(r_i, p) = h_j$, if r_i meets the score limit at h_j, then clearly $s_{i,j} - y_j \geq 0$. Hence $s_{i,j} - y_j + 1/2 > 0$ and it follows that $\omega_{i,p} = 1$. If r_i does not meet the score limit at h_j, then $s_{i,j} - y_j \leq -1$ and it follows that $s_{i,j} - y_j + 1/2 < 0$ and $\omega_{i,p} \in \{0, 1\}$. Thus Constraint 8.36 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$.

Now, for some i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ where $pref(r_i, p) = h_j$, if r_i does not meet the score limit at h_j, then by construction $\omega_{i,p} = 0$ and $\phi_{i,p} = 0$. Hence, Constraints 8.37 and 8.38 are satisfied in J. If r_i is assigned to a better hospital than h_j, then $\phi_{i,p} = 0$ and $1 - \sum_{p'=1}^{p-1} x_{i,p'} = 0$. Hence, Constraints 8.37 and 8.39 are satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$.

Now, if r_i meets the score limit at h_j *and* r_i is not assigned to a better hospital than h_j, then by construction $\omega_{i,p} = 1$ and $\phi_{i,p} = 1$. Now since r_i meets the score limit at h_j it follows that Constraint 8.38 is satisfied in J. Further, as r_i is not assigned to a better hospital than h_j then $1 - \sum_{p'=1}^{p-1} x_{i,p'} = 1$ and Constraint 8.39 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$. Now we have that $\omega_{i,p} = 1$ and $1 - \sum_{p'=1}^{p-1} x_{i,p'} = 1$ and thus Constraint 8.37 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$.

For a given hospital h_j if a coalition of h_j's acceptable residents of size less than c_j^- prefer h_j to their assigned partner, then it follows that $c_j^- - \sum_{q=1}^{l(h_j)} \{\phi_{i,p} : (r_i, p) \in R(h_j, q)\} > 0$ and thus $\lambda_j^- = 1$. Hence Constraint 8.40 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$. Otherwise, a coalition of h_j's acceptable residents of size at least c_j^- prefer h_j to their assigned partner and it follows that $c_j^- - \sum_{q=1}^{l(h_j)} \{\phi_{i,p} : (r_i, p) \in R(h_j, q)\} \leq 0$ and thus $\lambda_j^- \in \{0, 1\}$. Hence Constraint 8.40 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$.

For a given hospital h_j if a coalition of h_j's acceptable residents of size greater than c_j^+ prefer h_j to their assigned partner, then it follows that $\sum_{q=1}^{l(h_j)} \{\phi_{i,p} : (r_i, p) \in R(h_j, q)\} - c_j^+ > 0$ and thus $\lambda_j^+ = 1$. Hence Constraint 8.41 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$. Otherwise, a coalition of h_j's acceptable residents of size at most c_j^+ prefer h_j to their assigned partner and it follows that $\sum_{q=1}^{l(h_j)} \{\phi_{i,p} : (r_i, p) \in R(h_j, q)\} - c_j^+ \leq 0$ and thus $\lambda_j^+ \in \{0, 1\}$. Hence Constraint 8.41 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$.

We now consider Constraint 8.42. For each $h_j \in H$, h_j is either open or closed. If h_j is open, then $\beta_j = 1$ and Constraint 8.42 is trivially satisfied in J. However, if h_j is closed, then $\beta_j = 0$. Since $\hat{t}$ is a set of H-stable score limits in I and M is the assignment induced in I from $\hat{t}$ any coalition of h_j's acceptable residents who either prefer h_j to their assigned partner in M or are unassigned must be of size less than c_j^-, in which case $\lambda_j^- = 1$, or of size greater than c_j^+, in which case $\lambda_j^+ = 1$. Thus $\lambda_j^- + \lambda_j^+ \geq 1$ and Constraint 8.42 is satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$.

Conversely, consider a feasible solution, $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_j^-$, $\boldsymbol{\lambda}_j^+$ $\rangle$, to J. From such a solution we form in I a set of pairs, M, and a set of score limits $\hat{t}$ as follows. Initially let

$M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(r_i))$ if $x_{i,p} = 1$, then add (r_i, h_j) to M where $h_j = \mathit{pref}(r_i, p)$. Further, for each $y_j \in Y$ set the score limit at hospital h_j in $\hat{t}$ to be $y_j \in Y$.

Again Theorem 8.5.1 proves that as a feasible solution to J, $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_j^-$, $\boldsymbol{\lambda}_j^+$ $\rangle$ satisfies all of the requirements of being a set of H-stable score limits in I and the assignment induced in I from them. It remains to consider whether the constraints intended to meet the extended capacity requirements in the HR LQ SLT context are satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_j^-$, $\boldsymbol{\lambda}_j^+$ $\rangle$.

We now demonstrate that as $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_j^-$, $\boldsymbol{\lambda}_j^+$ $\rangle$ satisfies Constraints 8.33 and 8.34 then for each $h_j \in H$, if h_j is closed in M and $c_j^- \geq 1$, then $\beta_j = 0$. Otherwise $\beta_j = 1$. Assume not. Assume $\beta_j = 0$ and further assume that h_j has one or more assignees in M. However, we now have that $\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : \mathit{pref}(r_i, p) = h_j\} \geq 1$ and $\beta_j c_j^+ = 0$ and Constraint 8.33 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$, a contradiction. Now assume $\beta_j = 1$ and further assume h_j has zero assignees in M where $c_j^- \geq 1$. However, we not have that $\sum_{i=1}^{n_1} \sum_{p=1}^{l(r_i)} \{x_{i,p} \in X : \mathit{pref}(r_i, p) = h_j\} = 0$ and $\beta_j c_j^- = c_j^- > 1$ and Constraint 8.34 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$, a contradiction. Hence, for every $h_j \in H$ if h_j is open, then $\beta_j = 1$. Otherwise $\beta_j = 0$.

As $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_j^-$, $\boldsymbol{\lambda}_j^+$ $\rangle$ satisfies Constraint 8.33 each resident in M must have at most c_j^+ partners. Moreover, since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_j^-$, $\boldsymbol{\lambda}_j^+$ $\rangle$ satisfies Constraint 8.34 any open hospital $h_j \in H$ must have at least c_j^- partners.

Now assume that some r_i is not assigned in M to an open hospital h_j at which they achieved the score limit and moreover r_i is not assigned to a better hospital than h_j. Hence, $(\bar{s}_j + 1) \sum_{p'=1}^{p} x_{i,p'} = 0$ and $s_{i,j} \geq y_j$, and thus Constraint 8.35 is not satisfied in $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$, a contradiction.

Since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$ $\rangle$ satisfies Constraint 8.36 it follows that if $\omega_{i,p} = 0$, then $s_{i,j} < y_j$ and thus r_i does not meet the score limit at h_j. Further, since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$ $\rangle$ satisfies Constraint 8.38 then $\phi_{i,p} = 0$ if $\omega_{i,p} = 0$.

Clearly, if $1 - \sum_{p'=1}^{p-1} x_{i,p'} = 1$, then r_i is not assigned to a better partner than h_j. Now, since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$ satisfies Constraint 8.39 it follows that if r_i is not assigned to a better partner than h_j, then $\phi_{i,p} = 0$. Further, since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$ $\rangle$ satisfies Constraint 8.37 we have that $\phi_{i,p} = 1$ if and only if r_i is not assigned to a better partner than h_j *and* r_i meets the score limit at h_j.

Since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$ satisfies Constraint 8.40 it follows that if a given hospital h_j has less than c_j^- acceptable partners who achieve the score limit at h_j and prefer h_j to their current partner, then $\lambda_j^- = 1$ and $\lambda_j^- \in \{0, 1\}$ otherwise. Similarly since $\langle$ **x**, **y**, $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$, $\boldsymbol{\omega}$, $\boldsymbol{\lambda}^-$, $\boldsymbol{\lambda}^+$ $\rangle$ satisfies Constraint 8.41 it follows that if a given hospital h_j has greater than c_j^+ acceptable partners who achieve the score limit at h_j and prefer h_j to their current partner, then $\lambda_j^+ = 1$ and $\lambda_j^+ \in \{0, 1\}$ otherwise.

Now since $\langle\, \mathbf{x}, \mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\omega}, \boldsymbol{\lambda}^-, \boldsymbol{\lambda}^+ \,\rangle$ satisfies Constraint 8.42 it follows that for each closed hospital $h_j \in H$ there are either fewer than c_j^- residents who meet the score limit at h_j and prefer h_j to their assigned partner or are unassigned *or* there are more than c_j^+ residents who meet the score limit at h_j and prefer h_j to their assigned partner or are unassigned. Thus M admits no blocking coalition in I. Hence we have that a feasible solution to J is equivalent to a set of H-stable score limits $\hat{t}$ in I and the result is proven. □

The existence of the objective function (given by Expression 8.43) immediately leads to the following corollary.

Corollary 8.6.2. *Given an instance I of* HR LQ SLT*, let J be the corresponding IP model as defined in Sections 8.6.1 and 8.6.2. A set of H-stable score limits $\hat{t}$ in I such that the assignment induced in I from $\hat{t}$ assigns the maximum number of residents possible in a set of H-stable score limits in I is exactly equivalent to an optimal solution in J.*

Chapter 9

Complexity results and integer programming models for TAP variants

9.1 Introduction

In this chapter we describe and prove the correctness of IP models for the NP-complete problems, TAP and STABLE TAP. We show empirical data from the application of the IP model for TAP to the process of allocating Trainee teachers studying at P.J. Šafárik University in Košice, Slovakia for the Spring 2013/14 allocation. We demonstrate how the IP model for the TAP problem may be adapted to the STABLE TAP context, where the applicants and schools have a preference ordering over their acceptable partners. We also show how complexity results from the TAP context allow us to prove the NP-completeness of the problem of deciding whether an arbitrary graph involving paired vertices admits a complete matching.

In Section 9.2 we present and prove the correctness of an IP model for TAP and in Section 9.3 we show data obtained when this TAP IP model was applied to the data from the allocation process for trainee teachers at P.J. Šafárik University for the Spring 2013/14 allocation. In Section 9.4 we apply complexity results from the TAP context to allow us to establish the NP-completeness of the problem of deciding whether a graph involving paired vertices admits a complete matching.

In Section 9.5 we show how the model presented in Section 9.2 may be adapted to the STABLE TAP context where the applicants and schools have a preference ordering over their acceptable partners;. In Section 9.6 we show how STABLE TAP and HRC are related through a polynomial-time reduction from STABLE TAP to HRC.

9.2 An integer programming formulation for TAP

We now present an IP model for the Teachers Allocation Problem (TAP) as defined in Section 2.8. Recall that an instance I of TAP comprises a set $A = \{a_1, a_2 \dots a_{n_1}\}$ containing applicants, a set $S = \{s_1, s_2 \dots s_{n_2}\}$ containing schools and a set $D = \{d_1, d_2 \dots d_{n_3}\}$ containing subjects (where, for example, d_1 might be maths, d_2 chemistry, etc.). Further, each applicant $a_i \in A$ has a vector v_i of length n_3, such that $v_{i,r} = 1$ if a_i specialises in subject d_r $(1 \leq r \leq n_3)$ and $v_{i,r} = 0$ otherwise. For a given i $(1 \leq i \leq n_1)$, since each applicant specialises in exactly two subjects it follows that $v_{i,r} = 1$ for exactly two values of r.

Each applicant $a_i \in A$ has a list of length $l(a_i)$ consisting of individual schools $s_j \in S$; these schools are acceptable to a_i, all other schools being unacceptable. Each school $s_j \in S$ has a list of acceptable applicants $a_i \in A$ of length $l(s_j)$. The lists expressed in this fashion are reciprocal, thus each school's list contains only those applicants for whom that school is acceptable. Further, we define a variable $c_{j,r}$ for each j $(1 \leq j \leq n_2)$ and r $(1 \leq r \leq n_3)$ such that $c_{j,r}$ is the partial capacity of school s_j with respect to subject d_r.

9.2.1 Variables in the IP model for TAP

Let J be the following IP formulation of I. In J, for each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(a_i))$, define a variable $x_{i,p}$ such that

$$x_{i,p} = \begin{cases} 1 & \text{if } a_i \text{ is assigned to his } p^{th} \text{ choice school} \\ 0 & \text{otherwise} \end{cases}$$

For $p = l(a_i) + 1$ the intuitive meaning is that applicant a_i is unassigned. Thus we also have

$$x_{i,l(a_i)+1} = \begin{cases} 1 & \text{if } a_i \text{ is unassigned} \\ 0 & \text{otherwise} \end{cases}$$

Let $X = \{x_{i,p} : 1 \leq i \leq n_1 \wedge 1 \leq p \leq l(a_i) + 1\}$. Let $pref(a_i, p)$ denote the school at position p of applicant a_i's preference list where $1 \leq i \leq n_1$ and $1 \leq p \leq l(a_i)$.

9.2.2 Constraints in the IP model for TAP

The following constraint simply confirms that each variable $x_{i,p}$ must be binary valued for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(a_i) + 1)$:

$$x_{i,p} \in \{0, 1\} \tag{9.1}$$

As each applicant $a_i \in A$ is assigned to exactly one school or is unassigned, we introduce the following constraint for all i $(1 \leq i \leq n_1)$:

$$\sum_{p=1}^{l(a_i)+1} x_{i,p} = 1 \tag{9.2}$$

Since a school s_j may be assigned to at most $c_{j,r}$ applicants in subject r $(1 \leq r \leq n_3)$, it follows that $x_{i,p} = 1$ for at most $c_{j,r}$ applicants where $pref(a_i, p) = s_j$ and $v_{i,r} = 1$. We thus obtain the following constraint for all j $(1 \leq j \leq n_2)$ and r $(1 \leq r \leq n_3)$:

$$\sum_{i=1}^{n_1} \sum_{p=1}^{l(a_i)} \{x_{i,p} \in X : pref(a_i, p) = s_j \wedge v_{i,r} = 1\} \leq c_{j,r} \tag{9.3}$$

Objective Function A maximum cardinality matching M in I is a matching in which the maximum number of applicants are assigned taken over all of the matchings admitted by I. To maximise the size of the matching derived from the solution to J we apply the following objective function:

$$\max \sum_{i=1}^{n_1} \sum_{p=1}^{l(a_i)} x_{i,p} \tag{9.4}$$

9.2.3 Proof of correctness of the IP model for TAP

Theorem 9.2.1. *Given an instance I of TAP, let J be the corresponding IP model as defined in Sections 9.2.1 and 9.2.2. A matching in I is exactly equivalent to a feasible solution to J.*

Proof. We first show that a matching in I represents a represents a feasible solution to J. Let M be a matching in I. From M we form an assignment of values to the variables $\mathbf{x}$ as follows. Initially $x_{i,p} = 0$ for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(a_i) + 1)$. Then for each $(a_i, s_j) \in M$ where $s_j = pref(a_i, p)$, $x_{i,p} = 1$. If a_i is unassigned then $x_{i,l(a_i)+1} = 1$. Clearly, since all $x \in X$ are constrained take a value of zero or one, Constraint 9.1 holds in the assignment derived from M. As each applicant is assigned to a single school or is unassigned (but not both), for a given i $(1 \leq i \leq n_1)$, for exactly one value of p in the range $1 \leq p \leq a(i) + 1$, $x_{i,p} = 1$, and for each other value of p in the same range, $x_{i,p} = 0$. Thus Constraint 9.2 holds in the assignment derived from M.

Since each school is assigned in M to at most $c_{j,r}$ acceptable applicants for all r $(1 \leq r \leq n_3)$, Constraint 9.3 also holds in the assignment derived from M. Now, since all of the constraints in J hold for an assignment derived from a matching M in I, M represents a feasible solution to J.

Conversely, consider a feasible solution $\langle \mathbf{x} \rangle$ to J. We form a set of pairs M from $\langle \mathbf{x} \rangle$ as follows. Initially let $M = \emptyset$. For each i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(a_i))$, if $x_{i,p} = 1$ then add (a_i, s_j) to M where $s_j = \mathit{pref}(a_i, p)$. Now, since $\langle \mathbf{x} \rangle$ satisfies Constraint 9.2 each applicant in M must be assigned to exactly one school or be unassigned (but not both). Further since $\langle \mathbf{x} \rangle$ satisfies Constraint 9.3 each s_j in M can have at most $c_{j,r}$ partners for all r $(1 \leq r \leq n_3)$. Thus the set of pairs M created from $\langle \mathbf{x} \rangle$ is a matching in I and the theorem is proven. □

The existence of the objective function (given by Inequality 9.4) immediately leads to the following corollary.

Corollary 9.2.2. *Given an instance I of* TAP, *let J be the corresponding IP model as defined in Sections 9.2.1 and 9.2.2. A maximum cardinality matching in I is exactly equivalent to an optimal solution to J.*

9.3 The IP model for TAP applied to real data

We applied our model to the real data from the allocation process for trainee teachers at P.J. Šafárik University for the Spring 2013/14 allocation. Approximately 500 students are training to be teachers at any given time. However, during each single allocation process only a subset of these trainee teachers will be involved in the allocation; the other trainee teachers will be involved in other aspects of their course. In Spring 2013/14 only 138 of the trainee teachers were to be allocated. There were 175 schools having approved supervising teachers during this year. We show the number of approved teachers for the subjects available during the Spring 2013/14 allocation and the number of students seeking each subject in Table 9.1.

Table 9.1 shows that during 2013/14 there were insufficient supervising teachers in Košice for Geography, History and Psychology. Although, there were sufficient numbers for Geography and History when the available places in the areas surrounding Košice are taken into consideration.

However, a very limited number of positions are available for trainee teachers who wish to study Psychology, wherever they wish to study. In practice, in the event that a teacher cannot be assigned to a school offering a Psychology post the trainee teacher should instead be assigned to a post teaching Ethics or Citizenship. We add the constraint which follows to enforce this further restriction in this allocation process.

Thus for each applicant who wishes to study a pair of subjects where one of those subjects is Psychology and the other is some $d_r \in D$ $(1 \leq r \leq n_3)$ that is not Psychology we create two new cloned applicants such that: the original applicant wishes to study Psychology and d_r; the first cloned applicant wishes to study Ethics and d_r; and the second cloned applicant

	Supervising Teachers in Košice	Total Supervising Teachers	Applicants
Maths	74	288	13
Physics	36	158	9
Biology	50	172	43
Chemistry	38	142	21
Informatics	44	137	4
Geography	31	127	35
Slovak	54	243	31
English	57	216	14
German	35	129	22
Latin	0	3	1
Civics	24	119	21
Psychology	2	12	22
Ethics	16	80	12
History	23	135	28

Table 9.1: Places available by subject for trainee teachers in Košice and surrounding areas

wishes to study Civics and d_r. Since these three applicants represent the same real trainee teacher, it cannot be the case that more than one of these applicants is assigned. Let g be the number of applicants wishing to study Psychology and without loss of generality the applicants constructed as above are applicants $1 \dots 3g$ and those applicants who do not wish to study Psychology are applicants $3g + 1 \dots n_1$.

Since only one of the three applicants representing the original applicant who wishes to study Psychology may be assigned, at least two of these applicants must be unassigned. We thus obtain the following constraint for all i $(0 \leq i \leq g - 1)$:

$$\sum_{z=1}^{3} x_{3i+z,l(a_{3i+z})+1} \geq 2 \tag{9.5}$$

With this additional constraint present in the model we applied the IP model to the allocation data for Spring 2013/14 and were able to allocate 122 of 138 trainee teachers to schools in Košice. When we also included schools in the areas surrounding Košice we were able to allocate 137 of the 138 trainees. The single trainee teacher who could not be allocated in the latter case wished to study Latin and only wished to study in Košice. However, since no supervising teacher was available at any school in Košice this trainee teacher could not be allocated to any school.

9.4 Complexity results derived from TAP

An instance I of the Bipartite Matching Problem with Couples (BMPC) is defined as follows. Let I be a bipartite graph $(R \cup H, E)$ where $R = \{r_1, r_2 \ldots r_{2n_1}\}$ and $H = \{h_1, h_2 \ldots h_{n_2}\}$.

Let the members of the set R (henceforth the *residents*) be in pairs of the form (r_{2i-1}, r_{2i}) $(1 \leq i \leq n_1)$ (henceforth the *couples*). Each couple (r_{2i-1}, r_{2i}) $(1 \leq i \leq n_1)$ has a list of acceptable hospital pairs of the form (h_{j_1}, h_{j_2}) where $h_{j_1} \in H, h_{j_2} \in H$ and possibly $h_{j_1} = h_{j_2}$. Any hospital pair not on this list is unacceptable to (r_{2i-1}, r_{2i}). Moreover, each hospital in I has capacity c_j, the maximum number of residents that it may be assigned.

If, in a matching M in I, a couple (r_{2i-1}, r_{2i}) $(1 \leq i \leq n_1)$ is assigned to an acceptable acceptable pair (h_{j_1}, h_{j_2}) where $h_{j_1} \in H, h_{j_2} \in H$ then $\{(r_{2i-1}, h_{j_1}), (r_{2i}, h_{j_2})\} \subseteq M$. A complete matching in I is a matching in which all couples are assigned to exactly one hospital pair.

Theorem 9.4.1. *Given an instance I of* BMPC*, the problem of deciding whether I admits a complete matching is NP-complete. The result holds even if each couple finds at most three hospital pairs acceptable and all of the hospitals have capacity of one.*

Proof. The proof of this result uses a polynomial-time reduction from FULL TAP as defined in Section 2.8. Cechlárová et al. [18] showed that FULL TAP is NP-complete even when each applicant finds at most three schools acceptable, there are fours subjects in total and each school has a partial capacity of one with respect to each subject.

Clearly BMPC is in NP as any set of pairs in I may be verified to be a complete matching in polynomial time. To show NP-hardness, let J be an instance of TAP with applicants $A = \{a_1, a_2, \ldots a_{n_1}\}$, schools $S = \{s_1, s_2, \ldots s_{n_2}\}$ and subjects $D = \{d_1, d_2, \ldots d_{n_3}\}$. Each school s_j has a vector $c_{j,k}$ $(1 \leq k \leq n_3)$, where $c_{j,k}$ represents the partial capacity of school s_j with respect to subject d_k. Further, each applicant in J find at most three schools acceptable, each school has a partial capacity of one with respect to each $d_k \in D$ and $|D|= 4$.

We form an instance I of BMPC from J as follows. For each school $s_j \in S$ and subject $d_k \in D$, if s_j has a partial capacity of $c_{j,k} = 1$ with respect to subject d_k, create a hospital $h_{j,k}$ in I with capacity $c_{j,k}$. For each applicant $a_i \in A$ create a resident couple (r_{2i-1}, r_{2i}) in I. Let $d_{k_1} \in D$ and $d_{k_2} \in D$ be the two subjects of choice for applicant $a_i \in A$. If school $s_j \in S$ is acceptable to applicant $a_i \in A$ then add the hospital pair (h_{j,k_1}, h_{j,k_2}) to the list of acceptable hospital pairs for the resident couple (r_{2i-1}, r_{2i}) in I. Thus, each couple finds acceptable at most three hospital pairs and each hospital has capacity one.

We claim that I admits a complete matching if and only if J admits a complete matching. Let M_J be a complete matching in J. Define a set of pairs M_I in I as follows. If $(a_i, s_j) \in M_J$, where $d_{k_1} \in D$ and $d_{k_2} \in D$ are the two subjects of choice for $a_i \in A$, then add the pairs (r_{2i-1}, h_{j,k_1}) and (r_{2i}, h_{j,k_2}) to M_I. Since each a_i in J is assigned to at most one acceptable school, each (r_{2i-1}, r_{2i}) in I is assigned in M_I to at most one acceptable hospital pair. Further, since no s_j is oversubscribed in M_J with respect to any subject $d_k \in D$, it follows that no $h_{j,k}$ in I may have greater than $c_{j,k}$ assignees. Hence M_I is a matching in I. Moreover, since every $a_i \in A$ is assigned in M_J, every couple in M_I must be assigned to exactly one acceptable hospital pair. Thus, M_I is a complete matching in I.

Conversely, suppose that M_I is a complete matching in I. Define a set of pairs M_J in J as follows. If (r_{2i-1}, r_{2i}) is assigned to the hospital pair (h_{j,k_1}, h_{j,k_2}), then add the pair (a_i, s_j) to M_I, where $d_{k_1} \in D$ and $d_{k_2} \in D$ are the two subjects of choice for $a_i \in A$. Since each (r_{2i-1}, r_{2i}) in I is assigned to at most one hospital pair, each a_i in J has at most one partner. Since no $h_{j,k}$ $(1 \leq j \leq n_2, 1 \leq k \leq n_3)$ is oversubscribed in M_I then no $s_j \in S$ is oversubscribed with respect to any subject $d_k \in D$. Hence M_J is a matching in J. Since each couple is assigned in M_I it must be the case that each applicant is assigned in M_J. Hence, M_J is a complete matching in J and the result is proven. □

9.5 An IP formulation for STABLE TAP

STABLE TAP is a variant of TAP in which the applicants express a preference order over their acceptable schools and the schools express preferences over those applicants who find them acceptable. In this section we show how to extend the IP model for TAP presented in Section 9.2 to the STABLE TAP context. In order to do so we first define some additional notation.

For an acceptable applicant-school pair (a_i, s_j), let $rank(s_j, a_i)$ denote the rank that school s_j assigns to applicant a_i where $1 \leq j \leq n_2$ and $1 \leq i \leq n_1$. Thus, $rank(s_j, a_i)$ is an integer in $[1, 2 \ldots, l(s_j)\}$ equal to the number of applicants that s_j prefers to a_i plus one.

For each j $(1 \leq j \leq n_2)$ and q $(1 \leq q \leq l(s_j))$, let the set $A(s_j, q)$ contain applicant-integer pairs (a_i, p) such that $rank(s_j, a_i) = q$ and $pref(a_i, p) = s_j$. Hence:

$$A(s_j, q) = \{(a_i, p) \in A \times \mathbb{Z} : rank(s_j, a_i) = q \wedge 1 \leq p \leq l(a_i) \wedge pref(a_i, p) = s_j\}.$$

9.5.1 Additional variables in the IP model for STABLE TAP

For each j $(1 \leq j \leq n_2)$, q $(1 \leq q \leq l(s_j))$ and r $(1 \leq r \leq n_3)$ where $rank(s_j, a_i) = q$ define a new variable $\alpha_{j,q,r} \in \{0, 1\}$ such that if s_j is full with respect to subject d_r with assignees better than rank q then $\alpha_{j,q,r}$ may take the value zero or one. However, if s_j is not full with assignees better than rank q with respect to subject d_r then $\alpha_{j,q,r} = 1$.

$$\alpha_{j,q,r} \geq 1 - \frac{\sum_{q'=1}^{q-1} \{x_{i',p} \in X : (a_{i'}, p) \in A(s_j, q') \wedge v_{i',r} = 1\}}{c_{j,r}} \tag{9.6}$$

Thus, if s_j is full with respect to subject d_r with assignees better than rank q then $\alpha_{j,q,r}$ may take the value zero or one. However, if s_j is not full with assignees better than rank q with respect to subject d_r then $\alpha_{j,q,r} = 1$.

9.5.2 Additional constraints in the IP model for STABLE TAP

For each j $(1 \leq j \leq n_2)$, q $(1 \leq q \leq l(s_j))$ and r $(1 \leq r \leq n_3)$ where $rank(s_j, a_i) = q$ define a new variable $\alpha_{j,q,r} \in \{0, 1\}$ such that:

$$\alpha_{j,q,r} \geq 1 - \frac{\sum_{q'=1}^{q-1} \{x_{i',p} \in X : (a_{i'}, p) \in A(s_j, q') \wedge v_{i',r} = 1\}}{c_{j,r}} \tag{9.7}$$

Thus, if s_j is full with respect to subject d_r with assignees better than rank q then $\alpha_{j,q,r}$ may take the value zero or one. However, if s_j is not full with assignees better than rank q with respect to subject d_r then $\alpha_{j,q,r} = 1$.

We apply the following constraint to ensure stability as defined in Definition 2.8.1 in Section 2.8.1. An applicant a_i may not be assigned to a worse school than s_j unless s_j is fully subscribed with partners it prefers to a_i in at least one subject in which a_i specialises. Let the two subjects of choice for a_i be d_{r_1} and d_{r_2}. Thus we obtain the following constraint for all i $(1 \leq i \leq n_1)$ and p $(1 \leq p \leq l(a_i))$, where $pref(a_i, p) = s_j$ and $rank(s_j, a_i) = q$:

$$\sum_{p'=p+1}^{l(a_i)+1} x_{i,p'} + \alpha_{j,q,r_1} + \alpha_{j,q,r_2} \leq 2 \tag{9.8}$$

Objective Function A maximum cardinality stable matching M in I is a stable matching in which the maximum number of teachers are assigned taken over all of the stable matchings admitted by I. To maximise the size of the stable matching derived from the solution to J we apply the following objective function:

$$\max \sum_{i=1}^{n_1} \sum_{p=1}^{l(a_i)} x_{i,p} \tag{9.9}$$

9.5.3 Proof of correctness of the IP model for STABLE TAP

Theorem 9.5.1. *Given an instance I of STABLE TAP, let J be the corresponding IP model as defined in Sections 9.5.1 and 9.5.2. A stable matching in I is exactly equivalent to a feasible solution to J.*

Proof. Let M be a stable matching in I. From M we form an assignment of values to the variables $\langle\ \mathbf{x}, \boldsymbol{\alpha}\ \rangle$ as follows. We assign values to the variables $\langle\ \mathbf{x}\ \rangle$ as shown in Theorem 9.2.1. Further, for each j $(1 \leq j \leq n_2)$, q $(1 \leq q \leq l(s_j))$ and r $(1 \leq r \leq n_3)$ if h_j is fully subscribed with assignees with rank better than q with respect to subject r then $\alpha_{j,q,r} = 0$. Otherwise, $\alpha_{j,q,r} = 1$.

By Theorem 9.2.1 the assignment of values to $\mathbf{x}$ derived from a matching in I satisfies Constraints 9.1, 9.2, 9.3 and 9.4. It remains to prove that $\langle\ \mathbf{x}, \boldsymbol{\alpha}\ \rangle$ satisfies Constraints 9.7 and 9.8.

Assume $\langle\ \mathbf{x}, \boldsymbol{\alpha}\ \rangle$ does not satisfy Constraint 9.7 for some j $(1 \leq j \leq n_2)$, q $(1 \leq q \leq l(s_j))$ and r $(1 \leq r \leq n_3)$. Since Constraint 9.7 is trivially satisfied if $\alpha_{j,q,r} = 1$, it follows that $\alpha_{j,q,r} = 0$. Now, from the construction h_j is fully subscribed with assignees with rank better than q with respect to subject r and $\sum_{q'=1}^{q-1}\{x_{i',p} \in X : (a_{i'}, p) \in A(s_j, q') \wedge v_{i',r} = 1\} = c_{j,r}$ and Constraint 9.7 is satisfied, a contradiction.

Assume $\langle\ \mathbf{x}, \boldsymbol{\alpha}\ \rangle$ does not satisfy Constraint 9.8 where the two subjects of choice for applicant a_i are d_{r_1} and d_{r_2}. Thus, $\sum_{p'=p+1}^{l(a_i)+1} x_{i,p'} = 1$, $\alpha_{j,q,r_1} = 1$ and $\alpha_{j,q,r_2} = 1$. Now, from the construction, a_i is assigned in M to a worse school than s_j and, since $\alpha_{j,q,r_1} = 1$ (respectively $\alpha_{j,q,r_2} = 1$) s_j is not fully subscribed with partners better than a_i in subject d_{r_1} (respectively d_{r_2}). Thus a_i blocks M with s_j, a contradiction.

Conversely, let J be an assignment of values to $\langle\ \mathbf{x}, \boldsymbol{\alpha}\ \rangle$ such that all constraints are satisfied in J. We form a set of pairs M in I as in Theorem 9.2.1. By Theorem 9.2.1, M is a matching in I. It remains to show that M is stable.

Assume a pair (a_i, s_j) blocks M where $rank(s_j, a_i) = q$ and $d_{r_1} \in D$ and $d_{r_2} \in D$ are the two subjects of choice for applicant a_i. It follows that a_i is assigned to a worse partner than s_j or is unassigned, and simultaneously s_j is not fully subscribed with better partners than a_i with respect to both subjects d_{r_1} and d_{r_2}. Hence, $\sum_{p'=p+1}^{l(a_i)+1} x_{i,p'} = 1$, $\alpha_{j,q,r_1} = 1$ and $\alpha_{j,q,r_2} = 1$ and Constraint 9.9 is not satisfied in J, a contradiction. □

Again, the existence of the objective function (given by Inequality 9.9) immediately leads to the following corollary.

Corollary 9.5.2. *Given an instance I of STABLE TAP, let J be the corresponding IP model as defined in Sections 9.5.1 and 9.5.2. A maximum cardinality stable matching in I is exactly equivalent to an optimal solution to J.*

9.6 A reduction from STABLE TAP to HRC

We now present a polynomial-time reduction from STABLE TAP to HRC. We can construct an instance of HRC from an instance of STABLE TAP in polynomial time. However, it is not immediately obvious how a reduction in the opposite direction might be constructed.

Lemma 9.6.1. *An instance I of STABLE TAP may be transformed in polynomial time to an instance J of HRC such that I admits a stable matching if and only if J does.*

Proof. Let I be an instance of STABLE TAP as defined in Section 2.8.1. We now show how to form an instance J of HRC from I as follows. For each school $s_j \in S$ and subject $d_k \in D$, if s_j has a partial capacity of $c_{j,k} \geq 1$ with respect to subject d_k, create a hospital $h_{j,k}$ in J with capacity $c_{j,k}$. For each applicant $a_i \in A$ create a resident couple (r_{2i-1}, r_{2i}) in J. Let $d_{k_1} \in P$ and $d_{k_2} \in D$ be the two subjects of choice for some applicant $a_i \in A$. Clearly $k_1 \neq k_2$ since no applicant expresses a preference for the same subject twice. For each school $s_j \in S$ acceptable to applicant $a_i \in A$ add the hospital pair (h_{j,k_1}, h_{j,k_2}) to the preference list of resident couple (r_{2i-1}, r_{2i}) in J.

Let M_I be a stable matching in I. Define a set of pairs M_J in J as follows. For each $(a_i, s_j) \in M_I$, where $d_{k_1} \in D$ and $d_{k_2} \in D$ are the two subjects of study for $a_i \in A$, let couple (r_{2i-1}, r_{2i}) be assigned to the hospital pair (h_{j,k_1}, h_{j,k_2}) in M_J. If a_i finds s_j acceptable, where $d_{k_1} \in D$ and $d_{k_2} \in D$ are the two subjects of study for $a_i \in A$, then (r_{2i-1}, r_{2i}) must find (h_{j,k_1}, h_{j,k_2}) acceptable. Since each a_i in I has at most one partner or is unassigned but not both, each (r_{2i-1}, r_{2i}) in J is assigned to at most one hospital pair or is unassigned. Further, since no s_j in I is over-subscribed with respect to any subject $d_k \in D$ then no $h_{j,k}$ in J may have more than $c_{j,k}$ assignees and thus cannot be over-subscribed. Hence M_J is a matching in J.

Assume that (r_{2i-1}, r_{2i}) blocks M_J with (h_{j,k_1}, h_{j,k_2}) in J. It must be the case that (r_{2i-1}, r_{2i}) is either unassigned or prefers (h_{j,k_1}, h_{j,k_2}) to $(M_J(r_{2i-1}), M_J(r_{2i}))$. Note that $h_{j,k_1} \neq M_J(r_{2i-1})$ and $h_{j,k_2} \neq M_J(r_{2i})$. It also follows that h_{j,k_1} (respectively h_{j,k_2}) is either undersubscribed or prefers r_{2i-1} (respectively r_{2i}) to some member of $M_J(h_{j,k_1})$ (respectively $M_J(h_{j,k_2})$). However, this implies that a_i either prefers s_j to its assigned partner in M_I or is unassigned and s_j is either undersubscribed in M_I or prefers a_i to one of its assigned partners in M_I with respect to both d_{k_1} and d_{k_2}. Thus (a_i, s_j) blocks M_I in I, a contradiction.

Conversely, suppose that M_J is a stable matching in J. Define a set of pairs M_I in I as follows. If (r_{2i-1}, r_{2i}) is assigned to the hospital pair (h_{j,k_1}, h_{j,k_2}) in M_J then add (a_i, s_j) to M_I; it follows that d_{k_1} and d_{k_2} must be the two subjects of study for a_i in J.

Since each each (r_{2i-1}, r_{2i}) in J is assigned to at most one hospital pair or is unassigned, each a_i in I has at most one partner or is unassigned. Further, since no hospital $h_{j,k} \in H$ is over-subscribed in M_J, no school is over-subscribed in I with respect to any subject $d_k \in D$.

Assume that (a_i, s_j) blocks M_I in I. Let d_{k_1} and d_{k_2} be the two subjects of choice for a_i. Thus, a_i either prefers s_j to his partner in M_I or is unassigned and for each $r \in \{1, 2\}$ either s_j is undersubscribed in d_{k_r} or prefers a_i to some $a_t \in M_I(s_j)$ where d_{k_r} is one of a_t's subjects.

However, this implies that in J, either (r_{2i-1}, r_{2i}) is unassigned in M_J or (r_{2i-1}, r_{2i}) prefers (h_{j,k_1}, h_{j,k_2}) to $(M_J(r_{2i-1}), M_J(r_{2i}))$ and h_{j,k_1} (respectively h_{j,k_2}) is either undersubscribed or prefers r_{2i-1} (respectively r_{2i}) to some member of $M_J(h_{j,k_1})$ (respectively $M_J(h_{j,k_2})$). It follows that (r_{2i-1}, r_{2i}) blocks M_J with (h_{j,k_1}, h_{j,k_2}) in J, a contradiction. Hence, the result is proven. □

It is not at all clear how to formulate a reduction from HRC to STABLE TAP. To see the difficulties associated with this, observe that in the above reduction from STABLE TAP to HRC, we modelled applicants' preference lists and their subjects in the constructed HRC instance by carefully choosing hospital pairs whose subscripts retained information about the applicants' subjects throughout. In a general HRC instance, the arbitrary nature of a couple's preferences implies that there is no straightforward relationship between hospital pairs and school preferences that retains information about the applicants' chosen subjects.

Chapter 10

Inapproximability results for a class of minimisation problems.

10.1 Introduction

In this chapter we present a framework for classifying certain minimisation problems in which the measure function may take only integer values π with NP-complete decision versions which will be used to prove that π is inapproximable to within a given bound, unless P=NP. These minimisation problems will be referred to as *decomposable* problems. We first consider the measure function with respect to which approximation is defined in stable matching problems. For minimisation problems such as the minimisation variants of stable matching problems the measure of an optimal solution may be zero – in this special case the performance guarantee of an approximation algorithm for π as described in Section 2.4 is not well defined. In Section 10.2 we present an adjusted measure function that naturally extends the previous measure function for minimisation problems and moreover leads to a well-defined notion of performance guarantee in the special case of minimisation problems having an optimal solution with a measure of zero.

For such minimisation problems by considering the performance guarantee to be defined relative to the adjusted measure function we are able to present a general proof in Section 10.3 that a class of minimisation problems having NP-complete decision versions must be inapproximable to within a given bound, unless P=NP. Further, we define the members of this class of problems to be decomposable problems.

In Section 10.4 we show how the general result in Section 10.3 implies that natural minimisation variants of the stable matching problems shown to be NP-complete elsewhere in this thesis are inapproximable to within a given bound, unless P=NP. Further, in Section 10.5 we show that this framework might be applied to show that minimisation variants of a number of NP-complete decision problems involving stable matchings are inapproximable to within a given bound, unless P=NP.

In Section 10.6 we discuss how this framework may be applied to prove that NP-complete decision problems involving k-colouring must have inapproximable minimisation variants. Further, we discuss how this framework may be applied to show that NP-complete decision problems involving SAT instances must also have minimisation variants that are inapproximable to within a given bound, unless P=NP.

10.2 An adjusted measure function for minimisation problems

Given an NPO problem π and an instance I of π, if $opt(I) = 0$, then the performance guarantee of an approximation algorithm for π with respect to I as described in Equation 2.2 in Section 2.4 (and shown again below) is not well-defined.

$$R_A(I) = \frac{m(I, A(I))}{opt(I)}. \tag{10.1}$$

To address this problem we define an adjusted measure function such that the properties of an approximation algorithm for π relative to the adjusted measure function are well-defined even in instances of the problem where the original measure function has optimal value zero.

Assume that π is a minimisation problem in which the measure function may take only integer values. Define a new adjusted measure function m' as follows: $m'(I, S) = \max\{m(I, S), 1\}$ for any instance I of π and for any $S \in SOL(I)$. Further, let $opt'(I) = \min\{m'(I, S) : S \in SOL(I)\}$. Given any constant $c \geq 1$, a *c-approximation algorithm relative to m'*, A is a polynomial time algorithm that outputs a feasible solution $A(I)$ for any instance I of π such that $m'(I, X(I)) \leq c \cdot opt'(I)$.

Proposition 10.2.1 demonstrates that the properties an approximation algorithm for some problem in NPO relative to the adjusted measure function are well defined even in instances of the problem where the optimal measure may take a value of zero. Further, the proposition demonstrates that the characteristics of such an approximation algorithm fit naturally with the expectations of an approximation algorithm.

Proposition 10.2.1. *Let π be a minimisation problem. If A is a c-approximation algorithm relative to m' for π where $c \geq 1$, then for any instance I of π, if $opt(I) = 0$, then $m(I, A(I)) \leq c$ and if $opt(I) > 0$, then $m(I, A(I)) \leq c \cdot opt(I)$.*

Proof. By definition, A returns a feasible solution S in I such that $m'(I,S) \leq c \cdot opt'(I)$. We now consider both cases in the proposition statement separately.

Case(i) – $opt(I) = 0$. For a contradiction suppose that $m(I,S) > c$. Then by the definition of $m'(I,S)$ it follows that $m'(I,S) = m(I,S)$. However, since $opt(I) = 0$ it follows that $opt'(I) = 1$. Hence, $m'(I,S) \leq c$ and thus $m(I,S) \leq c$, a contradiction. Thus if $opt(I) = 0$, then $m(I,S) \leq c$.

Case (ii) – $opt(I) > 0$. Since $opt(I) = \min\{m(I,S') : S' \in SOL(I)\}$ it follows that $m(I,S') > 0$ for all $S' \in SOL(I)$. Thus $m'(I,S') = m(I,S')$ for all $S' \in SOL(I)$ and it follows that $opt(I) = opt'(I)$. Thus $m(I,S) = m'(I,S) \leq c \cdot opt'(I) = c \cdot opt(I)$ as required. □

10.3 An inapproximability bound for a class of minimisation problems.

In Theorem 3.3.2 we proved that $(2,2)$-MIN BP HRC is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. In the proof of Theorem 3.3.2 we added a small unsolvable instance of HRC to the instance constructed as part of the proof to ensure that the measure of an optimal solution in the constructed instance could never take a value of zero. We now demonstrate that the form of this proof is quite general and may be applied to other minimisation problems. However, a generalisation of the proof in its current form seems unlikely since to prove the result for other minimisation problems, each problem would require a separate small unsolvable instance of the decision problem involved to be described in each proof.

By defining approximation relative to the adjusted measure function described in Section 10.2 we may state the result in a much more general form. Hence, in Theorem 10.3.1 we present a general proof that a group of NP-complete decision problems have a minimisation variant that is inapproximable to within a given bound, unless P=NP, where approximation is defined relative to the adjusted measure function described in Section 10.2. We now define the class $\mathcal{P}$ of problems, called *decomposable* minimisation problems, that we will be able to apply this result to.

Definition 10.3.1. *Let $\mathcal{P}$ be a family of tuples (π, f, g, h, g', h') where:*

- *π is a minimisation problem;*
- *f is a polynomially computable function that maps an instance I of π to a positive real number $f(I) \leq |I|$;*

- *g is a polynomially computable function that maps a positive integer and an instance of π to an instance of π;*
- *h is a polynomially computable function that maps a pair of instances of π to an instance of π;*
- *g' is a polynomially computable function that maps a positive integer, an instance of π and a feasible solution of an instance of π to a feasible solution of an instance of π;*
- *h' is a polynomially computable function that maps a pair of feasible solutions to an instance of π to a feasible solution of an instance of π.*

Let I be an instance of π. Define $J_2 = h(g(1, I), g(2, I))$, and for any $i > 2$ define $J_i = h(J_{i-1}, g(i, I))$. Now let S be a feasible solution of I. Define $S_2 = h'(g'(1, I, S), g'(2, I, S))$. For any $i > 2$ define $S_i = h'(S_{i-1}, g'(i, I, S))$. For $k \geq 2$, let $I' = J_k$ and let $S' = S_k$. The tuple (π, f, g, h, g', h') must additionally satisfy the following properties:

1. *$f(I') = kf(I)$;*
2. *If $S \in SOL(I)$, then $S' = S_k \in SOL(I')$. Moreover $m(I', S') = km(I, S)$ must hold.*

   Conversely if $S' \in SOL(I')$, then $S' = H(T_1, \ldots, T_k)$ where $T_i \in SOL(I)$ $(1 \leq i \leq k)$, and $H(T_1, \ldots, T_k) = H_k(T_1, \ldots, T_k)$, where for $3 \leq i \leq k$,

   $$H_i(T_1, \ldots, T_i) = h'(H_{i-1}(T_1, \ldots, T_{i-1}), g'(i, I, T_i))$$

   and $H_2(T_1, T_2) = h'(g'(1, I, T_1), g'(1, I, T_2))$. Moreover $m(I', S') = \sum_{i=1}^{k} m(I, T_i)$ must hold.
3. *The following decision problem is NP-complete:*
   Instance: *Any instance I of π*
   Question: *Is $opt(I) = 0$? That is, is there a feasible solution $S \in SOL(I)$ such that $m(I, S) = 0$?*

An optimisation problem π is said to be decomposable *if there exist functions f, g, h, g' and h' such that $(\pi, f, g, h, g', h') \in \mathcal{P}$.*

Intuitively, given an instance I of π, $f(I)$ is some measure of a constituent component of I that is no greater than the size of I; in our inapproximability results, hardness of approximation will be will be established relative to $f(I)$. For example, in an instance I of HRC, $f(I)$ will denote the number of residents in I.

Intuitively, $g(i, I)$ is an instance of π formed by adding a subscript i to the entities involved in I to differentiate between discrete copies of I. This implies that if $i \neq j$, then $g(i, I)$ and $g(j, I)$ are two disjoint copies of I with no entities in common. For a feasible solution S in I, $S_i = g'(i, S, I)$ is the feasible solution in $g(i, I)$ constructed by adding the subscript i to the entities involved in S. For example if I is an instance of HRC, $g(i, I)$ is the instance obtained by adding a subscript i to every agent and to every entry in each agent's preference list in I. Further, if S is a matching in I, then $g'(i, S, I)$ is the matching in $g(i, I)$ obtained by adding the subscript i to every agent in S.

Intuitively, now let i and j be two integers and let $I_i = g(i, I)$ and $I_j = g(j, I)$. Define $h(I_i, I_j)$ as the instance of π obtained by combining the two instances of π in some fashion, Further, $h'(S_i, S_j)$ is the feasible solution of $h(I_i, I_j)$ obtained by combining two feasible solutions S_i in I_i and S_j in I_j in the same manner. For example if I is an instance of HRC, $h(I_i, I_j)$ is the instance of HRC obtained by taking the union of the two disjoint subinstances I_i and I_j. Further, if S is a matching in I, then $h'(S_i, S_j)$ is the matching in $h(I_i, I_j)$ obtained by taking the union of the two disjoint solutions S_i in $g(i, I)$ and S_j in $g(j, I)$.

Having defined the necessary notation we may now state the following inapproximability result for decomposable minimisation problems. Intuitively Theorem 10.3.1 below implies that if π is a decomposable minimisation problem where $(\pi, f, g, h, g', h') \in \mathcal{P}$, then relative to the adjusted measure function, π is not approximable within $n^{1-\varepsilon}$, for any $\varepsilon > 0$, where $n = f(I)$, unless P=NP.

Theorem 10.3.1. *Let π be a decomposable minimisation problem where $(\pi, f, g, h, g', h') \in \mathcal{P}$. For an arbitrary instance I' of π, opt$'(I')$ is not approximable within $n^{1-\varepsilon}$, for any $\varepsilon > 0$, where $n = f(I')$, unless P=NP.*

Proof. Let $P = (\pi, f, g, h, g', h')$ be a member of $\mathcal{P}$. Now, let I_0 be an instance of π and further let $n_0 = f(I_0)$. Assume $\varepsilon > 0$. Choose $c = \lceil 2/\varepsilon \rceil$ and $k = n_0^c$. Now, let $g(1, I_0), g(2, I_0), \ldots, g(k+1, I_0)$ be $k+1$ disjoint instances derived from I_0 and define $I_j = g(j, I_0)$ $(1 \leq j \leq k+1)$. Let $I' = J_{k+1}$ where $J_2 = h(g(1, I_0), g(2, I_0))$, and $J_{z+1} = h(J_z, g(z+1, I_0))$ for $z > 2$. Now, let $n = (k+1)n_0$. Then by Property 1 of Definition 10.3.1, $n = f(I')$.

Clearly if I_0 admits a feasible solution S_0 such that $m(I_0, S_0) = 0$, then each I_j $(1 \leq j \leq k+1)$ must admit a feasible solution $S_j = g'(j, I_0, S_0)$ with $m(I_j, S_j) = 0$. Hence, $S' = S_{k+1}$ is a feasible solution in I', where $S_2 = h'(g'(1, I_0, S_0), g'(2, I_0, S_0))$ and for any $z \geq 2$, $S_{z+1} = h'(S_z, g'(z+1, I_0, S_0))$. Moreover $m(I', S') = 0$ by Property 2 of Definition 10.3.1. Hence *opt*$'(I') = 1$.

Now suppose that I_0 admits no feasible solution S with $m(I_0, S) = 0$. Let S' be any feasible solution in I'. Then by Property 2 of Definition 10.3.1, $S' = H(T_1, \ldots, T_{k+1})$ where $T_j \in$

$SOL(I_j)$ $(1 \leq j \leq k+1)$. Thus $m(I_j, T_j) \geq 1$ for each j $(1 \leq j \leq k+1)$ and hence $m(I', S') \geq k+1$ by Property 2 of Definition 10.3.1. Hence $opt(I') = opt'(I') \geq k+1$. We now show that $n^{1-\varepsilon} \leq k$.

Firstly $n = (k+1)n_0 \leq 2kn_0 = 2n_0^{c+1}$. Hence

$$\frac{n}{2} \leq n_0^{c+1}$$

which implies

$$\left(\frac{n}{2}\right)^{1/(c+1)} \leq n_0$$

Since $k = n_0^c$ it follows that

$$\left(\frac{n}{2}\right)^{c/(c+1)} \leq k$$

and hence

$$2^{-c/(c+1)} n^{c/(c+1)} \leq k. \tag{10.2}$$

We know that $n = (k+1)n_0 \geq n_0^{c+1} \geq n_0^c$ and we lose no generality by assuming that $n_0 \geq 2$. Hence $n \geq 2^c$ and it follows that $n^{-1} \leq 2^{-c}$ and thus

$$n^{-1/(c+1)} \leq 2^{-c/(c+1)}.$$

Hence

$$n^{-1/(c+1)} n^{c/(c+1)} \leq 2^{-c/(c+1)} n^{c/(c+1)} \tag{10.3}$$

and it follows by Inequalities 10.2 and 10.3 that

$$n^{(c-1)/(c+1)} = n^{c/(c+1)} n^{-1/(c+1)} \leq 2^{-c/(c+1)} n^{c/(c+1)} \leq k \tag{10.4}$$

We now show that $n^{1-\varepsilon} \leq n^{(c-1)/(c+1)}$. Observe that $c \geq 2/\varepsilon$ and thus $c+1 \geq 2/\varepsilon$. Hence

$$\varepsilon \geq \frac{2}{c+1}$$

and thus

$$1 - \varepsilon \leq 1 - \frac{2}{c+1} = \frac{c+1-2}{c+1} = \frac{c-1}{c+1}$$

and hence by Inequality 10.4, $n^{1-\varepsilon} \leq k$.

Now, assume that A is a polynomial-time approximation algorithm for π, relative to the adjusted measure function m', with a performance guarantee of $n^{1-\varepsilon} \leq k$. Let I_0 be an instance of the decision version of π and form the instance I' of π as described above. If

I_0 admits a feasible solution S_0 with $m(I_0, S_0) = 0$, then $opt'(I') = 1$ and A must return a feasible solution S' in I' where $m'(I', S') \leq k$. Otherwise, $opt'(I') \geq k + 1$ and A must return a feasible solution S' in I' with $m'(I', S') \geq k + 1$. Thus algorithm A may be used to determine in polynomial time whether I_0 admits a feasible solution S with $m(I_0, S) = 0$, a contradiction to property 3 of Definition 10.3.1, unless P=NP. Hence, no such approximation algorithm can exist, unless P = NP. □

10.4 Inapproximability bounds for problems in this thesis

In this section we show how the proof of Theorem 10.3.1 may be used to show that NP-complete decision problems described elsewhere in this thesis must have a decomposable minimisation variant that is inapproximable to within a given bound, unless P=NP. For HRC and TAP variants we will use this result to show that the number of blocking pairs in a 'most stable' matching is not approximable to within $n_1^{1-\varepsilon}$, for any $\varepsilon > 0$, unless P=NP, where n_1 is the number of residents (or applicants) in a given instance. In Section 10.4.1 we discuss in more detail how Theorem 10.3.1 can be applied to HRC variants, whilst in Section 10.4.2 we consider how Theorem 10.3.1 can be applied to TAP variants.

10.4.1 HRC variants

We first consider MIN BP HRC as defined in Section 2.4. Let MIN BP HRC be the problem of finding a matching in an instance I of HRC that admits the fewest blocking pairs taken over all of the matchings in I. We describe MIN BP HRC as an optimisation problem following the notation in Definition 2.4.1 as follows:

MIN BP HRC

Instance: An instance I of HRC;
Feasible solutions: All the matchings admitted by I;
Measure: The number of blocking pairs admitted by a matching in I;
Goal: min;
Optimisation version: Minimise the number of blocking pairs taken over all of the matchings admitted by I;
Decision version: Is there a matching in I that admits no blocking pairs?

We now show that MIN BP HRC is decomposable by defining functions f, g, h, g' and h' such that $(\pi, f, g, h, g', h') \in \mathcal{P}$ where π represents an instance of MIN BP HRC. The generality of Theorem 10.3.1 implies that inapproximability bounds hold for the minimisation variants of all of the HRC decision problems that are shown to be NP-complete in Chapter 3. These results are stated formally as Theorems 10.4.1, 10.4.2, 10.4.3 and 10.4.4 below with the full proof being stated for Theorem 10.4.1 only.

Theorem 10.4.1. $(\infty, 1, \infty)$-MIN BP HRC *is not approximable within* $n_1^{1-\varepsilon}$, *for any* $\varepsilon > 0$, *unless P=NP, where* n_1 *is the number of residents in a given instance. The result holds even if each hospital has capacity one.*

Proof. We show that $(\infty, 1, \infty)$-MIN BP HRC is a decomposable minimisation problem. Let I_0 be an arbitrary instance of MIN BP HRC. Further, let $f(I_0)$ be the number of residents in I_0. Moreover, let $I_1 = g(1, I_0)$ where $g(i, I_0)$ is the instance of MIN BP HRC formed by adding a subscript i to every agent and to every entry in each agent's preference list in I_0. Let $h(I_i, I_j)$ be the instance formed by taking the union of the two instance I_i and I_j. Let S_0 be a matching in I_0 and more generally let $S_i = g'(i, S_0, I_0)$ be the matching in I_i obtained by adding a subscript i to every agent in the matching S_0 in I_0. Further, let $h'(S_i, S_j)$ be the matching formed by taking the union of the two matchings S_i in I_i and S_j in I_j.

Now for any $k \geq 1$ let $I' = \bigcup_{i=1}^{k+1} g(i, I_0)$, be the instance of MIN BP HRC formed by taking the disjoint union of $k + 1$ copies of I_0. Since $f(I_0)$ is the number of residents in I, clearly $f(I') = (k + 1)f(I_0)$ and thus Property 1 of 10.3.1 holds for all variants of HRC.

Let S_0 be a matching in I_0. Clearly, $S_1 = g(1, S_0, I_0)$ is a matching in I_1 and thus a feasible solution of I_1. We may construct a feasible solution S' of I' by letting $S' = \bigcup_{i=1}^{k+1} g'(i, S_0, I_0)$. Clearly S' is a feasible solution of I' and $m(I', S') = |bp(I', S')| = k|bp(I_0, S_0)| = km(I_0, S_0)$ where $bp(I_0, S_0)$ denotes the set of blocking pairs of the matching S_0 in I_0.

Conversely, let S' be a feasible solution of I'. Then S' must be a union of feasible solutions to each of the individual subinstances I_j of I' and hence Property 2 of Definition 10.3.1 holds. We have previously shown in Theorem 3.2.1 that deciding whether an instance of $(\infty, 1, \infty)$-HRC admits a stable matching is NP-complete and thus Property 3 of Definition 10.3.1 holds. Thus $(\infty, 1, \infty)$-MIN BP HRC is a member of $\mathcal{P}$ and the result is proven. □

Theorem 10.4.2. $(2, 2)$-MIN BP HRC *is not approximable within* $n_1^{1-\varepsilon}$, *where* n_1 *is the number of residents in a given instance, for any* $\varepsilon > 0$, *unless P=NP. The result holds even if there are no single residents and each hospital has capacity one.*

Theorem 10.4.3. $(2, 3)$-MIN BP HRC *is not approximable within* $n_1^{1-\varepsilon}$, *where* n_1 *is the number of residents in a given instance, for any* $\varepsilon > 0$, *unless P=NP. The result holds even if there are no single residents and each hospital has capacity one and the preference list of each couple and hospital is derived from a strictly ordered master list of pairs of hospitals and residents respectively.*

Theorem 10.4.4. $(2,3)$-MIN BP HRC DUAL MARKET *is not approximable within* $n_1^{1-\varepsilon}$*, where* n_1 *is the number of residents in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The result holds even if each hospital has capacity one and the preference list of each single resident, couple and hospital is derived from a strictly ordered master list of hospitals, pairs of hospitals and residents respectively.*

10.4.2 TAP variants

In this section we consider inapproximability results for variants of TAP. Let I be an instance of STABLE TAP as defined in Section 2.8. Cechlárová et al. [17] showed that deciding whether an instance of STABLE TAP admits a stable matching is NP-complete under a number of restrictions. We define MIN BP STABLE TAP as the problem of finding a matching in an instance of STABLE TAP that admits the fewest blocking pairs taken over all of the matchings admitted by I. We define MIN BP STABLE TAP as an optimisation problem following the notation in Definition 2.4.1 as follows:

MIN BP STABLE TAP

Instance: An instance I of STABLE TAP;
Feasible solutions: All the matchings admitted by I;
Measure: The number of blocking pairs admitted by a matching in I;
Goal: min;
Optimisation version: Minimise the number of blocking pairs taken over all of the matchings admitted by I.
Decision version: Is there a matching in I that admits no blocking pairs?

If I' is an instance of MIN BP STABLE TAP and the functions f, g, h, g', h' are as defined in the proof of Theorem 10.4.1, then it follows that MIN BP STABLE TAP is decomposable. It then follows from Theorem 10.3.1 that MIN BP STABLE TAP is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of applicants in a given instance, for any $\varepsilon > 0$, unless P=NP, under the same restrictions for which the NP-completeness of the decision problem variants were proven. These results are stated formally as Theorems 10.4.5 and 10.4.6 below.

Theorem 10.4.5. MIN BP STABLE TAP *is not approximable within* $n_1^{1-\varepsilon}$*, where* n_1 *is the number of applicants in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The result holds even if there are at most three subjects, each partial capacity of a school is at most two and the preference list of each applicant is of length at most three.*

Theorem 10.4.6. MIN BP STABLE TAP *is not approximable within* $n_1^{1-\varepsilon}$*, where* n_1 *is the number of applicants in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The result holds even if there are at most three subjects, each partial capacity of a school is at most one, the preference lists of the schools are derived from subject specific master lists of applicants and the preference lists of the applicants are derived from a master list of schools.*

Now we consider the Bipartite Matching Problem with Couples (BMPC) as defined in Section 9.4. Recall that an instance I of BPMC is a bipartite graph $(R \cup H, E)$ where $R = \{r_1, r_2 \dots r_{2n_1}\}$, $H = \{h_1, h_2 \dots h_{n_2}\}$ and the residents in R are in pairs of the form (r_{2i-1}, r_{2i}) $(1 \leq i \leq n_1)$. Theorem 9.4.1 shows that deciding whether an instance of BPMC admits a complete matching is NP-complete. We define MIN UNASSIGNED COUPLES BPMC as an optimisation problem following the notation in Definition 2.4.1 as follows:

MIN UNASSIGNED COUPLES BPMC

Instance: An instance I of BPMC;
Feasible solutions: All the matchings admitted by I;
Measure: The number of unassigned couples in a matching in I;
Goal: min;
Optimisation version: Minimise the number of unassigned couples taken over all of the matchings admitted by I.
Decision version: Is there a matching in I in which no couples are unassigned?

MIN UNASSIGNED COUPLES BPMC is decomposable as can be seen by defining the functions f, g, h, g', h' are as defined in the proof of Theorem 10.4.1. Since Theorem 9.4.1 shows that deciding whether an instance of BPMC admits a complete matching is NP-complete it follows from Theorem 10.3.1 that MIN UNASSIGNED COUPLES BPMC is inapproximable to within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. This result is formally stated as Theorem 10.4.7 below; the structure of the proof follows that applied in Theorem 10.4.1.

Theorem 10.4.7. MIN UNASSIGNED COUPLES BPMC *is not approximable within* $n_1^{1-\varepsilon}$*, where* n_1 *is the number of residents in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The result holds even if all the hospitals have capacity one.*

10.5 Inapproximability bounds for NP-complete stable matching problems

10.5.1 Introduction

In this section we show that Theorem 10.4.1 implies that the minimisation variant of a number of NP-complete decision problems involving stable matchings are inapproximable to within a given bound, unless P=NP.

In Section 10.5.2 we show the minimisation variant of HR LQ, as defined in Section 2.6.1, is inapproximable to within a given bound, unless P=NP and in Section 10.5.3 we show that the minimisation variant of HR CQ, as defined in Section 2.6.2, is inapproximable to within a given bound, unless P=NP.

In Section 10.5.4 we show that the NP-completeness of a decision problem involving complete stable matchings in HRT instances may be used to show that two minimisation variants of the decision problem are necessarily inapproximable to within a given bound, unless P=NP.

10.5.2 HR LQ

Now we consider the HR LQ problem as defined in Section 2.6.1. Biró et al. [11] showed that the problem of deciding whether an instance of HR LQ admits a stable matching is NP-complete in the case that each hospital has upper and lower quota equal to three. We define MIN BP HR LQ as the problem of finding a matching in an instance of HR LQ that admits the minimum number of blocking pairs taken over all of the matchings admitted by I. MIN BP HR LQ may be defined as an optimisation problem following the notation in Definition 2.4.1 as follows:

MIN BP HR LQ

Instance: An instance I of HR LQ;
Feasible solutions: All the matchings admitted by I;
Measure: The number of blocking pairs admitted by a matching in I.
Goal: min;
Optimisation version: Minimise the number of blocking pairs in the matchings admitted by I;
Decision version: Is there a stable matching in I?

MIN BP HR LQ is decomposable which may be established by defining the functions f, g, h, g', h' as in the proof of Theorem 10.4.1. It follows from Theorem 10.3.1 that MIN BP HR LQ is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP, under the same restrictions for which the NP-completeness of the decision problem variants were proven in [11]. Thus, Theorem 10.5.1 below follows; the proof of the result follows the same structure as the proof of Theorem 10.4.1.

Theorem 10.5.1. MIN BP HR LQ *is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. The result holds even if each hospital has upper and lower quota equal to three.*

10.5.3 HR CQ

We now consider the HR CQ problem as defined in Section 2.6.2. It is known that the problem of deciding whether an instance of HR CQ admits a stable matching is NP-complete [11] even if the following three properties hold simultaneously: (i) each hospital and each bounded set has upper quota one; (ii) each bounded set contains two hospitals and (iii) each hospital appears in at most two bounded sets. We define MIN BP HR CQ as an optimisation problem following the notation in Definition 2.4.1 as follows:

MIN BP HR CQ

Instance: An instance I of HR CQ;
Feasible solutions: All the matchings admitted by I;
Measure: The number of blocking pairs admitted by a matching in I;
Goal: min;
Optimisation version: Minimise the number of blocking pairs over the matchings admitted by I;
Decision version: Is there a stable matching in I?

MIN BP HR CQ is decomposable which may be seen by applying the functions f, g, h, g', h' as defined in the proof of Theorem 10.4.1. Thus, it follows from Theorem 10.3.1 that MIN BP HR CQ is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP, under the same restrictions for which the NP-completeness of the decision problem variants were proven in [11]. We state this result formally as Theorem 10.5.2 below.

Theorem 10.5.2. MIN BP HR CQ *is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. The result holds even if the* *following three properties hold simultaneously: (i) each hospital and each bounded set has upper quota one; (ii) each bounded set contains two hospitals and (iii) each hospital appears in at most two bounded sets.*

10.5.4 HRT

It is known that COM HRT, the problem of deciding whether an instance of HRT admits a complete stable matching, is NP-complete even if each agents' preference list is of length at most three, each resident's list is strictly ordered, each hospital's preference list is either strictly ordered or is a tie of length two and each hospital has capacity one [55, 42].

We now define two optimisation problems MIN UNASSIGNED HRT and MIN BP COM HRT following the notation in Definition 2.4.1 as follows:

MIN UNASSIGNED HRT

Instance: An instance I of HRT;
Feasible solutions: All the stable matchings admitted by I;
Measure: The number of residents who are left unassigned in a stable matching in I.
Goal: min;
Optimisation version: Minimise the number of residents who are left unassigned in a stable matching in I;
Decision version: Is there a stable matching in I in which no residents remain unassigned?

MIN BP COM HRT

Instance: An instance I of HRT;
Feasible solutions: All the complete matchings admitted by I;
Measure: The number of blocking pairs in a complete matching in I.
Goal: min;
Optimisation version: Minimise the number of blocking pairs in a complete matching in I;
Decision version: Is there a complete stable matching in I?

MIN UNASSIGNED HRT or MIN BP COM HRT are both decomposable as can be seen by applying the functions f, g, h, g', h' as defined in the proof of Theorem 10.4.1. It follows from Theorem 10.3.1 that MIN UNASSIGNED HRT and MIN BP COM HRT are not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. This result holds under the same restrictions for which the NP-completeness of the decision problem variants were proven in [55, 42]. These results are stated formally as Theorems 10.5.3 and 10.5.4.

Theorem 10.5.3. MIN UNASSIGNED HRT *is not approximable within* $n_1^{1-\varepsilon}$*, where* n_1 *is the number of residents in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The results holds even if each preference list is of length at most three, each resident's list is strictly ordered, each hospital's preference list is either strictly ordered or is a tie of length two and each hospital has capacity one.*

Theorem 10.5.4. MIN BP COM HRT *is not approximable within* $n_1^{1-\varepsilon}$*, where* n_1 *is the number of residents in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The results holds even if each preference list is of length at most three, each resident's list is strictly ordered, each hospital's preference list is either strictly ordered or is a tie of length two and each hospital has capacity one.*

Now we have two seemingly contradictory results, namely that MAX HRT is approximable to within $3/2$ [54, 45] yet MIN UNASSIGNED HRT is inapproximable within $n_1^{1-\varepsilon}$, where n_1 is the number of residents in a given instance, for any $\varepsilon > 0$, unless P=NP. Yet the two problems are equivalent up to polynomial-time reductions. However, this apparent contradiction has been shown to exist in a number of other problem contexts. For example, consider two related problems; the problem of finding a minimum vertex cover and the problem of finding a maximum independent set in an arbitrary graph. Given a graph $G = (V, E)$, there is a straightforward correspondence between a minimum vertex cover in G and a maximum independent set in G – namely a set of vertices is only a maximum independent set if its complement is a minimum vertex cover. However the approximation properties of the two problems are not the same. The problem of finding a minimum vertex cover is approximable within $2 - \dfrac{\log\log|V|}{2\log|V|}$ [57, 7], yet the problem of computing a maximum independent set is not approximable to within $|V|^{1-\varepsilon}$, for any $\varepsilon > 0$, unless P=NP [85].

10.6 An inapproximability bound for k-COLOURING & SAT problems

In this section we show that the proof of Theorem 10.3.1 may be generalised to prove that the minimisation variants of two well known NP-complete problems are hard to approximate. Zuckerman [84] has shown previously using a different methodology that NP-complete problems have a version that is hard to approximate. First, we show that the proof of Theorem 10.3.1 may be used to establish that a well known NP-complete combinatorial problem, the k-COLOURING PROBLEM [29], has a natural minimisation variant that is inapproximable to within a given bound, unless P=NP.

The k-COLOURING PROBLEM is defined as follows. Given a graph $G = (V, E)$, a *colouring* in G is an assignment of colours to the vertices in G. In an arbitrary colouring adjacent vertices may have the same colour. A *proper colouring* in G is a colouring in which no two adjacent vertices share the same colour. A *k-colouring* is a colouring containing at most k colours, a *proper k-colouring* is a proper colouring containing at most k colours. For each fixed $k \geq 3$, it is NP-complete to decide if an arbitrary graph G admits a proper k-colouring. Further, the minimum value of k for which G admits a proper k-colouring is inapproximable to within $n^{1-\varepsilon}$, where n is the number of vertices in the graph, for any $\varepsilon > 0$, unless P=NP [85]. We define the optimisation problem MIN IMPROPERLY COLOURED EDGES following the notation in Definition 2.4.1 as follows:

MIN IMPROPERLY COLOURED EDGES

Instance: A graph $G = (V, E)$ and an integer k;
Feasible solutions: All colourings (not necessarily proper) of the vertices in V using at most k colours;
Measure: The number of edges where the two endpoints share the same colour;
Goal: min;
Optimisation version: Minimise the number of edges where the two endpoints share the same colour taken over all of the k-colourings of G;
Decision version: Does G admit a proper k-colouring?

Thus, given a graph $G = (V, E)$ and an integer k, MIN IMPROPERLY COLOURED EDGES is the problem of finding a k-colouring that induces the minimum number of edges having two endpoints sharing the same colour, taken over all of the possible k-colourings in the graph. We define the functions f, g, h, g', h' in this context as follows.

Let I_0 be an arbitrary instance of the k-COLOURING PROBLEM comprising a graph $G_0 = (V_0, E_0)$ and an integer k. Define $f(I_0)$ to be $|V_0|$. Moreover, for any $i \geq 1$, let $I_i = g(i, I_0)$, where $g(i, I_0) = \langle G_i, k \rangle$, where G_i is the graph obtained from G_0 by by adding the subscript i to each vertex in V and to the endpoints of each edge in E_0. Given two integers r and s, let $h(I_r, I_s) = I_t$ where $I_t = \langle G_t, k \rangle$, $G_t = (V_t, E_t)$, $V_t = V_r \bigcup V_s$ and $E_t = E_r \bigcup E_s$. Let S_0 be a colouring in I_0 and for any $i \geq 1$ let $S_i = g'(i, S_0, I_0)$ be the colouring in I_0 obtained by adding the subscript i to every vertex in the colouring S_0 in I_0. Further, let $h'(S_i, S_j)$ be the colouring in $h(I_i, I_j)$ formed by taking the union of the colourings S_i in I_i and S_j in I_j.

If π represents an instance of MIN IMPROPERLY COLOURED EDGES and the functions f, g, h, g', h' are as defined above then $(\pi, f, g, h, g', h') \in \mathcal{P}$ and hence π is a decomposable minimisation problem. It follows from Theorem 10.3.1 that MIN IMPROPERLY COLOURED EDGES is not approximable within $n^{1-\varepsilon}$, where n is the number of vertices in a given instance, for any $\varepsilon > 0$, unless P=NP. This result holds under the same restrictions for which the NP-completeness of the decision problem variant is proven. Thus, we obtain the following result.

Theorem 10.6.1. MIN IMPROPERLY COLOURED EDGES *is not approximable within* $n^{1-\varepsilon}$*, where* n *is the number of vertices in a given instance, for any* $\varepsilon > 0$*, unless P=NP. The result holds for all fixed* $k \geq 3$*.*

We now consider how Theorem 10.3.1 might be applied to a variant of the well known Satisfiability problem (SAT), which is the problem of deciding, given a Boolean formula B, in conjuctive normal form, whether B is satisfiable i.e. whether there exists a truth assignment for B that makes every clause in B true. Cook [20] showed that SAT is NP-complete. The restriction of SAT in which each clause in B must contain at most k variables is denoted by k-SAT. 3-SAT was one amongst a list of 21 NP-complete problems discussed by Karp [43] in his paper on the interreducibility of NP-complete problems. It is known that the maximum number of clauses that may be satisfied by an assignment of truth values in an instance of SAT is not approximable to within $7/8 + \varepsilon$, for any $\varepsilon > 0$, unless P=NP [33]. We define the related problem MIN UNSATISFIED CLAUSES SAT following the notation in Definition 2.4.1 as follows:

MIN UNSATISFIED CLAUSES SAT

Instance: A Boolean formula B in CNF over a set of variables V;
Feasible solutions: All assignments of truth values to the variables in V;
Measure: The number of clauses in B that are unsatisfied by the assignment of values to the variables;
Goal: min;
Optimisation version: Minimise the number of clauses in B that are unsatisfied across all possible truth assignments for B;
Decision version: Is B satisfiable?;

Thus, given a Boolean formula B in CNF over a set of variables V, MIN UNSATISFIED CLAUSES SAT is the problem of finding an assignment of truth values to the variables in V such that the minimum number of clauses are unsatisfied, taken over all of the assignments of truth values to the variables in V. We define the functions f, g, h, g', h' in this context as follows. Given an instance B_0 of SAT, define $f(B_0)$ as the number of variables in B_0. Further, for any $i \geq 1$, let $B_i = g(i, B_0)$ denote the instance of SAT obtained by adding a subscript i to every variable in B_0. Let S_0 be a feasible solution in B_0. We denote by S_i the feasible solution in I_i obtained by adding a subscript i to every variable in the assignment of variables in S_0. Given any two integers i and j, let $h(B_i, B_j)$ denote the instance of SAT obtained by taking the conjunction of B_i and B_j. Further, let $h'(S_i, S_j)$ be the feasible solution of $h(B_i, B_j)$ obtained by taking the union of the truth assignments S_i in B_i and S_j in B_j.

If π represents MIN UNSATISFIED CLAUSES SAT and the functions $f, g, h, g', h')$ are as defined above, then $(\pi, f, g, h, g', h') \in \mathcal{P}$. Hence π is a decomposable minimisation function. It follows from Theorem 10.3.1 that MIN UNSATISFIED CLAUSES SAT is not approximable within $n_1^{1-\varepsilon}$, where n_1 is the number of variables in a given instance, for any $\varepsilon > 0$, unless P=NP. We thus obtain the following result.

Theorem 10.6.2. MIN UNSATISFIED CLAUSES SAT *is not approximable within* $n^{1-\varepsilon}$*, where* n *is the number of variables in a given instance, for any* $\varepsilon > 0$*, unless P=NP.*

Chapter 11

Conclusions and open problems

The new NP-completeness results presented in Chapter 3 suggest that an efficient algorithm for finding a stable matching in an instance of HRC, even when the length of the agents' preference lists are restricted, is very unlikely. However, it remains open to establish the exact frontier between polynomial time solvability and NP-completeness for further restricted variants of HRC where the length of the preference list of each couple is exactly one and the preference list of each hospital and resident is of bounded length. For example, we have shown that the problem of deciding whether a stable matching exists is NP-complete even for instances of $(\infty, 1, \infty)$-HRC and $(2, 2)$-HRC and we presented a polynomial-time algorithm for $(2, 1, 2)$-HRC. However, the complexity of $(2, 1, 3)$-HRC and $(3, 1, 2)$-HRC remains open. It is also possible that restrictions of HRC that do not necessarily involve restrictions on the length of the agents' preference lists might admit polynomial-time algorithms for finding stable matchings.

Since an efficient algorithm for problems related to HRC is unlikely, in this thesis we have considered whether IP techniques can be successfully applied to find optimal solutions in real world matching applications. We have presented a number of models for finding maximum cardinality stable matchings in a variety of centralised matching schemes and shown the practical results from the application of these models to real and randomly generated instances reflecting the properties of the instances arising in these schemes. The success of IP techniques in finding maximum cardinality matchings in such instances demonstrates that these techniques can offer a viable path to exact optimal solutions in real world allocation problems.

The empirical work in Chapters 6 and 7 demonstrates that the IP models for finding maximum cardinality stable matchings in HRC instances presented in Chapters 4 and 5 perform well when solving instances that are similar in structure and size to the instances arising in the SFAS application. It remains open to investigate the performance of the model as we increase the size of the instance substantially beyond the size of the SFAS application.

It would also be of interest to compare an IP model producing maximum cardinality BIS-stable matchings against the sizes of the BIS-stable matchings returned by HRC heuristics such as those compared and contrasted by Biró et al. [12]. We demonstrated that the IP model for HRCT can be applied to instances arising from the SFAS data from 2010-2012 and that guaranteed optimal solutions may be obtained in a practically useful timescale. It might be of further interest to investigate other modelling frameworks for problems of this type, for example involving CSP strategies – previous work has investigated the application of CSP techniques to HR [23, 52] and HRT [61] problems.

In Chapter 8 we described an IP model for finding a set of stable score limits in an instance of HR SLT with additional restrictions reflecting the process applied in the Hungarian Higher Education matching scheme. As we combine restrictions in the matching problems the resulting IP models become much more complex to represent and prove. However decisions on the restrictions in the application are obviously a matter for the programme co-ordinators in the allocation scheme. Further work in this area involves implementing the IP models described in Chapter 8 and applying them to real instances arising from the Hungarian Higher Education matching scheme. This scheme is on a far larger scale than any of the matching schemes to which the IP techniques have been applied in this thesis, it remains to be seen how well an implementation of these models would perform with such large instances and to establish how the performance varies with each additional restriction.

In Chapter 9 we demonstrated how IP techniques might be applied to the process of allocating Trainee teachers studying at P.J. Šafárik University in Košice, Slovakia for the Spring 2013/14 allocation. We found that the TAP model returned a guaranteed optimal outcome in a matter of seconds in the instances arising from this allocation. Cechlárová et al. [17, 18] have published a number of results in this problem context. However, it remains open to investigate other definitions of fairness in the TAP context - e.g. if the teachers are able to express a preference order over their acceptable schools and the schools remain indifferent over all of their acceptable partners, then we might consider an alternative definition of fairness such as Pareto optimality with respect to the teachers [3].

As the number of agents involved in a matching scheme increases, we may encounter a limit on the size of the problems that can be solved in a practically useful timescale by IP solvers. It may be possible to take advantage of some underlying structural properties of the problems to be solved and thus increase the size of the instances that can be solved in a practically useful timescale. In other problem contexts, techniques such as column generation [8] have been applied to achieve this end. It remains open to establish whether column generation techniques can be applied to increase the size of instances which may be solved by the models presented in this work.

The framework presented in Chapter 10 defines the class of decomposable minimisation problems and we showed that such problems are inapproximable to within a given bound, unless P=NP. The list of examples of decomposable minimisation problems is by no means exhaustive, particularly with respect to stable matching problems, suggesting in general terms that natural minimisation variants of a number of stable matching problems are inapproximable to within a given bound, unless P=NP. It remains open to consider whether this framework might be applicable in other problem contexts than those described in this work.

Bibliography

[1] A. Abdulkadiroğlu and T. Sönmez. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689–701, 1998.

[2] D.J. Abraham, P. Biró, and D.F. Manlove. "Almost stable" matchings in the Roommates problem. In *Proceedings of WAOA '05: Workshop on Approximation and Online Algorithms*, volume 3879 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2006.

[3] D.J. Abraham, K. Cechlárová, D.F. Manlove, and K. Mehlhorn. Pareto optimality in house allocation problems. In *Proceedings of ISAAC 2004: the 15th Annual International Symposium on Algorithms and Computation*, volume 3341 of *Lecture Notes in Computer Science*, pages 3–15. Springer, 2004.

[4] B. Aldershof and O.M. Carducci. Stable matching with couples. *Discrete Applied Mathematics*, 68:203–207, 1996.

[5] M. Baïou and M. Balinski. The stable admissions polytope. *Mathematical Programming, Series A*, 87:427–439, 2000.

[6] M. Balinski and T. Sönmez. A tale of two mechanisms: student placement. *Journal of Economic Theory*, 84(1):73–94, 1999.

[7] R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the weighted vertex cover problem. In *Analysis and Design of Algorithms for Combinatorial Problems*, volume 109 of *North-Holland Mathematics Studies*, pages 27 – 45. North-Holland, 1985.

[8] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W. P. Savelsbergh, and P.H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Oper. Res.*, 46(3):316–329, March 1998.

[9] P. Berman, M. Karpinski, and Alexander D. Scott. Approximation hardness of short symmetric instances of MAX-3SAT. ECCC Report, number 49, 2003.

[10] P. Biró. Student admissions in Hungary as Gale and Shapley envisaged. Technical Report TR-2008-291, University of Glasgow, Dept. of Computing Science, 2008.

[11] P. Biró, T. Fleiner, R.W. Irving, and D.F. Manlove. The College Admissions problem with lower and common quotas. *Theoretical Computer Science*, 411:3136–3153, 2010.

[12] P. Biró, R.W. Irving, and I. Schlotter. Stable matching with couples: an empirical study. *ACM Journal of Experimental Algorithmics*, 16, 2011. Section 1, article 2.

[13] P Biró and S Kiselgof. College admissions with stable score-limits. *Central European Journal of Operations Research*, pages 1–15, 2013.

[14] P. Biró and F. Klijn. Matching with couples: a multidisciplinary survey. *International Journal of Game Theory*, 15, 2013.

[15] P. Biró, D.F. Manlove, and I. McBride. The Hospitals / Residents problem with Couples: Complexity and Integer Programming Models. In *Proceedings of SEA2014: the 8th Symposium on Experimental Algorithms*, volume 8504 of *Lecture Notes in Computer Science*, pages 10–21. Springer International Publishing, 2014.

[16] P. Biró and I. McBride. Integer Programming Methods for Special College Admissions problems. In *Combinatorial Optimization and Applications*, volume 8881 of *Lecture Notes in Computer Science*, pages 429–443. Springer International Publishing, 2014.

[17] K. Cechlárová, T. Fleiner, D.F. Manlove, and I. McBride. Stable matchings of teachers to schools. *CoRR*, abs/1501.05547, 2015.

[18] K. Cechlárová, T. Fleiner, D.F. Manlove, I. McBride, and E. Potpinková. Modelling Practical Placement of Trainee Teachers to Schools. *Central European Journal of Operations Research*, 23(3):547–562, 2015.

[19] Y. Chen and T. Sönmez. Improving efficiency of on-campus housing: An experimental study. *American Economic Review*, 92(5):1669–1686, 2002.

[20] S.A. Cook. The complexity of theorem-proving procedures. In *Proceedings of STOC '71: the 3rd Annual ACM Symposium on the Theory of Computing*, pages 151–158. ACM, 1971.

[21] X. Deng, C. Papadimitriou, and S. Safra. On the complexity of equilibria. In *Proceedings of STOC '02: the Thirty-fourth Annual ACM Symposium on Theory of Computing*, pages 67–71, New York, NY, USA, 2002. ACM.

[22] J. Drummond, A. Perrault, and F. Bacchus. SAT is an effective and complete method for solving stable matching problems with couples. In *Proceedings of the Thirty-fourth Joint Conference on Artificial Intelligence*, IJCAI '15, 2015.

[23] P. Eirinakis, D. Magos, I. Mourtos, and P. Miliotis. Hyperarc consistency for the stable admissions problem. In *Proceedings of ICTAI 2007: the 19th IEEE International Conference on Tools with Artificial Intelligence*, volume 1, pages 239–242. IEEE Computer Society, 2007.

[24] P. Eirinakis, D. Magos, I. Mourtos, and P. Miliotis. Finding all stable pairs and solutions to the many-to-many stable matching problem. *INFORMS Journal on Computing*, 2(24):245–259, 2012.

[25] K. Eriksson and O. Häggström. Instability of matchings in decentralized markets with various preference structures. *International Journal of Game Theory*, 36(3-4):409–420, 2008.

[26] T. Fleiner. On the stable b-matching polytope. *Mathematical Social Sciences*, 46:149–158, 2003.

[27] D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.

[28] D. Gale and M. Sotomayor. Some remarks on the stable matching problem. *Discrete Applied Mathematics*, 11:223–232, 1985.

[29] M.R. Garey and D.S. Johnson. *Computers and Intractability*. Freeman, San Francisco, CA., 1979.

[30] N. Garg, T. Kavitha, A. Kumar, K. Mehlhorn, and J. Mestre. Assigning papers to referees. *Algorithmica*, 58(1):119–136, 2010.

[31] D. Gusfield and R.W. Irving. *The Stable Marriage Problem: Structure and Algorithms*. MIT Press, 1989.

[32] M.M. Halldórsson, R.W. Irving, K. Iwama, D.F. Manlove, S. Miyazaki, Y. Morita, and S. Scott. Approximability results for stable marriage problems with ties. *Theoretical Computer Science*, 306(1-3):431–447, 2003.

[33] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, July 2001.

[34] A. Hylland and R. Zeckhauser. The efficient allocation of individuals to positions. *Journal of Political Economy*, 87(2):293–314, 1979.

[35] R.W. Irving. An efficient algorithm for the "stable roommates" problem. *Journal of Algorithms*, 6:577–595, 1985.

[36] R.W. Irving. Matching medical students to pairs of hospitals: a new variation on a well-known theme. In *Proceedings of ESA '98: European Symposium on Algorithms*, volume 1461 of *Lecture Notes in Computer Science*, pages 381–392. Springer, 1998.

[37] R.W. Irving. Personal communication, 2011.

[38] R.W. Irving. Personal communication, 2014.

[39] R.W. Irving and P. Leather. The complexity of counting stable marriages. *SIAM Journal on Computing*, 15(3):655–667, 1986.

[40] R.W. Irving and D.F. Manlove. The Stable Roommates Problem with Ties. *Journal of Algorithms*, 43:85–105, 2002.

[41] R.W. Irving, D.F. Manlove, and G. O'Malley. Stable marriage with ties and bounded length preference lists. In *Proceedings of ACiD 2006: the 2nd Algorithms and Complexity in Durham workshop*, volume 7 of *Texts in Algorithmics*, pages 95–106. College Publications, 2006.

[42] R.W. Irving, D.F. Manlove, and G. O'Malley. Stable marriage with ties and bounded length preference lists. *Journal of Discrete Algorithms*, 7(2):213–219, 2009.

[43] R.M. Karp. Reducibility among combinatorial problems. In R.E. Miller and J.W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[44] L.G. Khachiyan. Polynomial algorithms in linear programming. {*USSR*} *Computational Mathematics and Mathematical Physics*, 20(1):53 – 72, 1980.

[45] Z. Király. Linear time local approximation algorithms for the stable marriage problem. In *Proceedings of MATCH-UP '12: the 2nd International Workshop on Matching Under Preferences*, pages 99–110.

[46] D.E. Knuth. *Mariages Stables*. Les Presses de L'Université de Montréal, 1976. English translation in *Stable Marriage and its Relation to Other Combinatorial Problems*, volume 10 of CRM Proceedings and Lecture Notes, American Mathematical Society, 1997.

[47] A. Kwanashie and D.F. Manlove. An integer programming approach to the hospitals / residents problem with ties. Technical Report arXiv:1308:4064, Computing Research Repository, Cornell University Library, 2013.

[48] D.F. Manlove. *Algorithmics of Matching Under Preferences*. World Scientific, 2013.

[49] D.F. Manlove. Personal communication, 2015.

[50] D.F. Manlove, R.W. Irving, K. Iwama, S. Miyazaki, and Y. Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261–279, 2002.

[51] D.F. Manlove and G. O'Malley. Paired and altruistic kidney donation in the uk: Algorithms and experimentation. In Ralf Klasing, editor, *Experimental Algorithms*, volume 7276 of *Lecture Notes in Computer Science*, pages 271–282. Springer, 2012.

[52] D.F. Manlove, G. O'Malley, P. Prosser, and C. Unsworth. A Constraint Programming Approach to the Hospitals / Residents Problem. In *Proceedings of CP-AI-OR 2007: the 4th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization*, volume 4510 of *Lecture Notes in Computer Science*, pages 155–170. Springer, 2007.

[53] I. McBride and D.F. Manlove. An Integer Programming Model for the Hospitals / Residents problem with Couples. In *Operations Research Proceedings 2013*, pages 293–299. Springer, 2014.

[54] E. McDermid. A 3/2 approximation algorithm for general stable marriage. In *Proceedings of ICALP 2009: the 36th International Colloquium on Automata, Languages and Programming*, volume 5555 of *Lecture Notes in Computer Science*, pages 689–700. Springer, 2009.

[55] E.J. McDermid and D.F. Manlove. Keeping partners together: Algorithmic results for the hospitals / residents problem with couples. *Journal of Combinatorial Optimization*, 19(3):279–303, 2010.

[56] D. McVitie and L.B. Wilson. The stable marriage problem. *Communications of the ACM*, 14:486–490, 1971.

[57] B. Monien and E. Speckenmeyer. Ramsey numbers and an approximation algorithm for the vertex cover problem. *Acta Informatica*, 22(1):115–123, 1985.

[58] C. Ng and D.S. Hirschberg. Lower bounds for the stable marriage problem and its variants. *SIAM Journal on Computing*, 19:71–77, 1990.

[59] C. Ng and D.S. Hirschberg. Three-dimensional stable matching problems. *SIAM Journal on Discrete Mathematics*, 4:245–252, 1991.

[60] T. Nguyen and R. Vohra. Near feasible stable matchings. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC '15, pages 41–42, New York, NY, USA, 2015. ACM.

[61] G. O'Malley. *Algorithmic Aspects of Stable Matching Problems*. PhD thesis, University of Glasgow, Department of Computing Science, 2007.

[62] A. Podhradský. Stable marriage problem algorithms. Master's thesis, Masaryk University, Faculty of Informatics, 2010. Available from http://is.muni.cz/th/172646/fi_m.

[63] National Resident Matching Program. National resident matching program, results and data: 2015 main residency match., 2015.

[64] I. Ríos, T. Larroucau, P. Giorgiogiulio, and R. Cominetti. College admissions problem with ties and flexible quotas. *SSRN*, 2478998, 2014.

[65] A. Romero-Medina. Implementation of stable solutions in a restricted matching market. *Review of Economic Design*, 3(2):137–147, 1998.

[66] E. Ronn. NP-complete stable matching problems. *J. Algorithms*, 11:285–304, 1990.

[67] A.E. Roth. The evolution of the labor market for medical interns and residents: a case study in game theory. *Journal of Political Economy*, 92(6):991–1016, 1984.

[68] A.E. Roth. On the allocation of residents to rural hospitals: a general property of two-sided matching markets. *Econometrica*, 54:425–427, 1986.

[69] A.E. Roth. New physicians: A natural experiment in market organization. *Science*, 250:1524–1528, 1990.

[70] A.E. Roth. A natural experiment in the organization of entry level labor markets: Regional markets for new physicians and surgeons in the UK. *American Economic Review*, 81:415–440, 1991.

[71] A.E. Roth and E. Peranson. The effects of the change in the NRMP matching algorithm. *Journal of the American Medical Association*, 278(9):729–732, 1997.

[72] A.E. Roth, T. Sönmez, and M. U. Ünver. Pairwise kidney exchange. *Journal of Economic Theory*, 125:151–188, 2005.

[73] A.E. Roth, T. Sönmez, and M.U. Ünver. Kidney exchange. *Quarterly Journal of Economics*, 119(2):457–488, 2004.

[74] A.E. Roth, T. Sönmez, and M.U. Ünver. Coincidence of wants in markets with compatibility based preferences. *American Economic Review*, 97(3):828–851, 2007.

[75] A.E. Roth and M.A.O. Sotomayor. *Two-sided matching: a study in game-theoretic modeling and analysis*, volume 18 of *Econometric Society Monographs*. Cambridge University Press, 1990.

[76] U.G. Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54:57–67, 1992.

[77] L. Shapley and H. Scarf. On cores and indivisibility. *Journal of Mathematical Economics*, 1:23–37, 1974.

[78] J.E. Vande Vate. Linear programming brings marital bliss. *Operations Research Letters*, 8(3):147–153, 1989.

[79] H. Paul Williams. *Model Building in Mathematical Programming*. Econometric Society Monographs. Wiley, 1999.

[80] Laurence A. Wolsey. *Integer Programming*. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, 1998.

[81] H. Yanagisawa. *Approximation Algorithms for Stable Marriage Problems*. PhD thesis, Kyoto University, School of Informatics, 2007.

[82] Y. Yuan. Residence exchange wanted: a stable residence exchange problem. *European Journal of Operational Research*, 90:536–546, 1996.

[83] H. Zhang. *An analysis of the Chinese college admissions system*. PhD thesis, University of Edinburgh, School of Economics, 2009.

[84] D. Zuckerman. NP-complete problems have a version that's hard to approximate. In *Structure in Complexity Theory Conference, 1993., Proceedings of the Eighth Annual*, pages 305–312, May 1993.

[85] D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proceedings of STOC '06: the Thirty-eighth Annual ACM Symposium on Theory of Computing*, pages 681–690, New York, NY, USA, 2006. ACM.

[86] Canadian Resident Matching Service. http://www.carms.ca.

[87] Central Applications Office Ireland website. http://www.cao.ie.

[88] European network for research on matching practices in education and related markets - Matching in Practice Website. http://www.matching-in-practice.eu/.

[89] Higher Education Allocation in Hungary - Matching in Practice Website. http://www.matching-in-practice.eu/higher-education-in-hungary.

[90] Higher Education Allocation in Ireland - Matching in Practice Website. http://www.matching-in-practice.eu/higher-education-in-ireland.

[91] Japan Resident Matching Program. http://www.jrmp.jp.

[92] National Resident Matching Program. http://www.nrmp.org.