Automating the Discovery and Emergence of Life

Silke Asche

A thesis submitted in fulfilment of the requirements for the Degree of MSc(R)

School of Chemistry

June 2019

Abstract

The processes that led to the emergence of life is a not yet understood question. Bottom up approaches to the problem aim to understand how simple molecular building blocks give rise to complex, organized chemical systems. Historically, this approach has been limited by the combinatorial nature of organic chemistry and laborious analytical processes. In this work, these limitations have been circumvented by leveraging the power of automation to perform long-term, continuous experiments with the aim of constructing artificial life from abiotic starting points. To explore the complexification of simple molecule building blocks into chemical reaction networks, we perform recursive experiments. These are reactions that are periodically replenished with new starting material, thereby forcing the system far from equilibrium. This process leads to more complex product mixtures than non-recursive reactions, by inducing a historical dependence in the reaction network. We built a platform reactor system with in-line analysis, driven by a decision making algorithm and software with minimal human intervention. This gives us the opportunity to perform experiments 24/7, increasing the reaction throughput and screen a large chemical space in a short period of time. Using algorithmic feedback, the autonomous platform can make fast decisions on whether to change the chemical composition and reaction parameters. The ultimate aim of this project is to contribute to find the “missing link” in the progress of bottom-up synthesis of life like systems. The search for increasing complexity using analytical feedback allows us to look for the route by which simple molecules become complex systems, ultimately leading to a chemical to biological transition.

Table of Contents

Acknowledgement

The work detailed in this thesis was undertaken in the group of Professor Lee Cronin at the University of Glasgow School of Chemistry between July 2018 and June 2019. Many people have contributed to this work, directly and indirectly, and I sincerely thank them all, even if they are not named below. I would particularly like to thank:

Professor Lee Cronin for giving me guidance and support and for challenging me with projects I would have not dared to imagine before joining the group.

Dr. Geoff Cooper and **Dr. Cole Mathis** for all the support and guidance.

Graham Keenan for writing the platform code, showing me how to fix pumps and being endless patient with any of the billion code questions.

David Doran for all the scientific discussions that really taught me a lot.

I would also like to thank the whole Artificial Life and Complexity team and the glass office and especially my family and friends.

Abbreviations

In addition to standard notation, the following abbreviations were used in this thesis:

HPLC	High Performance Liquid Chromatography
DAD	Diode Array Detector
UPLC	Ultra Performance Liquid Chromatography
ESI	Electrospray Ionisation
MS	Mass Spectrometry
GC	Gas Chromatography
OoL	Origins of Life
BPC	Base Peak Chromatogram
TIC	Total Ion Current
EIC	Extracted Ion Chromatogram
MeCN	Acetonitrile
LC	Liquid Chromatography
ZIC-HILLIC	Zwitterionic-Hydrophilic Interaction Liquid Chromatography
DIN	Directions of the German Institute for Standardization
UVFD	UV Fluorescence Detection
ToF-MS	Time-of-flight mass spectrometry
DNA	Deoxyribonucleic Acid
RNA	Ribonucleic Acid

1 Introduction

The work undertaken in this thesis is driven by the question: How can we experimentally determine how life did, and does, arise? This is the theme of the project presented. Using automation as a tool to screen a wider chemical space, this work presents research in two fields. To increase the widespread acceptance of automation in chemistry especially for the exploration of chemical space, and to move a step further on the path to the origin of life.

This introduction is divided into two major sections: In the first section, the theories around the Origin of Life are highlighted. The discourse around the Origins of life field will be discussed as well as questions like: Where do we come from and how Life arose. In the second part of this introduction a small overview of the developments of automation in chemical laboratories will be given, highlighting ongoing work, advantages and disadvantages, as well as the use for the Origins of life research.

1.1 Origins of life

The origins of life are questions that have fascinated humanity for centuries. Nevertheless, it is still one of the biggest scientific problems of our time. We know how our planet did evolve and are able to determine how old our earth[1] the sun[2] and even our universe[3] are, but our life on Earth still appears as an unsolved problem. The reason solving this problem is so difficult, is that Life originated far back in prehistory and we have no Origin of Life occurring in our time, or no origin, which is not immediately out-competed or consumed by modern established life. This leads to the problem that we are not able to observe life emerging, or study it to draw conclusions from this process and make conclusions based on this observation about our own origin.

The origins of life research field is divided by many different theories and concepts that can be confusing (Figure 1) to those new to the field. The fact that there is no proven evidence on any fundamental theory on how life has started opens the door for many different theories and finding a common consensus in the field is

extremely difficult. This introductory chapter is going to give an overview of the main theories in the field and will highlight a niche in which this work could fit.

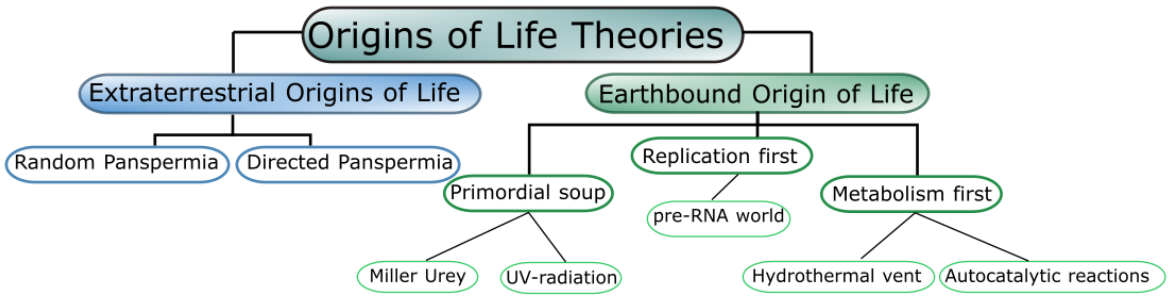


Figure 1: Overview of origins of life theories that will be discussed in this Chapter

1.1.1 The definition of Life

A big challenge in the origin of life research is not just the lack of proven evidence[4], it is as well the detection and the definition of when something is specified as alive. When approaching the problem of the origins of life, a different question must be put first. A definition of life itself is needed[5,6] as the scientist needs to define what to look for, when trying to approach the problem. Similar to the vast amount of origin of life theories4, there is an equal amount of definitions of what is life, as both of these problems are connected, none of them is proven in isolation and it is often difficult to distinguish between a theory of how life arose and what life exactly is. As there is no general or proven definition of life, all definitions used in origins of life research are used as a tool, to find a direction of how to answer the problem of how life started. Schrodinger[5] gave a series of lectures in Dublin in 1943, where he addressed the difficulties in finding an answer to this problem.

In 2010 Koshland suggested a definition where he presented his concept, depending on seven key points to define life, which he presented as the acronym "PICERAS"[7]. "P" stands for Program, where Koshland describes the setup of the living system, its chemistry and the environment. "I" means Improvisation, where the capability of adapting to the environment is included. Compartmentalisation is the third key point, which described the boundary of the living system and the

fourth point of Koshland's theory is Energy. The fifth point is Regeneration, taking into account that energy can be lost throughout chemical reactions, and needs to be restored to keep a system alive. With Adaptability, Koshland addresses the ability to acclimate for the system in case of a change of its chemistry, using this as an addition to the previous stated improvisation. Seclusion, which states the last key point, includes the selectivity of a living system. Further, this point includes the fact that favourable reactions and pathways would rather be preferred in the living system. Koshland's theory of his seven key points appears to be a too defined definition to describe any other kind of life than the terrestrial life.

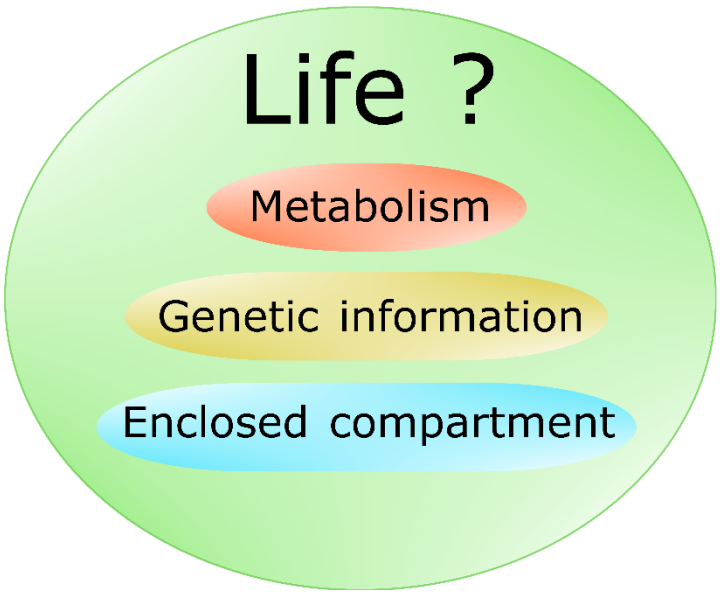


Figure 2: Based on the suggestions from Sagan[8] and Ruiz-Mirazo[11], the three key elements which define Life

Carl Sagan tried to split up the definition of life by different categories[8], like physiological, metabolic, biochemical, genetic and thermodynamic definition of life. He proposed that the physiological definition of life would be based on the functions of the living system, including the maintenance of a metabolism by consuming food, and producing waste as well as the ability to grow, reproduce and react on external factors. The metabolic definition stated is a living system with a distinct boundary, e.g. a cell that exchanges material with the environment. The biochemical definition suggested by Sagan is based on biochemical molecules, which can store hereditary information (as for example: DNA, RNA, proteins) and are able to metabolize and control the rate of enzymatic reactions. The genetic definition proposed by Sagan is based on the theories of Darwin's publication from 1859[9], where Darwin presents the theory of natural selection for the first time. Sagan describes this as the reason for molecules of higher complexity arising as they can reproduce and replicate, with an effect in which fitter genes are favoured. The last definition Sagan suggested was the thermodynamic definition, which stated that a living

system would use pathways, which would be thermodynamically favoured, and it would be part of an open system that is able to replicate.

In 1994, Sagan8 was part of a NASA committee which phrased a definition of life still in use today. Benner discussed this definition, which states that life is a chemical system that is able to sustain itself and able to go through a process of Darwinian evolution[10] and his suggestions show as well that this definition is again a tool, to design experiments to find life rather than the concluding definition of life.

Ruiz-Mirazo wrote an extensive review about the Origins of life field in 2011[11] where he suggested a consensus of three properties of life, here again first named the ability to handle heritable information, a metabolism that is far from equilibrium and a closed compartment to separate the living system from the environment (Figure 2).

Walker and Davies propose an alternative way to define life in 2012[12]. This is based on Joyce's[13] approach to the emergence of life based on that life did arise through the complexification of molecules which are able to self-replicate themselves which leads to the transformation into a living system. Walker and Davies are approaching the problem of how to define life from a new key aspect[12]. They define and distinguish dead from alive systems through their informational properties. The two main key points in this theory stand out as life itself is considered as being a dynamic system, which manages and controls information and is able to sustain its dynamics decoupled from the single components inside. Meaning that the living system does not react on individual single changes and can sustain itself, if once established and even through minor changes to the "reaction conditions". They further state, that the shift from a dead to a living system is based on a transition in the information of this system.

There are many more theories about the definition of life suggested and all these theories and definitions lead into completely different directions and are partially agreeing and disagreeing with each other. It is not clear if solving one question (What is life?) would lead to the answer of the other question (How does life arise?) but every theory on the origin of life mentioned below, is build on a theory of what life is.

1.1.2 Potential extra-terrestrial Origin of Life

With the field of prebiotic chemistry growing and the never-ending jungle of concepts, it is easy to forget the fact that it is not even known if life started on our planet. There are several theories about extra-terrestrial panspermia, the theory that microorganisms from outer space are responsible for life on earth[14]. On one hand, panspermia could have happened due to the spreading of spores or microorganisms due to meteorites. Some scientists believe that the heavy bombardment (which took place around 3.9 billion years ago[15)] happened just before life on earth arose15[,16] and experiments have been conducted to test if microorganisms would be able to survive under such extreme environmental conditions as they occur in space[17]. On the other hand, some scientists came up with theories about directed panspermia[18], suggesting that life was planted on earth by “alien organisms”18. Extra-terrestrial life or not, scientists agree on one point, if the origins of life did occur in a different location in our solar system, the problem about how life arose is simply just relocated, while the scientific question remains the same.

1.1.3 The emergence of prebiotic chemistry

Among the first pioneers addressing the question of the origin of life were the Russian scientist Oparin and the British scientist Haldane. Oparin published his theory in 1924[19], Haldane composed his theory independently from Oparin in 1929[20]. Interestingly, however both scientist proposed similar ideas about the origin of life. They present the hypothesis that the early Earth differed massively from the early earth conditions we know today[21] and state the assumption that the early Earth had a strongly reducing atmosphere, with methane, ammonia and hydrogen present. Further, both suggest that the ocean had an existential role in the build-up of "coacervates", droplets containing densely packed organic compounds. Through several steps, these were theorised to rise in complexity, leading to the assembly of the first cell. Both agree that a high amount of energy was needed to create the building blocks of life. The source of this energy is the point where both theories deliver different solutions. Oparin suggests that lightning is responsible for the formation of molecular assemblies that could act as precursor cells19 while Haldane suggests UV radiation as source for more complex molecules20 (Figure 3).

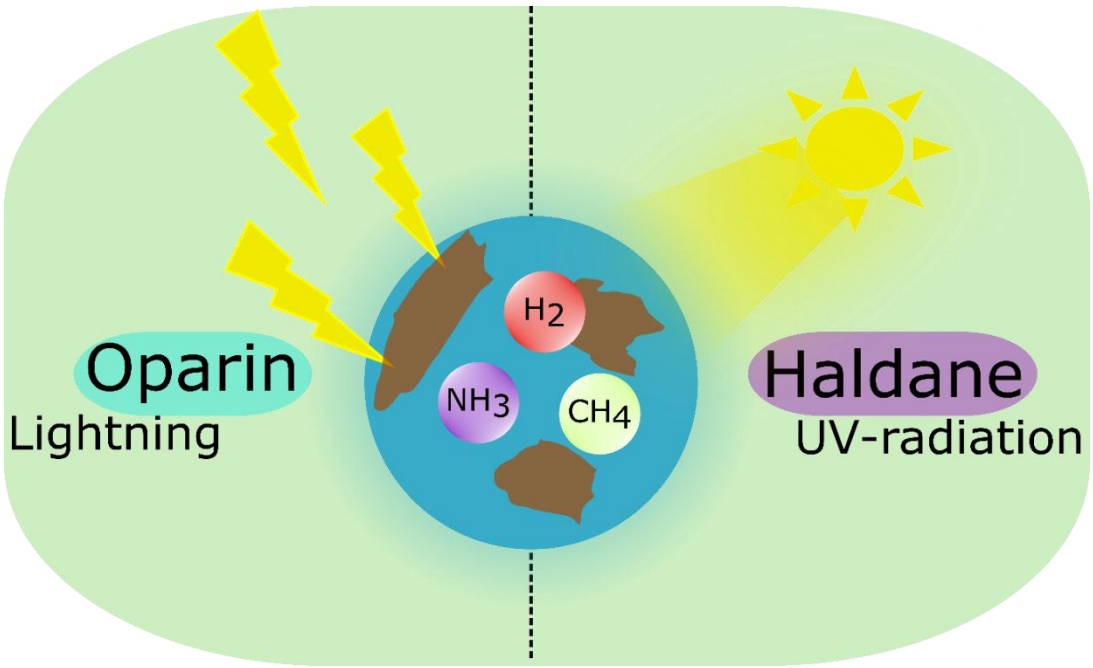


Figure 3: The early Earth as imagined by Oparin and Haldane. Hydrogen, ammonia and methane in high abundance with external energy provided by lightning (Oparin) or UV-radiation (Haldane)

Inspired by the “Oparin-Haldane Primordial Soup theory”, one of the most famous experiments in the field of chemistry was carried out. Miller, a student of Urey built an apparatus[22] where he was able to simulate the early earth atmosphere presented in this theory. The experimental build up consisted of a flask filled with water and heated to boil. This flask was part of a closed system in which the water mixes with the methane, ammonia and hydrogen gas and the mixture goes around a U- shaped tube. After the gas-water mixture passes an electric discharge consisting of two electrodes, which simulate lightning, the mixture would condensate and flow back into the boiling flask. Being very limited in their analytical techniques, only able to use paper chromatography, Miller identified glycine, α- and ß-alanine, aspartic acid and α-aminobutyric acid present in the product mixture. The demonstration that it was possible to make amino acids in this apparatus under the assumed early Earth conditions showed that it might have been possible that the key building blocks for proteins were available on the early Earth through the same chemical reaction. Miller later followed up his findings and reported that the mechanism, which led to the synthesis of amino acids in this experiment, was the Strecker synthesis,[23] where hydrogen cyanide and aldehydes, which are formed from the gas phase through the exposure to the spark discharge react together to form amino acids. Led by his former student Bada[24], old samples of the original experiments that were found after Miller retired, and have been analysed by more advanced methods, like HPLC-UVFD and ToF-MS. This led to the discovery of 22 different amino acids and five amines24. In this collection of old samples, a never before analysed experiment where H_2S-gas was added to the experimental apparatus was found and the analysis of this showed that Miller was able to make even more amino acids in this mixture than in the initial experiment. The sulphur-containing experiment[25] led to the synthesis of 23 amino acids, including examples never before seen generated from a spark discharge experiment, like threonine, leucine and isoleucine as well as 6 sulfur-containing amino acids, 4 amines and a range of organosulfur compounds.

With the first important prebiotic building blocks synthesised under early Earth conditions, a complete new field emerged. In 1960, Oró [26] found a mechanism for the synthesis of the nucleobase adenine under early Earth conditions, under the use of hydrogen cyanide as shown in Figure 4.

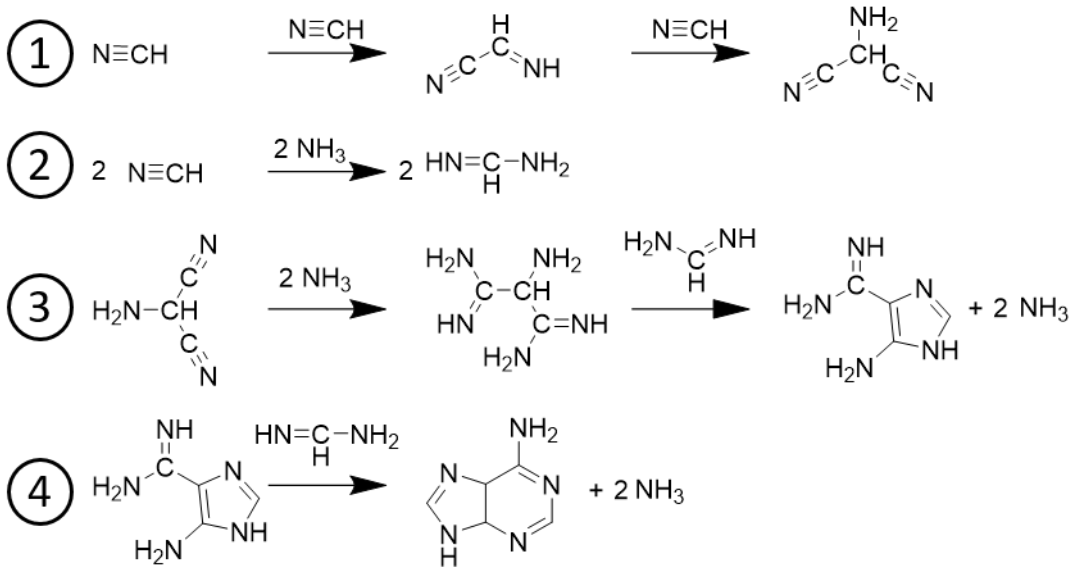


Figure 4: Synthesis of adenine in 4 steps, after the mechanism found by Oró26

The formose reaction is an autocatalytic reaction that makes sugars like ribose. Simple building blocks like formaldehyde and glycolaldehyde catalysed by calcium hydroxide lead to the desired products as it was discovered by Butlerov in 1861[27]. Breslow further investigated this reaction and presented a way in which sugars could have been built on the early earth as anticipated at that time in 1959[28]. Reid and Orgel[29] tested the reaction under different conditions by performing the reaction for example on an apatite mineral surface and with a more dilute formaldehyde solution. The emphasis in these studies was to make the reaction more “prebiotic plausible” by using milder conditions and a lowering the reagent concentration, but they concluded that it was not possible to lower the formaldehyde concentration under 0.01 M which they did not considered as an early earth likely concentration29.

The process of the discovery of the DNA double helix structure from its first isolation to the solving of the actual structure took almost 100 years The first isolation of DNA was done by the Swiss scientist Miescher[30], who first reported DNA without exactly knowing what his product was. Determining the structure of this molecule did require more advanced structural analytical techniques, which were not available at that time. Watson and Crick, together with Franklin and Wilkins succeeded to find the structure of a DNA double helix (Figure 5), which was published by the first two in 1953[31] benefiting of the advantages of X-ray

analytical techniques accessible at that time. The Nobel Prize in Physiology or Medicine honoured this breakthrough in 1962 for Watson, Crick and Wilkins.

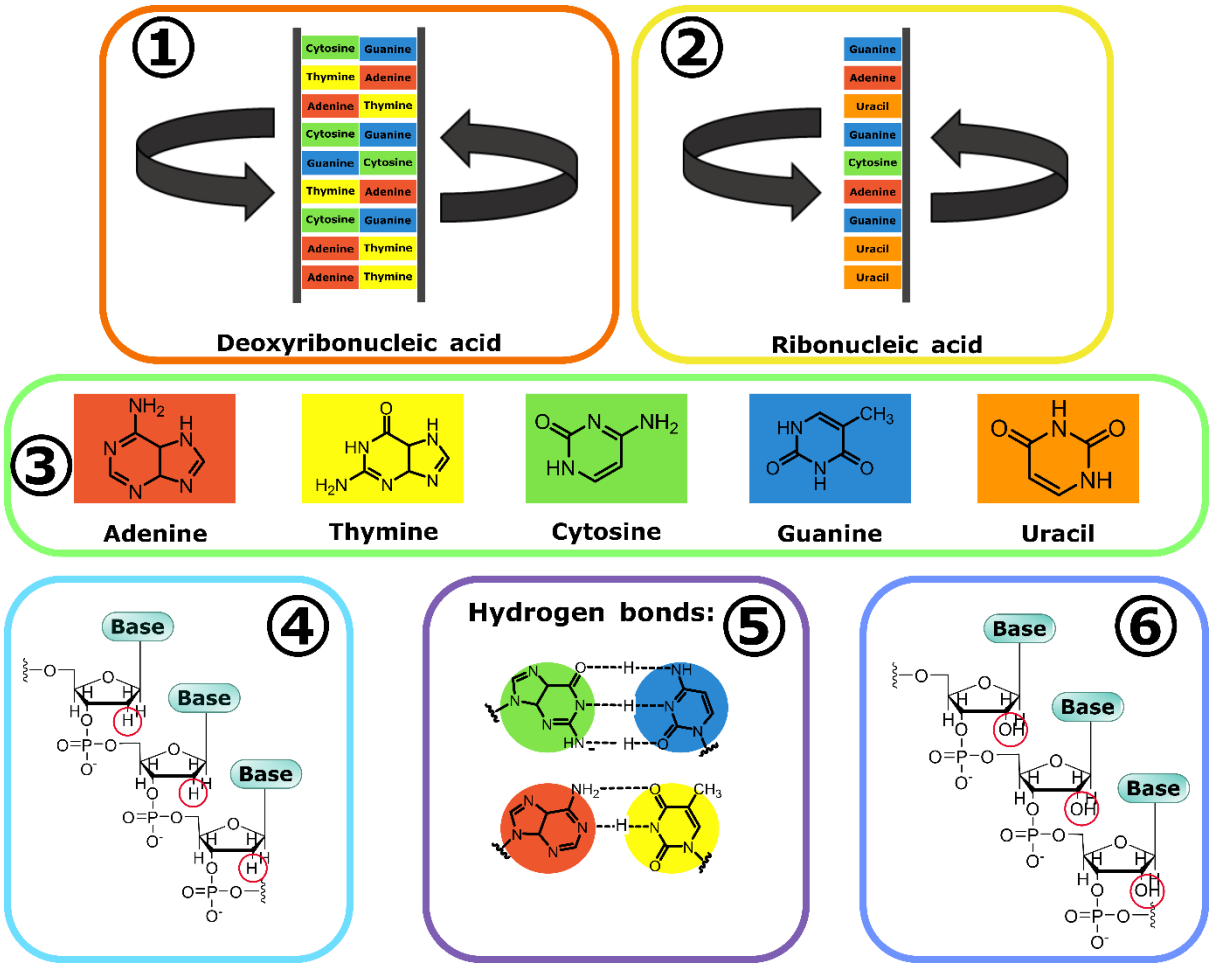


Figure 5: 1) Structural comparison of deoxyribonucleic acid (DNA). 2) Ribonucleic acid (RNA). 3) Their nucleobase building blocks, adenine, thymine, cytosine and guanine for DNA and adenine, cytosine, guanine and uracil for RNA. The helical structures are here presented in a linear form. 4) DNA backbone, consisting of phosphate bonded to deoxyribose, the functional group that is different to the ribose is highlighted. 5) Hydrogen bonds that connect the nucleobases in between the DNA structure. 6) RNA backbone containing phosphate and ribose. The hydroxyl-group of ribose, which differentiates it from deoxyribose, is highlighted.

Both deoxyribonucleic acid (DNA) and the closely related ribonucleic acid (RNA) consist of nucleobases bound to a sugar-phosphate backbone. They have a lot of similarities but big differences that make their unique tasks possible. DNA is the more stable molecule, being less reactive as its sugar is deoxyribose, consisting of one hydroxyl group less than ribose in RNA (Figure 5). As DNA is used to store and

replicate information, its high stability is beneficial, making it even stable in alkaline conditions[32], which would not be possible for RNA. The shorter RNA molecule has the ability to decode the genetic information stored in the DNA molecule and is much more reactive in comparison to DNA. The unique features or the RNA molecule leads to the next Origin of Life theory presented.

1.1.4 Replication or metabolism first

The Replication first theory35[,33] and the metabolism first theory44[,34] are the two most common theories discussed in the origin of life research. Both theories are looking into catalytic reaction networks, which lead to the complexification of prebiotic building blocks, but the difference is in the question, which molecules are needed for Darwinian evolution[9] to occur[4]. While the replication first theory suggests that a genetic polymer (RNA as self-replicating polymer, catalysed by itself) was needed for Darwinian evolution, the metabolism first theory proposes that Darwinian evolution can occur through complex reaction networks (enzymes, metal catalysts and peptides building complex self-replicating polymers), without the existence of a genetic polymer.

The focus in the field of the replication first theory is the model of an RNA world. This is based on the ability of the RNA molecule to act simultaneously as the primary genetic and catalytic polymer of a living system. Joyce[35] is one of the biggest advocates for this theory, suggesting that RNA arose directly from the early earth building blocks and developing DNA for storage of hereditable information. The ability of RNA to catalyse reactions inside ribosomes was discovered by Altman[36] and Cech[37] who shared the Nobel Prize for their work. RNA molecules with these properties are called ribozymes and can perform catalytic reactions almost enzyme like for the synthesis of peptides[38] and nucleotides[39] as well as the polymerization of more RNA molecules, which was reported by Johnston by using nucleoside triphosphate as a primer-template[40].

There are a few problems with the RNA world theory though. There are major difficulties in synthesising RNA, as the RNA molecule is unstable in water[41] and further, it is difficult to synthesise RNA under prebiotic plausible conditions42.

Powner and Sutherland are putting efforts into finding the missing links in that direction, but the furthest they got was the synthesis of pyrimidine ribonucleotides[42]. This synthesis itself from prebiotic building blocks might be interesting, but the reaction only proceeds under a very narrow set of constraints and is therefore unlikely to occur without human intervention. The RNA world is built on the premise that RNA can act simultaneously as the primary genetic and catalytic polymer in living systems. There is therefore no immediate need for DNA synthesis in an RNA world, even if an explanation of how the transition from RNA to DNA occurred is lacking. Coincidentally, just while this chapter was being written, an article on the missing step of how RNA could build DNA was published, by Teichert[43]. Where a way to synthesise DNA nucleosides from prebiotic building blocks in mild conditions was presented. For now, despite the evidence against it, the RNA-world theory might be a theory which could be proven right one day.

In the opposite of this theory is the metabolism first idea. The main key point in which, the metabolism first theory differentiates itself from the replication first theory is the fact that it suggests that Darwinian evolution was taking place before genetic polymers (i.e. RNA/DNA) arose. The metabolism first field is exploring catalytic reaction networks leading to the complexification of prebiotic building blocks. It is focussing on the chemical conditions on the early earth, looking into the effect of different metals or locations with special properties. Wächterhäuser shaped this theory with his Iron-Sulfur World Hypothesis, published in 1988[44] suggesting the synthesis of organic compounds through the energy released through the reaction of redox metal sulphide reaction systems. The Iron-sulfur idea is based on the abundance of iron sulphide on the surface of the early Earth. However, the Earths surface does not just provide interesting mineral compositions, in the ocean are hydrothermal vent systems, providing unique chemical environments and mineral surfaces in abundance49. Hydrothermal vents gained a rising interest in the Origins of Life (OoL) - research and provide a different approach to the metabolism first idea4. The fact that their environment provides a high energy and a carbon source and the amount of different inorganic surfaces that make the formation of biopolymers possible, draw high attention to this topic49. This theory does not need to be just earthbound valid, OoL researcher looking into life anywhere in our solar system are interested in hydrothermal vents. Rampelotto17 even suggested investigating hydrothermal vents, not just on

earth but as well on other planets of our galaxy. After being a pioneer with his Fe-S theory, Wächterhäuser, together with Huber, started looking into the build-up of amino acids[45] and peptides[46] in volcanic or hydrothermal vent systems. In 2000, a new type hydrothermal vent system[47] than the first discovered ones[48] have been reported. Previously, known hydrothermal vents have just been reported to be located along ocean ridges. These discovered "black smoker" systems have an abundance of metals like iron and manganese and an acidic environment with a pH range from 2 to 3 and a temperature of up to 405 °C[49]. The fluids in the black smoker contain high amounts of CO_2, H_2S, H_2 and CH_4. These newly discovered systems have extremely different conditions to the black smokers. These vent systems, often as referred to as "white smokers" have a more alkaline environment, with a pH range from 9 to 10. The temperature is usually lower than in the black smoker systems, as these vents are dislocated from the main oceanic ridges. There is an abundance of dissolved H_2 and CH_4 but dissolved CO_2 is thus far not reported.

These white smokers are in particular interesting for the origins community, as the process of serpentinization occurs through the unique conditions present. Serpentinization49 is a reaction in which the mineral olivine ($(Mg,Fe)_2SiO_4$) reacts with water to form the mineral serpentine ($(Mg,Fe)_3Si_2O_5(OH)_4$) and the production of magnetite (Fe_3O_4), brucite ($Mg(OH)_2$), CH_4 and H_2. Through this reaction substantial amounts of energy and chemical resources as well as a mixture of different metal catalytic compounds are available and could react together to build a metabolic or autocatalytic reaction network[50].

Autocatalytic networks are especially interesting for the metabolism first field, as this field does not deliver the one "universal replicator", in opposite to the RNA theory, which does build on the same time on autocatalytic networks too, but all based on the fact that the RNA molecule can replicate itself. The focus to understand the self-replication process in general and to find molecules, which are able to self-replicate, builds a key part of the question, of how life was able to arise. A range of very complex autocatalytic processes happen in living cells, but the level of complexity of these processes, makes studying autocatalysis in cells to a whole standalone research question. An autocatalytic network is defined as a set of reagents or "objects" which can replicate themselves[51], if once build,

which is in some cases even possible from more simple building blocks, leading to a reaction system, which can sustain itself. As an example for such a chemical autocatalytic reaction network and one of the most prebiotic interesting one is the tricarboxylic acid cycle[52] (Figure 6), as it provides a method of carbon fixation[53] and the build-up of a range of interesting organic compounds. The Moran group[54] is exploring this autocatalytic reaction cycle and not just succeeded in performing these reactions purely catalysed by metals (i.e. iron) instead of enzymes, but found links from the tricarboxylic acid cycle to the glycoxylate cycle, which is another prebiotically plausible-, autocatalytic network.

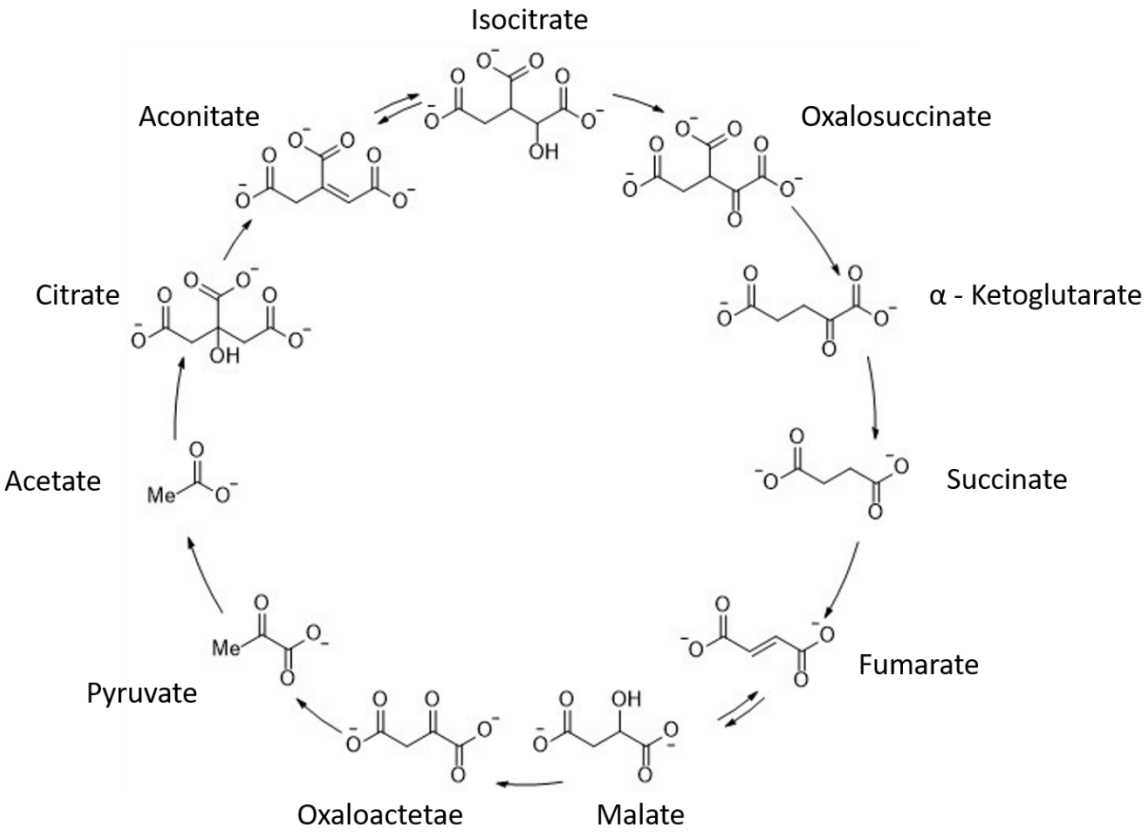


Figure 6: Product overview of the citric acid cycle

Of course, in a field with a lack of fundamental proven facts, there are far more theories about the Origins of Life. Nevertheless, for this work looking into the main theories shall be sufficient.

1.1.5 Our approach to the Origins of Life

All the theories above are different ways to try to approach an immense problem. Approaching this from a purely chemistry background, we attempt to build-up complex functional networks from small simple building blocks in a bottom up approach. Rather than looking into prebiotic plausible chemical options, simple chemical building blocks are used, leading to the complexification and the build-up of a reaction system. Instead of looking into distinct individual products, we are aiming to look into reaction networks from a system point of view, which is partially inspired by the definition of Life earlier mentioned12 by Walker and Davies from an information level. By looking for a process of emergence[55], where a chemical system goes through a transition, reaching a new biological stage[56]. It is necessary to understand that, while a bigger system is made out of small things it does not necessarily follow the same rules and fulfils the same properties, as its smaller building blocks. Through the emergence of a new bigger macro state, a bigger system or a complex messy chemical mixture can develop new distinctive system properties, compared to the properties of their building blocks and should therefore be viewed as two different things[57]. The aim in studying these systems is not limited to the attempt to find a missing link for the Origin of Life but rather an approach to find any kind of complex system that could make the rise of artificial life possible.

With this in mind, we build an automated system, which mixes simple chemical building blocks on a mineral surface in recursive cyclic reactions together. The reaction mixture was analysed in-line and an algorithm analysed the data and made input decisions based on the amount of change happening in between the reaction cycles. Running repeated cyclic reactions with distinct time points, is not feasible in the time constrains of a normal lab workday, but automation can lead to the ability to run reactions continuously 24/7. The development of automation in chemical labs and the advantages of this tool shall be further discussed in the following section.

1.2 Automation in Chemistry

Automation can have major advantages, not just in big scale industry processes but also directly on the lab bench of every researching scientist. Speeding up iterative and, for most chemists, “boring” lab tasks, making potential dangerous lab tasks safer by making on demand synthesis possible or just limiting the amount of contact the chemist has with hazardous materials. On top of that, reproducibility is always a challenge in the discovery of new compounds. Automation might be the answer to all the above, but especially the last problem.

In this chapter, we will look into developments of chemical automation and highlight current issues, bottlenecks and fields that have been engaged in this process. We will only look at developments in automation in bench scale chemical systems. Today, laboratory automation progressed fast, especially in the industrial sector with massive plant-scale production reactors.

The historian J.T. Morris described in his book[58] the development of chemistry laboratories in history. He builds his conclusions based on pictures from actual chemistry labs and through these, he reaches the conclusion that the standard chemistry laboratory bench set up has not changed much compared to laboratory set-ups in the past. Describing through pictures and photos is a very distant way to assign something and based on this he might miss the revolution happening in chemical labs, in the past century. The normal bench space is one of the only work environments that has not engaged much with the current trends in automation and digitalisation that happened in other scientific fields. However, even if we do not have massive automated set-ups on every bench in every lab yet we can find automated systems in every corner of a modern chemistry lab and this has changed the way in which chemists approach problems completely. In fact, we are so used to having these automated systems that we do not necessarily see them as what they are: automated lab robots. Peptide synthesizers, automated flash columns, auto samplers in LC systems and many more “hidden” automated systems are part of the standard kit in any big synthetic or analytical laboratory today.

1.2.1 Solid state peptide synthesis and the beginning of automation in chemistry

The first mention of an automated process in literature was in 1875 by T. M. Stevens[59]. He built a system that let you filter and wash substances unattended. At that time, even the automation of very specific tasks was a revolutionary step. R. B. Merrifield found a way to automate the synthesis of peptides[60] in 1965 by developing a way to synthesise peptides sequences and bond them to a supporting resin[61]. This is an especially useful technique for linear polymers, which contain the same iterative bonding motif used to link monomers together (Figure 7), as well as this practise suits itself through its sequential manner, perfectly for automation.

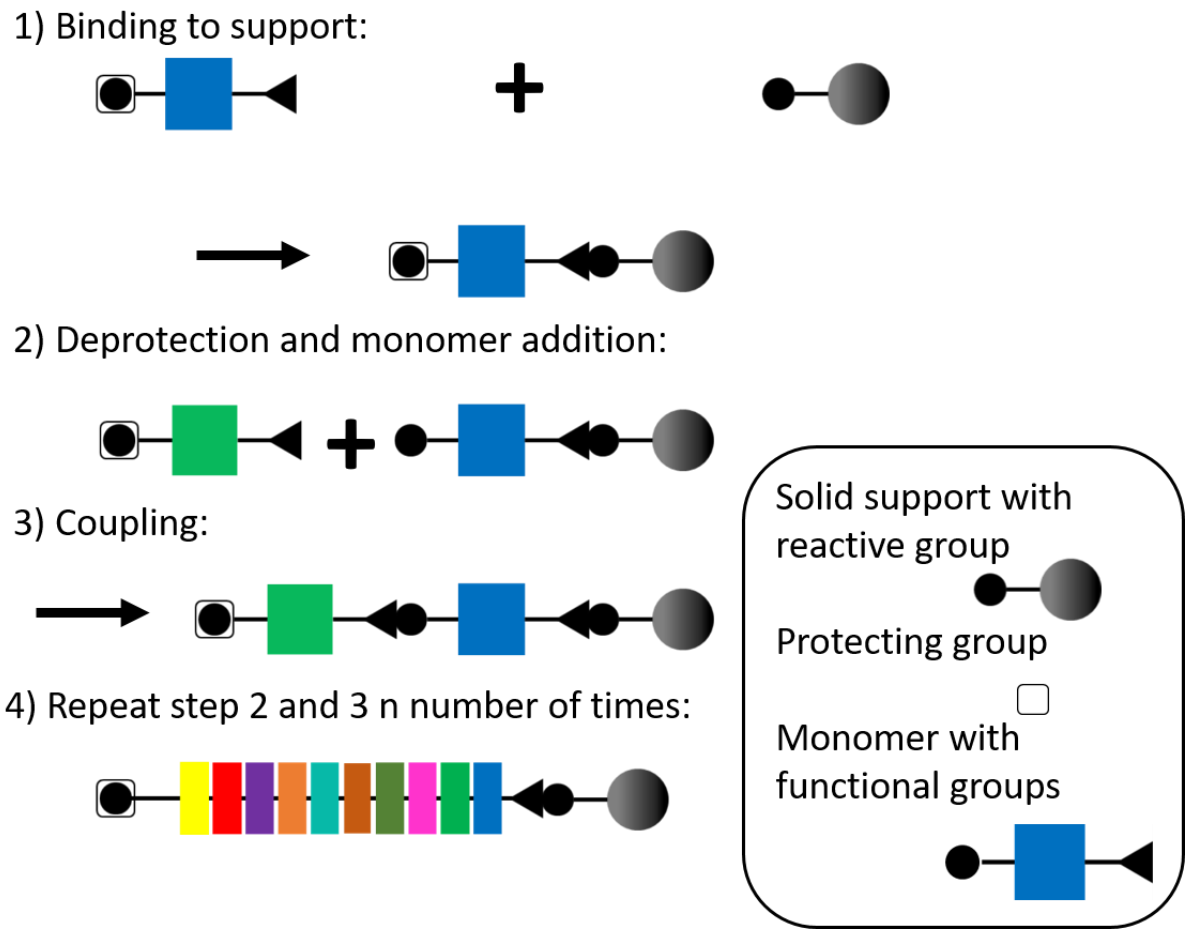


Figure 7: "Solid Phase Peptide Synthesis" Sequential synthesis of linear molecules on solid support

This resin is a polystyrene that was chloromethylated and bonded through a covalent benzyl ester linkage to the first amino acid of the sequence. Through this, the monomer sequence is kept in the solid phase of the reaction, making it

possible to do necessary purification steps just through washing with solvents and simple filtration. This leads to an improved yield, as coupling agents and amino acids can be added to the reaction in large excess, and products are not getting lost through any complicated purification steps. The peptide chain was synthesised by the iterated addition of protected amino acids, deprotection of the amino acid monomer and purification. Finally, after the peptide reached the desired length, the peptide was separated from the resin through treating the product mixture with HBr61 (Figure 8).

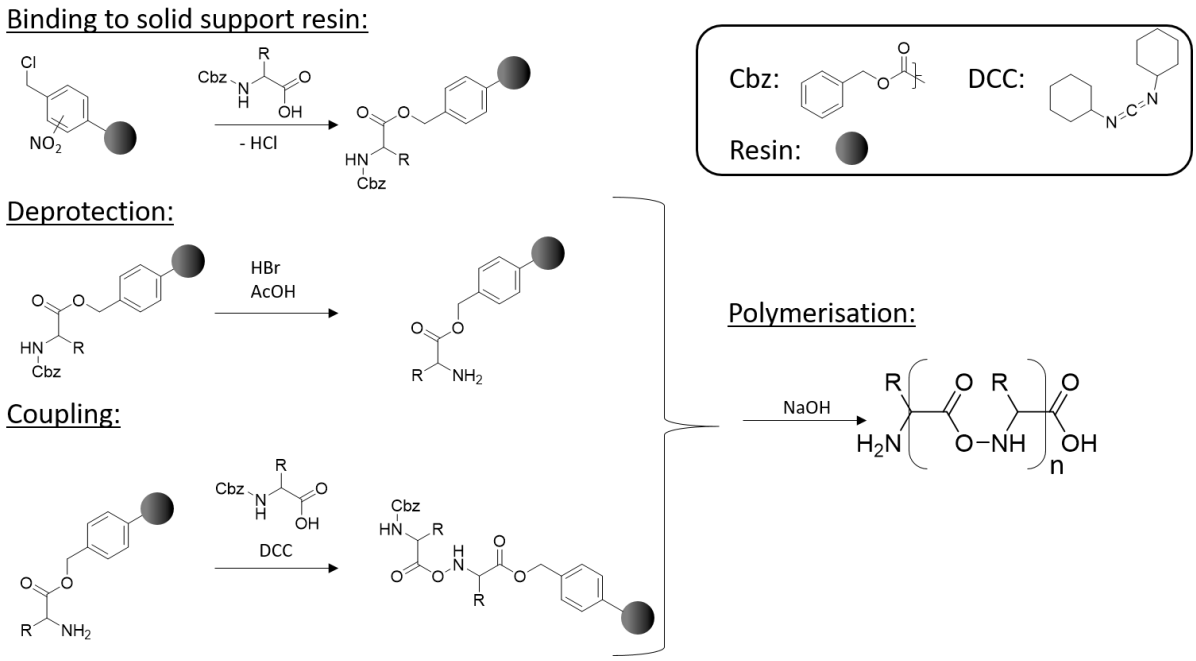


Figure 8: Solid phase peptide synthesis based on Merrifield's work61

The schematic build-up of Merrifield's machine looked, for its time, already very advanced[62]. The apparatus had two separate 12-way rotary selector valves of which one is used for the selection of the amino acid monomers while the other valve was exclusively used for solvent transfer. The reaction vessel itself was fitted with a frit with the supporting resin laid on top of it and placed on a shaker module, to ensure mixing between the reagents, solvents and resin.

Being able to attach six amino acids to a peptide chain in a 24-hour period was a small but ground-breaking step in the automation of chemistry. But not just that, Merrifield gained a massive advantage to the general field of organic synthesis by being able to synthesise long peptide chains without doing everything manually, compared to everyone competing in that field at that time. He was able to synthesise new molecules like ribonuclease A[63] in just a year after building the

automated platform[64] and received the Nobel Prize in Chemistry in 1984 for his solid phase peptide synthesis. On top of that, the most impressing thing about Merrifield's work is the timelessness. Even after more than 50 years since the development of the solid peptide synthesis, most commercially available machines are using the same exact principle of bonding the peptide to a supporting resin. Of course, there is a big variety of new resins, coupling agents and protected monomers but the way the process takes place is still the same.

1.2.2 The way to the first DNA synthesiser

In the 1920s, automation of laboratory processes became a new era[65] with a bigger group of scientists trying to automate botanical research, and particularly, creating an automated tool for liquid-liquid extraction. Palkin *et al*. succeeded in that effort[66] and published in 1925 the first continuous liquid-liquid extractor. So far, automation was basically just in the field of synthetic chemistry. Analytical chemistry was at that point behind in the automation development process. But closely following Palkin, a fully automated titration machine was developed in 1929 in New York59 that indicated the colour change through the help of a photocell. Surely, automation had already by that time proven very useful just to save the scientist time, but there is another massive advantage that comes with automated systems that was especially needed in the time of the Second World War, where trained laboratory staff was difficult to find. The fact that automated systems can make some specific task much easier to perform, on such a scale that trained personnel were not necessarily needed anymore, was especially very attractive for those trying to automate laboratory work.

As this advantage is rather useful for industrial chemistry labs, it is no surprise that in 1941 the Shell Oil Company developed a new type of titrator machine that was so advanced, it made the operation by a non-chemically trained person possible. On top of that, this titrator indicated the neutral point by measuring the electric potential, using a potentiometer, which made the use of an additional indicator solution unnecessary. The user of that machine just had to load the machine, press a button and read the titration result as the total volume of liquid in the titration chamber, as measuring the amount of titrant used was not possible

to automate at the time. Lingange et al. developed a “fully automated titrator” by 1984[67], this machine used a syringe to add the titrant in different adding speeds and allowed as well to execute different types of titration with one machine. Additionally, the apparatus was able to directly plot the resulting titration curve. One of the most time-consuming laboratory processes is chromatography. With paper chromatography being one of the most basic of them, in 1958 Martin[68] was able to develop an automated chromatography apparatus making it possible to just start and leave the chromatography unwatched until the next day.

Medical laboratories have always had to handle a high amount of samples. The need to find a way to analyse massive sample amounts was higher than in most other chemical fields and big efforts have been made from the industry side of the development of laboratory robotics for clinical use59. In the 1950s company instrument suppliers stared to enter the automated lab technology market and high-throughput screening technologies especially needed in the medicinal chemical industry was in high demand[69].With more advanced automated machines, the desire for complicated synthetic molecules rose.

Until 1981, one of the most demanded but synthetically difficult molecules to synthesise were polynucleotides[70] or DNA fragments. The standard synthesis of a 12- unit sequence could take a trained, highly skilled chemist around 3 months, according to Alvarado-Urbina70. However, as the synthesis of polynucleotides is an iterative process of adding monomers to a long molecular chain, scientists thought the automation of this process would be feasible. The fact that Alvarado-Urbina et al. have been able to overcome this challenge, is due to a few accomplishments in the general method development in the synthesis of DNA fragments over the years. Khorana developed the “diester method” in 1961[71]. A condensation reaction of protected mononucleotides resulted in the formation of a dimer. This was a major breakthrough in the attempt to make a chain of polynucleotides, as the procedure could be repeated, but at the same time was extremely slow and produced a massive amount of salt by-products. This salt product problem and the condensation time were improved in the “triester method” published by Letsinger and Ogilvie in 1968[72]. Letsinger improved the synthesis further and published in 1975 a method to synthesise nucleotide chains without major impurities or day-long condensation times[72]. Based on the solid

support peptide synthesis method that Merrifield invented, Alvarado-Urbina et al. started to develop ways to improve the synthesis of polynucleotides for automation and searched for a useful support material for their desired product. Matteucci et al.[73] found silica to be very useful for this synthesis and Alvarado-Urbina et al. developed a way to prepare silica gel with the first protected nucleoside chain attached via five reaction steps. After these preparation steps, the prepared silica was packed into a column that was inserted into to the actual automated machine. The platform was built from two pumps, one to dilute solvents and reactants and one to transfer the liquid onto the column in a constant flow rate of 5 mL per minute. The control over the liquid flow was achieved through a series of pneumatic valves. The sequence of the desired polynucleotide was computationally-controlled. The improvement and automation of this synthesis made an extensive lab procedure into an easy routine task and reduced the time chemists needed to synthesise a polynucleotide chain of up to 14 monomer units from several months to several hours. Over the years the automated systems were more and more improved, especially through the work of Horvath et al.[74] who published the development of a nucleotide synthesiser in 1987. This machine used the more robust phosphoramidite method (Figure 9) and today it is easy and cheap to just order artificial made DNA made out of several thousand base pairs from chemical companies.

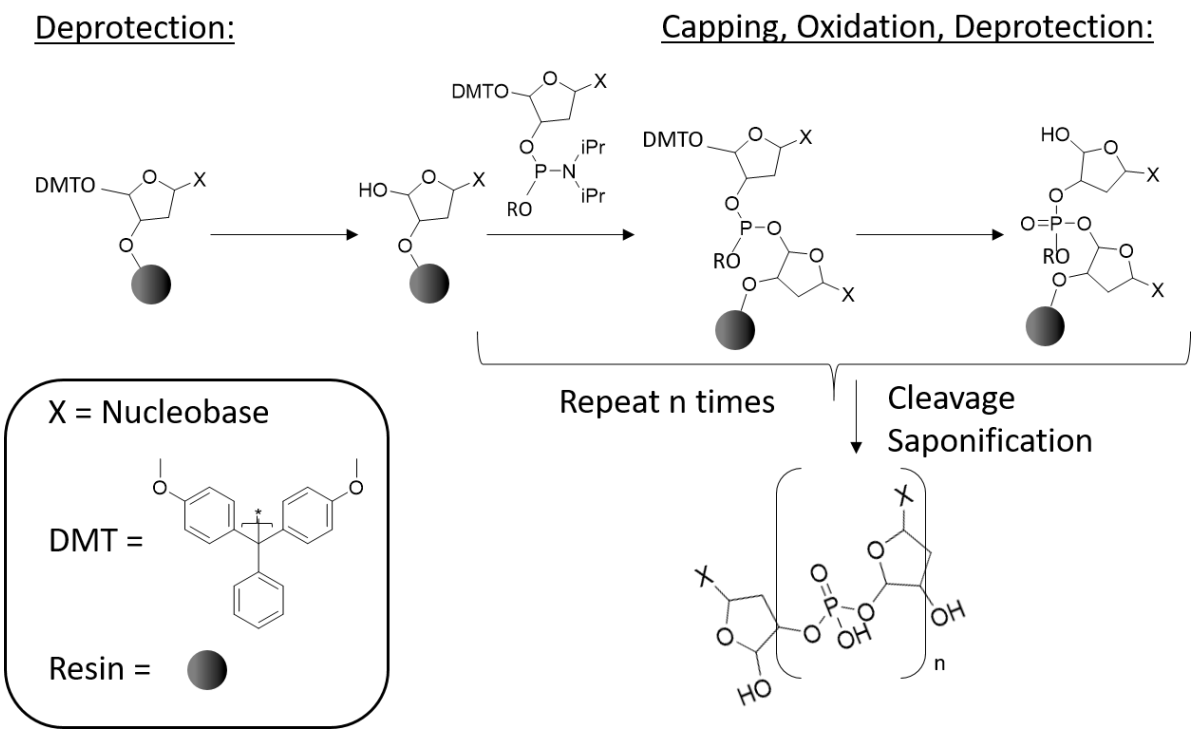


Figure 9: Oligonucleotide synthesis with the phosphoramidite method74

This changed many research fields, especially molecular biology, where one famous reaction would not be possible without the easy supply of oligonucleotides. The polymerase chain reaction uses these as a primer to make the replication of DNA strands possible[75]. Kary Mullis received a Nobel Prize in Chemistry in 1993 for the discovery of this reaction and it made research in DNA cloning, and sequencing possible as well as enabling DNA profiling and many other highly used modern techniques.

One of the most useful tools ever made for the analytical lab is the auto sampler. It took time until this tool was established, as it required the existence of robotic platforms and robotic arms. The first report of the development of a robotic arm was in 1983,[76] actually in synthetic chemistry[77] where Fuchs *et al*. built a robotic system to optimise organic reactions. The idea of using an automated, robotic system for the development of new compounds and reaction optimisation became more popular in the 1990s, when companies developed the first workstations to automatically prepare samples and run them in an online HPLC using an auto sampler[78]. Following these developments, these workstations started to be well established, especially for the use of compound screening in medical labs59.

1.2.3 The automation of oligosaccharide synthesis and a small excursion into flow chemistry

By the 1990s, automated systems for peptide and polynucleotide synthesis had been realised and both developments have been major breakthrough in the field of automation and synthetic chemistry itself. However, one molecular class was not automated yet and it took more than 20 years until another automation breakthrough was published. In 2001, Plante and co-workers published the automation of the synthesis of oligosaccharides[79]. This development was just possible after several improvements in the field of polysaccharide synthesis, especially the findings of Schuerch[80] who set a milestone in that process in 1971, as they developed a way to synthesise oligosaccharides similar to the approach Merrifield invented for peptide synthesis, by supporting the product chain on a solid phase resin. It is surprising that the actual automation of this synthesis took so many years afterwards, as in the same time, much progress was made in the

automation of polynucleotides. Until 2001, the synthesis of oligosaccharides would require high technical expertise and could just be executed in specialised laboratories, where it would remain a difficult and prolonged task. Similar to the automation processes before, the solid-phase synthetic approach proved to be the way to get the desired end product. The product needs to be supported on a solid-state resin to archive a high yielding reaction and to enable the addition of vast amounts of reagents that could be easily removed through filtering, as well as providing a straightforward way of purification through washing. Different to the synthesises before, the functionalisation and/or protection of multiple hydroxyl groups selectively at the time proved to be a challenge. Plante et al. developed a "protecting group scheme" were they have been able to manipulate each hydroxyl group on its own and used benzyl ethers and esters as either permanent or temporary protecting groups. Another challenge was the bond formation of glycosidic bonds and their stereoselectivity (Figure 10).

To link two saccharides together, the glycosyl donor need to be activated to perform as an electrophile, reacting with a glycosyl acceptor operated as a nucleophile, based on the orientation in which the nucleophile attacks, the resulting carbohydrate configuration can be either an α- or a ß-anomer.

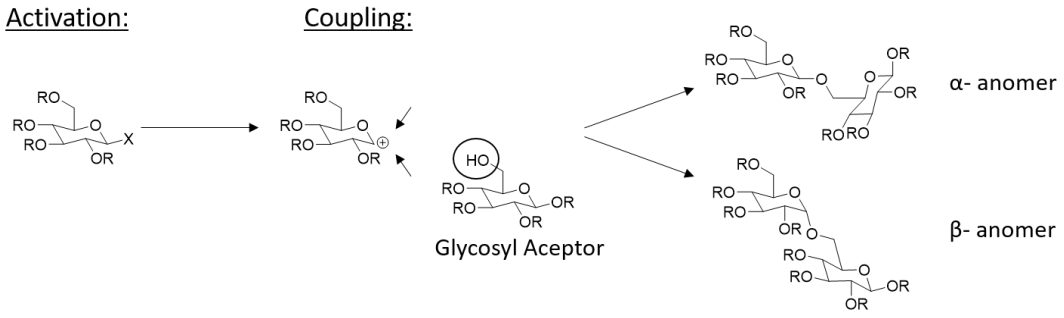


Figure 10: Carbohydrate synthesis and its stereochemical issues

As the configuration of the molecule is an important factor for the following synthesis, the group needed to find a way to control the configuration of the synthesised carbohydrate. Plante et al. found that the attachments of different functional groups to the glycosyl donor can sterically control the carbohydrate configuration (Figure 11).

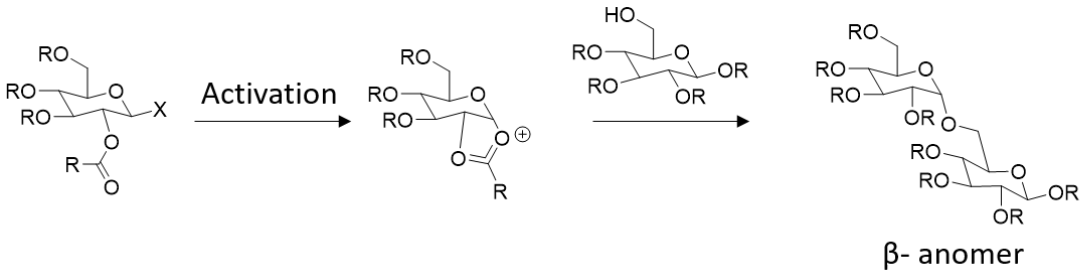


Figure 11: Stereospecific controlled carbohydrate synthesis: A functional ester group is used to control in which orientation the glycosyl acceptor attacks

Based on these developments, Plante *et al.* were able to modify a peptide synthesizer for their needs, adding a temperature-controlled reaction vessel and finally succeeding in the synthesis of several different oligosaccharides in a very similar way to Merrifield's strategy to synthesise peptides as shown in Figure 12. In their first paper79, they presented the automated synthesis of five different polysaccharides.

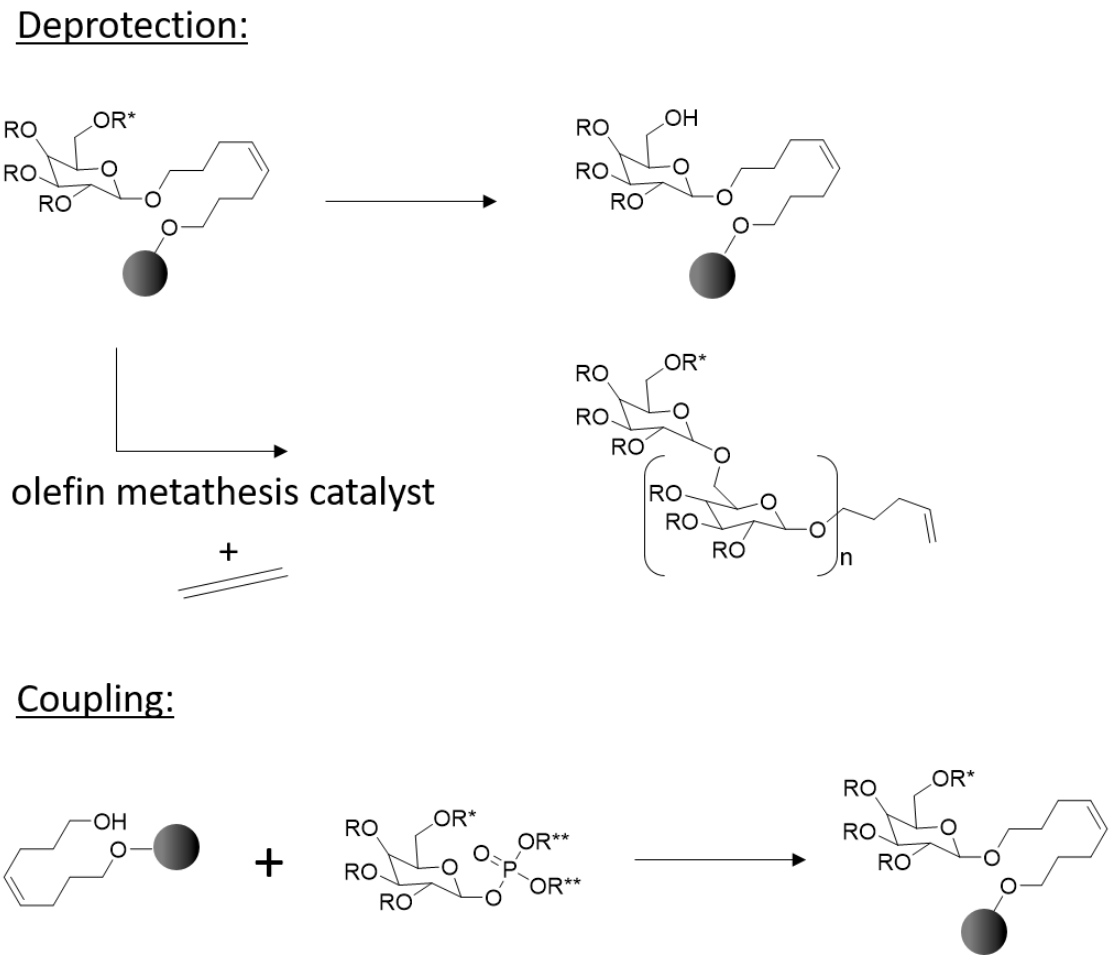


Figure 12: Principle of coupling sequence developed by Plante *et al.*[79]

Due to the molecular complexity, bespoke reagents were necessary for each specific synthesis but it was clearly a step forward to simplify the synthesis of oligosaccharides. Compared to synthesis by-hand, which would have taken 14 days to synthesise a heptamer, the automated synthesiser was able to make the molecule in 20 hours and with a higher total yield. Based on these developments, the Seeberger group was able to synthesise a branched dodecamer in 2004[81] and a 50mer polymannoside was reported in 2017[82]. To commercialize their invention, Seeberger patented the automated synthesis of oligosaccharides and founded a company, GlycoUniverse, which provides polysaccharides, their reagent solutions and a custom-made commercial oligosaccharide synthesiser called the GLYCONEER®.

Seeberger was apparently not just interested in the automation of a glucoside batch synthesiser, along with Plutschack, Pieber and Gilmore, he published a review paper[83] about another automation field, which we have not highlighted yet. Flow chemistry, a niche field in with its own advantages and disadvantages. The principle is the use of channels instead of a big batch reactor. It involves a collection of channels and tubes, and the reaction proceeds completely inside the tube. The reagents are added through a pump and as the liquid moves continuously, the reagents react in flow. Depending on the choice of tubing, this even opens up opportunity for reactions under high pressure, with liquid-gas mixtures, or in exothermic systems without having the need of an expensive and complicated temperature control. Due to the low volume in the tube at a given time, flow chemistry can improve the safety of a reaction and additionally, toxic compounds may be made in situ and immediately reacted further.

1.2.4 Most recent developments and the path to the universal synthesiser

By 2001, the synthesis of three different types of organic molecules had been successfully automated. All of these molecules had a similar synthetic pathway, chain elongation of a repeating building block, by iterative synthesis steps of deprotection/activation, coupling and washing. In 2009, a group from Tokyo published a paper about an automated platform that is able to synthesise a number of organic molecules[84]. The "ChemKonzert" can conduct a multistep synthesis in a one-pot reactor and is even able to separate solids from the product mixture.

An industrial approach to the automation of chemical processes was reported in 2013 from the pharmaceutical company, Eli Lilly69. They described a fully automatized robotic driven medicinal chemistry lab. Tring to combine combinatorial chemistry, the exploration of many molecular libraries and an automated workstation together. This workstation was split into four parts, where the first section was used to perform the chemical experiments. Bench robots transported the samples from one stage to the next and even heating and refluxing the reactions was possible. The second part of the workstation was reserved for special functions to execute microwaved reactions or to cool reactions down using a cryogenics reaction platform. The third part of the workstation was built out of three customised platforms to work up samples for their analysis. Godfrey69 et al. reports that the workstation was able to perform over 60 different reaction types in this robotic system.

The latest challenge chemists interested in automation are facing, is the development of a machine, that can synthesise any molecule desired: a "Universal Synthesizer". So far, every major automated synthesis was based on methods where the product is bound to a solid support material. With a universal synthesiser molecular approach, finding a "universal" resin is an impossible task. Completely new approaches to synthesis and automation are needed. In 2015, Burke and his group set the first milestone in this development, by building an automated platform that can synthesise a range of small organic molecules[85]. These are made without solid support resin but are all based on the same reaction mechanism. The trick in this approach is the use of N-methyliminodiacetic acid

(MIDA) protected building blocks that have been functionalised with a boronic acid and a halogen moiety[86]. This functionalised building block undergoes iterative Suzuki-Miyaura coupling to form a new C-C bond between the building block and an unprotected organoborate. Due to the MIDA protecting group, the reactivity of the boronic acids is lowered and uncontrolled oligomerisation is prevented.

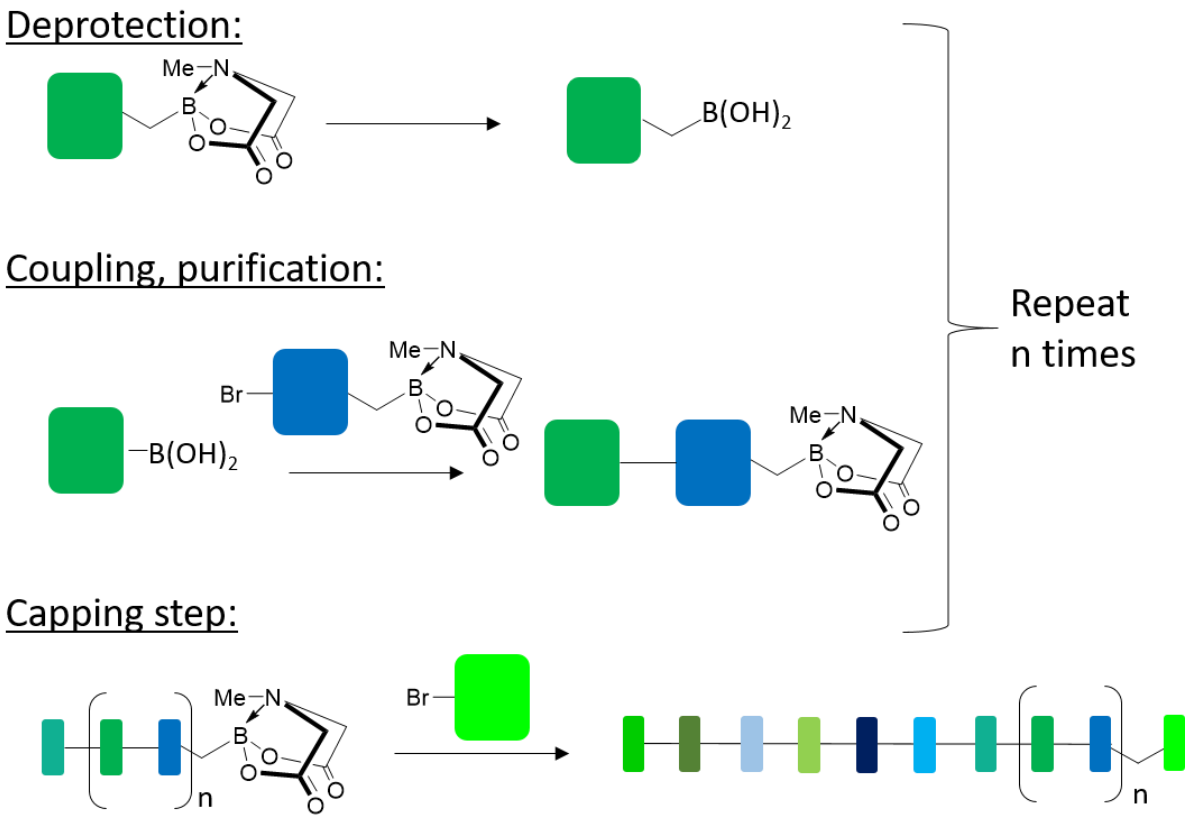


Figure 13: Schematic iterative Suzuki cross coupling using MIDA-protected ligand86

The MIDA protecting group has another very useful advantage. Through its high affinity to silica, it is possible to bind the MIDA coupled product temporarily to a silica resin and wash off impurities and excess reagents. After the washing step, the MIDA-bonded product can be washed off with tetrahydrofuran (THF). This "catch-and-release purification" led to a few disadvantages though. The MIDA building block was the limiting factor in the synthesis. It was not possible to add an excess of this starting material, as unreacted building blocks would bond to the silica phase in exactly the same way as the desired product. On another hand, at the end of a synthesis, the product was reacted with the final halide building block and as the MIDA was removed, there was no purification available for after capping the sequence, resulting in the need to manually purify the end product. For the

automation of this synthesis, Burke and group developed a procedure, which was separated into three modules, deprotection, coupling and purification to run every synthesis step sequentially. All reagents that were used had been pre packed in reaction cartridges to make the platform handling as easy as possible. In the paper from 2015 Burke et al. reported that they have been able to produce in total, 14 distinct classes of small molecules[85] and later published papers[87,88] suggests that a true universal small molecule synthesiser has not been realised.

Automation brings new interesting opportunities to the common chemistry lab, and should be more widely employed, especially by synthetic chemists. This will make a completely new approach to chemical discovery possible and open up new prospects for chemists[89]. An established synthesis might not be able to be massively improved or simplified but by inventing a universal machine that can perform the synthesis for the chemist, a lot of time is saved and the chemist gains time to look into more useful things, such as exploring new unknown synthesis, rather than repeat the same synthesis over and over again.

Cronin and co-workers have led the most recent developments in the universal automation of chemical synthesis. In keeping with his goal to "automate and digitise" chemistry, the Cronin group published three different automated systems for chemical synthesis, each with a different set up and approach and each platform pushing its field and the development of automated chemistry a step further. In July 2018[90] Granda, Cronin et al. published an automated platform that was not just able to perform a wide range of known organic synthesis but was able to explore unknown chemical space. By using a machine learning search algorithm, the platform was able to find four new organic reactions from common small reactive organic building blocks. Granda was exploring two- and three-component reactions, with immediate feedback through inline analytical feedback from NMR, MS and IR. With this platform, it was shown that the use of machine learning algorithms to explore chemical space was very useful and could lead to the fully automated discovery of new organic chemistry. Just a month after this work was published, Cronin's team reported another automated platform with a complete different idea. Caramelli et al.[91] developed an automated platform that was based on easy liquid handling with a focus on simplicity and affordability. In fact, they build several small platforms consisting of cheap peristaltic pumps, a webcam for

reaction monitoring and a pcDuino3 board to control the hardware through a computer. The platforms were connected to each other to enable a shared exploration of chemical space or the investigation of reaction reproducibility and to communicate experimental results in real time.

The most recent publication in the field was based on similar ideas to Burke's universal synthesiser. Steiner et al.[92] developed a fully automated batch scale organic synthesis platform, the "Chemputer" (Figure 14).

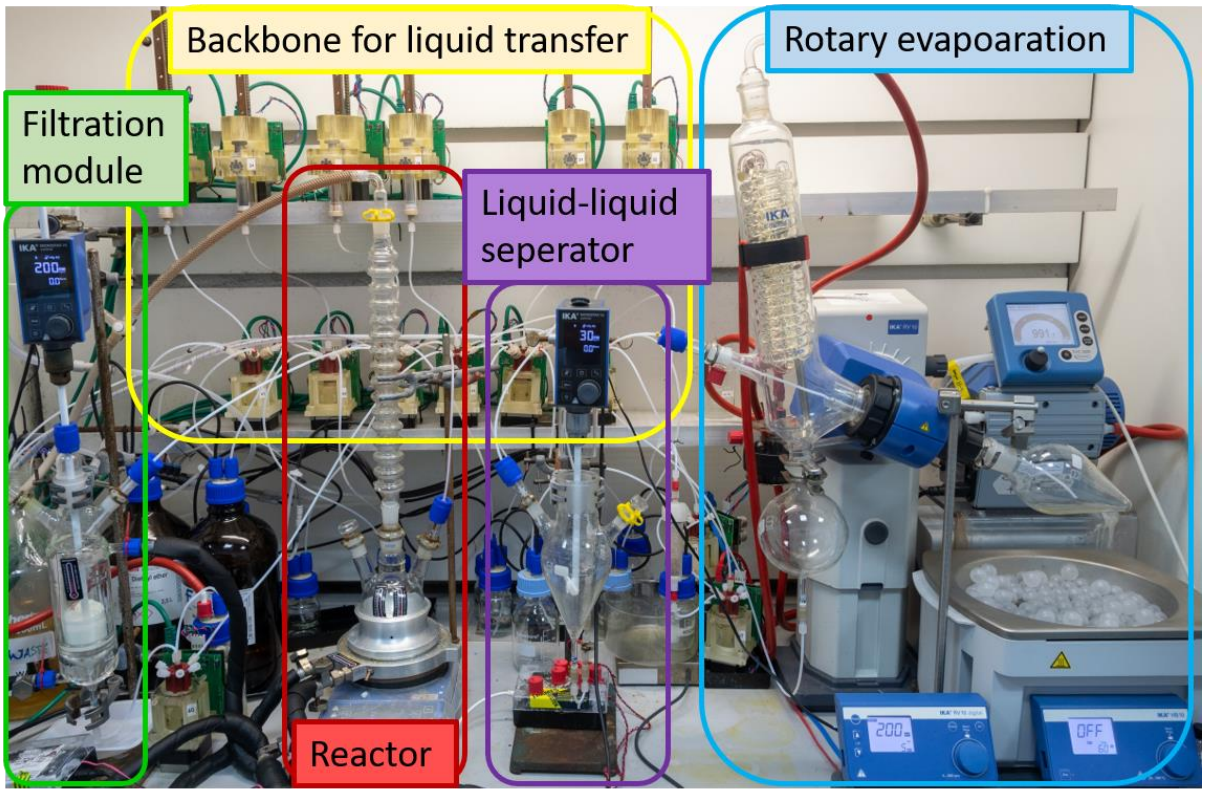


Figure 14: Chemputer rig build-up for the Sildenafil synthesis. Original picture taken by Jakob Wolf.

2 Experimental Approach, Outline and Aims

2.1 Platform concept

This chapter will outline the experimental approach of this work and present the hypothesis this work is based on. Further, this chapter will reason why recursive experimental systems are interesting to investigate, and discuss the chemical input sets that have been chosen for our automated platform runs.

2.1.1 Hypothesis

A paper presenting a theoretical concept to identify biosignatures published by Marshall in the Cronin group in 2017 inspired the initial idea of this project[93]. The idea is to use molecules/ structures and assign their "pathway complexity". This pathway complexity (or pathway assembly) measure calculates the complexity of a single molecule based on the number of steps that are required to join a complete structure together from unique fragments. This means that in general bigger structures have higher pathway complexity, but in the case of symmetrical / repeating structures the number of unique fragments is smaller so the pathway complexity number is lower again. This way to assign complexity for structures has boundaries, but it could be used to assign if something is dead or alive.

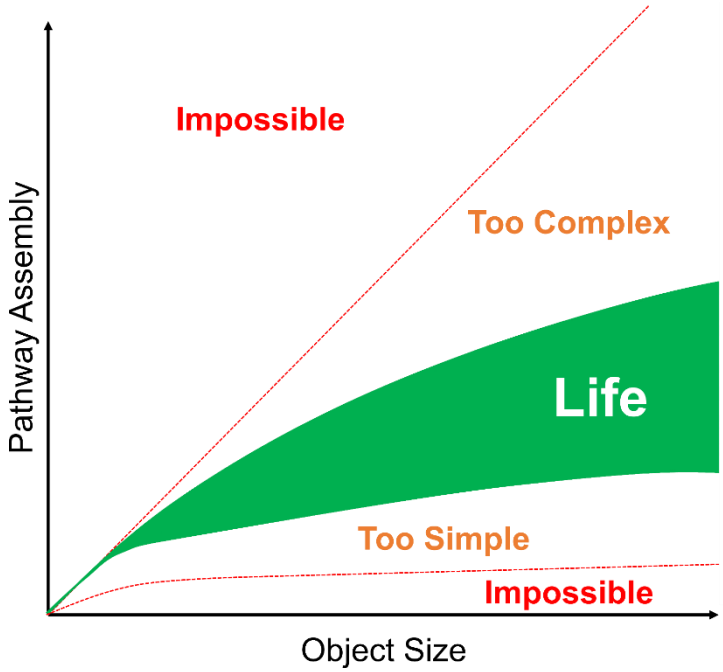


Figure 15: Pathway Assembly against object size. An illustrative graph reproduced from Marshall paper[93]

As shown in Figure 15, there is a big space that this the algorithm would not be able to assign (red area). In orange we see two areas of structures that are either too complex to be naturally produced in living systems (for example synthetic molecules like Cyclosulfamuron or Sildenafil), or that are too simple, meaning

they might be naturally occurring products (and may be found in living organisms) but that do not have the necessary structural complexity to be markers for life. The interesting zone of this graph is the green zone, where Marshall suggests that structures with this amount of complexity are neither too simple nor too complex and could therefore be alive or have been produced by something, which is or was alive. This means that if this theory can be tested, it could give scientists a handle (the range of Pathway Assembly value) when looking for life in reactions as well as on other planets, of which molecules can be used as markers. Furthermore, this theory applied to measuring instruments (for example a mass spectrometer) could enable the researcher to test if samples are alive through a simple measurement. The question of how to reach systems that can produce this kind of complexity from simple precursors is still unsolved.

Based on this idea and the desire to get further in finding an answer to this question, we present three key points/ assumptions or our theory on how to find complex structures in the lab:

1) Only living systems can produce molecules of complexity over a certain threshold.

2) It is possible to make complex chemical mixtures, networks and structures in messy “soup like” mixtures.

3) We are looking for phenomena rather than specific products.

Key point one is based on the Marshall paper, in which it is discussed extensively. Number 2 is more about the experimental side of the approach. We are suggesting that simple reactions of small molecule building blocks can produce complex chemical systems that can develop an effect of chemical evolution or build up to an autocatalytic system. The last key point is aiming to assign the complexity or messiness of these product mixtures. It suggests a more general way to look into a product mixture and is the contrary approach than to look for specific targeted products. This is important because work in this area of science often is stuck trying to identify molecules rather than looking at the phenomenon. Another problem with the general scientific method is that it is very slow and especially performing experiments in the lab and analysing them afterwards, takes a lot of

time. Therefore, it would be beneficial, to automate most of that process and let automation and computers work for us.

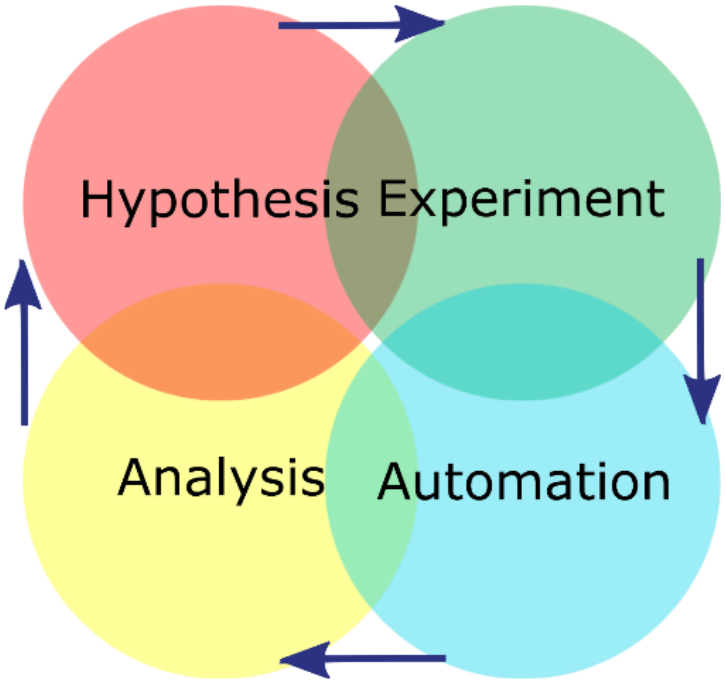


Figure 16: Experimental concept, the hypothesis leads to the experiment that is going to be automated, analysed and the results will lead back to an updated hypothesis

This work follows the "scientific method", accelerated through the use of automation (Figure 16), more suitable for modern day chemistry and allowing the performance of experiments 24/7. As well as performing many repeats or experimental systems with a cyclic approach. The approach to this work is the consideration of a bottom up approach to the question of how to make artificial life, but on the experimental scale, how to build a unconstrained complex chemical network from simple building blocks and how to assign the change of complexity or messiness in that mixture. For this question, an experiment was designed, which runs a system in a repeating cyclic manner.

So far, this is the same approach to any other scientific question elsewhere in chemistry. However, the next step is to develop an automated system to perform the planned experiments and to perform the analysis, meaning that it will run the actual analytic instrument as well as make sense of the collected data and decide something about necessary experimental changes. With this approach, the collection of vast amounts of data is possible and the hypothesis is updated more

often based on a much larger amount of collected data and knowledge gained from performed experiments.

2.1.2 Experiment

For these experiments, a library containing 18 previously chosen chemicals was created and they are reacted which each other in a random, recursive approach. Recursive reactions are different from conventional experiments, as they do not have a real finishing point as shown in Figure 17. In a conventional reaction, the starting material is added, stirred/ heated for a set amount of time until the end of the process is reached. At that point, the whole reaction is deemed “done” and the product is analysed and cleared out. Recursive reactions start in the same manner, starting material is added together and stirred/ heated for a set amount of time, but in contrast to a conventional reaction, a cycle time rather than a reaction time defines the experiment.

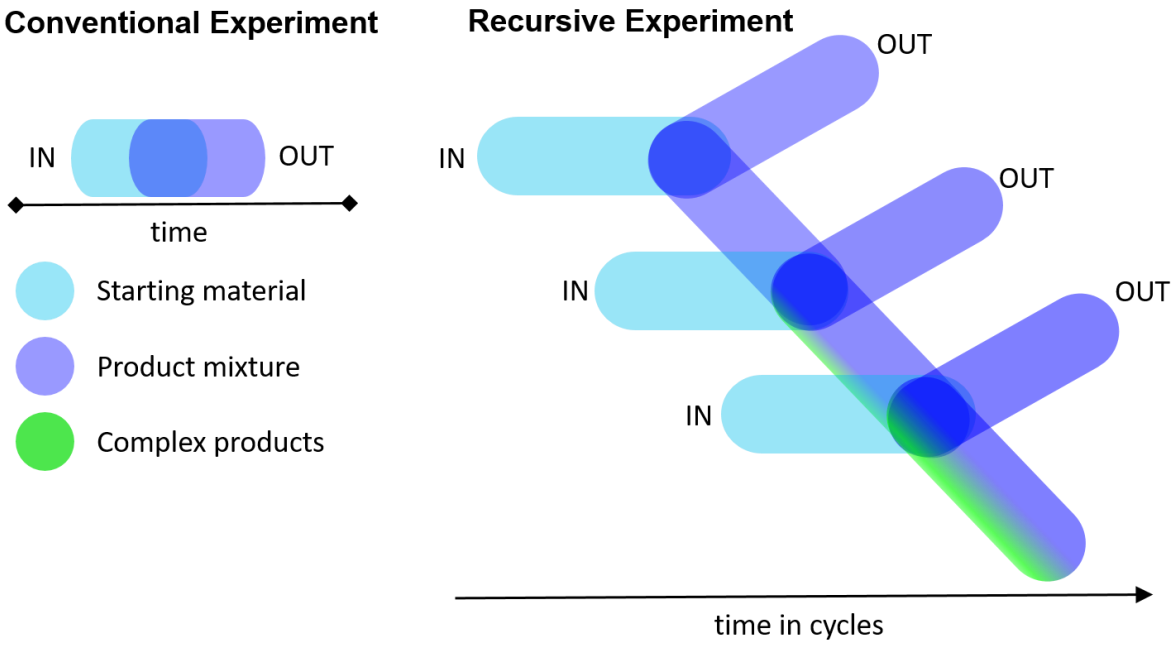


Figure 17: Schematic flow diagram to highlight the differences between conventional (left) and recursive (right) experimental systems. Over the course of the recursive experiment, a complex product mixture (green) is evolving.

After the cycle time, most of the product mixture is removed, but, importantly, a small amount of the mixture is also left in the reactor, which is replenished with

fresh starting material and subjected to the reaction conditions for another cycle time. Through this process, the chemical system is kept far from equilibrium, adding an effect of “chemical history” to the reaction. This approach can lead to a dramatic complexification of the product mixture in comparison to conventional experiments.

The effect can be amplified through the use of minerals, adding porous material with a high, functional surface area to the mixture and leaving it untouched in the reactor through each recursive reaction cycle. To tame a possible combinatorial explosion, the starting material is added under high dilution. As recursive reactions depend of the repeat of many reaction cycles with very exact reaction timings and the repeating withdrawal of product mixture and addition of starting material, automation is a very useful tool for this kind of reaction type.

2.1.3 Chemistry

The amount of different reactions and molecule combinations that could be screened is unlimited. Many known “prebiotically plausible” reactions and interesting chemical effects where considered. Reduced and oxidised sulphur compounds, monomers for polymerisation, including amino acids, interesting redox-active compounds, phosphorus- containing substances, acids, bases and various activating agents. The list of chemicals can be found in Table 1 and the choice of chemical inputs was approached in as open and “un-biased” way as possible, but there must be of course an awareness, that building a library of less than 20 substances is tiny and that every human decision is made with bias.

Table 1: Input solutions options given to the algorithm

Input	*chemical*	*Input*	*chemical*
1	Resorcinol	**10**	formamide
2	pyruvic acid	**11**	Catechol
3	acrylic acid	**12**	ruthenium-(III)-chloride hydrate
4	glycidol (2,3-epoxy-1-propanol)	**13**	Formaldehyde
5	glycerol	**14**	copper-(II)-sulphate pentahydrate
6	carbonyldimidazole	**15**	sulphuric acid
7	ethyl acetate	**16**	nitric acid
8	pyridine	**17**	ammonium thiosulfate
9	oxalic acid	**18**	potassium pyrophosphate

We decided to use resorcinol and catechol, both organic acids that can lead to polymerisation as well as pyruvic acid that could further lead to the build-up of interesting organic compounds. The alcohols glycidol and glycerol are both intermediates in carbohydrate and lipid metabolism and were added as well as acrylic acid that can further lead to polymerisation. As coupling agents and interesting ligands in several interesting organic reactions, adding carbonyldimidazole and pyridine seemed like a good choice. For further functional group diversity and the fact that these compounds are very basic organic building blocks, oxalic acid, ethyl acetate, formamide and formaldehyde were added. For catalytic effects, inorganic salts like ruthenium-(III)-chloride hydrate and copper-(II)-sulphate pentahydrate have been chosen. To complete the list we added acids, sulphuric acid and nitric acid, and basic molecules, ammonium thiosulfate and potassium pyrophosphate.

2.1.4 Workflow

As we were running several repeated recursive reactions requiring exact repeats and time points, building an automated system appeared useful. The workflow for this system can be found in Figure 18. Different small molecule inputs were chosen, all in high-diluted aqueous solutions that could be added to the mineral charged reactor that was stirred and heated under nitrogen with a set stirring and temperature. After every cycle, stirring and heating were stopped. The minerals in the reactor allowed to settle, and after a set amount of time, an automated sample was fed into the HPLC-MS system. Most of the remaining product mixture was taken from the reactor for further online analysis. Remaining in the reactor from there is a small amount or product mixture and the minerals. This remaining reaction mixture is then replenished with fresh starting material and the reaction, including stirring and heating, was started again in a new cycle. One cycle was repeated several times to add an effect of chemical history, expecting an effect of “chemical evolution”. This means the complexification of the reaction system over time, could lead for example to data, which shows a hill-climbing pattern. The reaction mixture was directly, inline analysed via HPLC-MS, to make sure no product breaks down while being stored or prepared for a different way to analyse data. The algorithm assigns the outcome of the reaction cycle in a fully automated way with almost immediate reaction feedback, making the input decision for the next cycle.

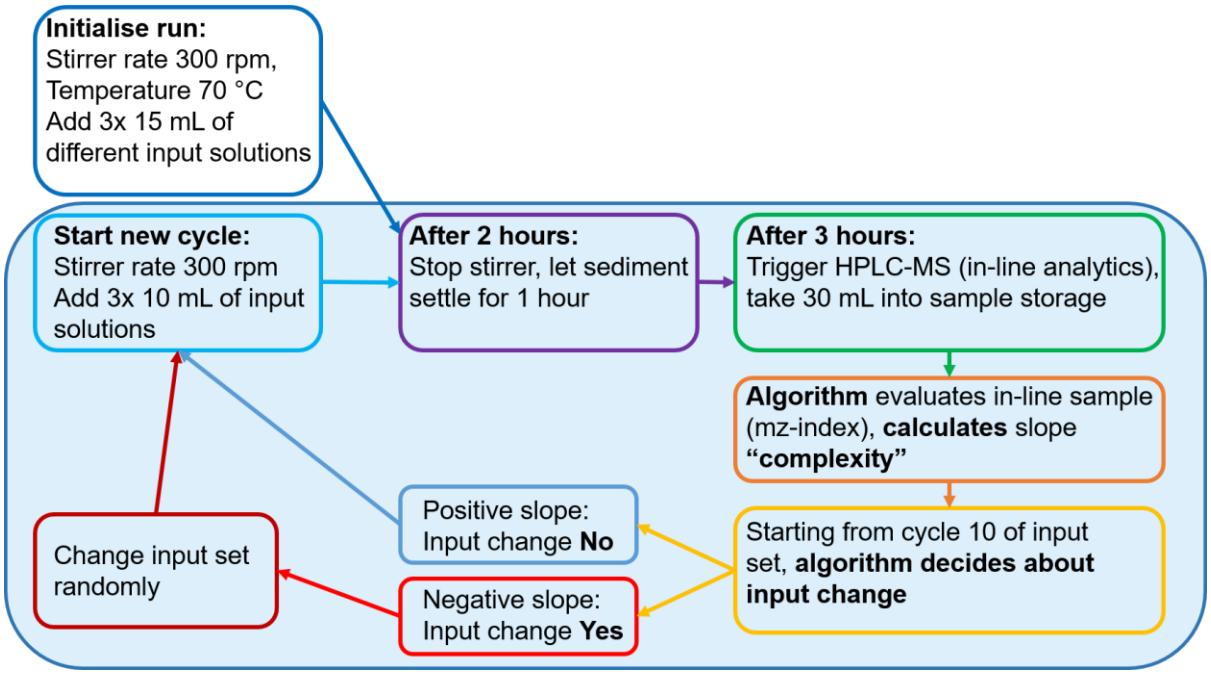


Figure 18: Suggested system workflow. The white box is the initial cycle while everything in the blue box is the workflow during the recursive cycles.

3 Results and Discussion

3.1 Build up

Developing an automated platform had various practical reasons. The recursive experiments require a lot of lab time and repeating a cycle around 100 times per reaction demands a high consistency in time. Inline analysis is incredibly useful observing products that are not necessary stable for a long period of time. In addition, automation can be helpful with reproducibility issues common in modern chemistry. The platform went through many different development stages during this project and this chapter is going to give a broad overview of the build-up process.

The first iteration of this platform was composed from various pieces of equipment that were already present in the lab. A 600 mL wide neck screw bottle was used as reactor, topped with a reflux condenser and kept under nitrogen gas controlled by a flow controller, model 0254 by Brooks Instrument, ensuring a controlled reaction atmosphere. This was heated by silicone heating mantles, tailor made heaters from the company BriskHeat, controlled via an Arduino, as well as an IKA RET control-visc hotplate with an USB interface.

For liquid movement, five Tricontinent C3000 syringe pumps with 3 port valves and 5 mL syringes connected via daisy-chain connection and RS232 cables, along with in-house build 6-way distribution valves that are controlled and powered via Ethernet connection have been used. Three of these pumps and valves have been designated for the input system, containing 3 x 5 reagent bottles along with one water bottle per valve. This allowed for a “library” of 15 reagent solutions in the system. Another pump and valve set was used as “offline sampling system” allowing the storage of up to six product mixture samples without the need of changing bottles over. The last tricont pump was used to inject sample solution from the reactor into a sample loop in an additional HPLC sample valve. All components of the platform have been controlled via Python code, based on code written by previous group members.

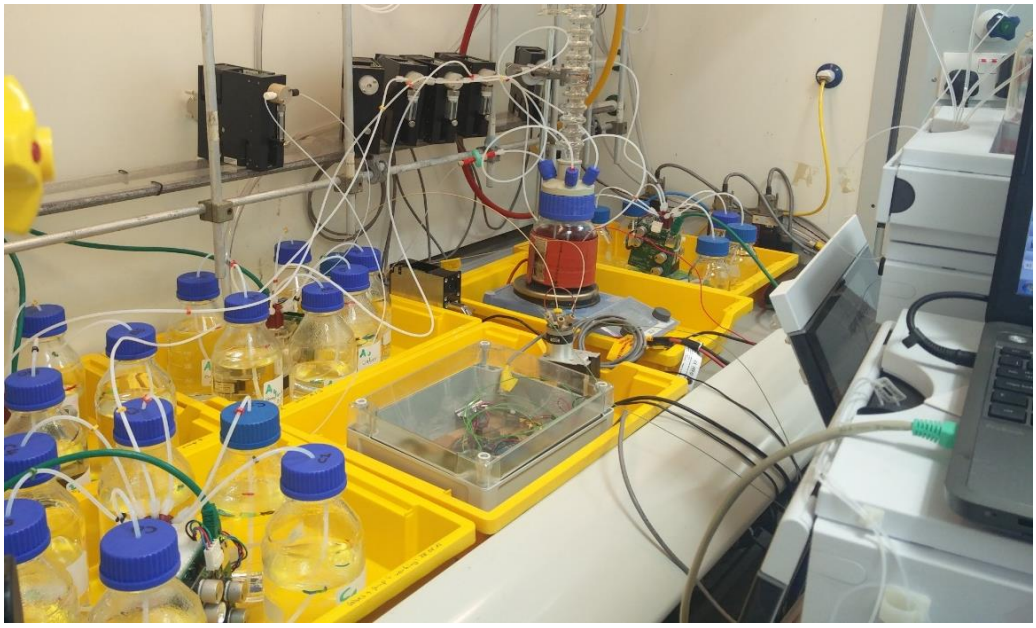


Figure 19: Picture of the platform in the early stages. Four in-house valves connected via Tricontinent pumps. The reactor surrounded by the heating mantle in red. On the right is a part of the inline analytical system (HPLC-DAD) visible

The execution of a number of runs with the set up in Figure 19 was attempted; however, the inline sampling system was not working at that time. All data collection was performed offline via HPLC-DAD or UPLC-CAD. The platform was put together in “quite a hurry” and many bits turned out as not fully functional. Two big pitfalls made running the system incredible difficult. The existing software was not proper working and the in-house valves turned out to confuse their position constantly.

For the next stage of the platform, the build-up was simplified and the code was completely rewritten by another group member, Graham Keenan. The new code is written in python, well commented, modular and can even be used by a chemist that does not own skills in programming. The in-house build valves have been changed to Tricontinent rotary 6-way distribution valves. Due to delivery times of the valves, there was some ”cheating” with the input solution system. The code got the option to choose of (at that time) all 15 reagent options of that reactions set, however the actual input system got only one valve and one pump, allowing for just six reagent solutions being present, these got changed manually depending on the input decision. With this, very much simplified set up, it was possible to perform a number of successful runs, yet still running the reaction recursively and storing the sample for offline analysis, not performing any kind of inline analysis.

After executing a fair amount of runs with the reduced system, the platform needed another upgrade to actually be able to complete the planned experiments. At that time there have been as well enough offline collected samples present to develop and calibrate decision making algorithms. More pumps and valves were delivered and the system was expanded to its previously intended size, see Figure 20. At this point, the input system consisted of three six-way distribution valves attached to a Tricontinent pump each. Every valve was connected to six reagent bottles containing an individual input each. The sample storage was expanded to a set of two six-way valves connected to a pump with a four-way distribution valve allowing connecting to both valves and expanding the storage system to 12 spaces. The only thing missing at this stage was a proper working inline analytical system.

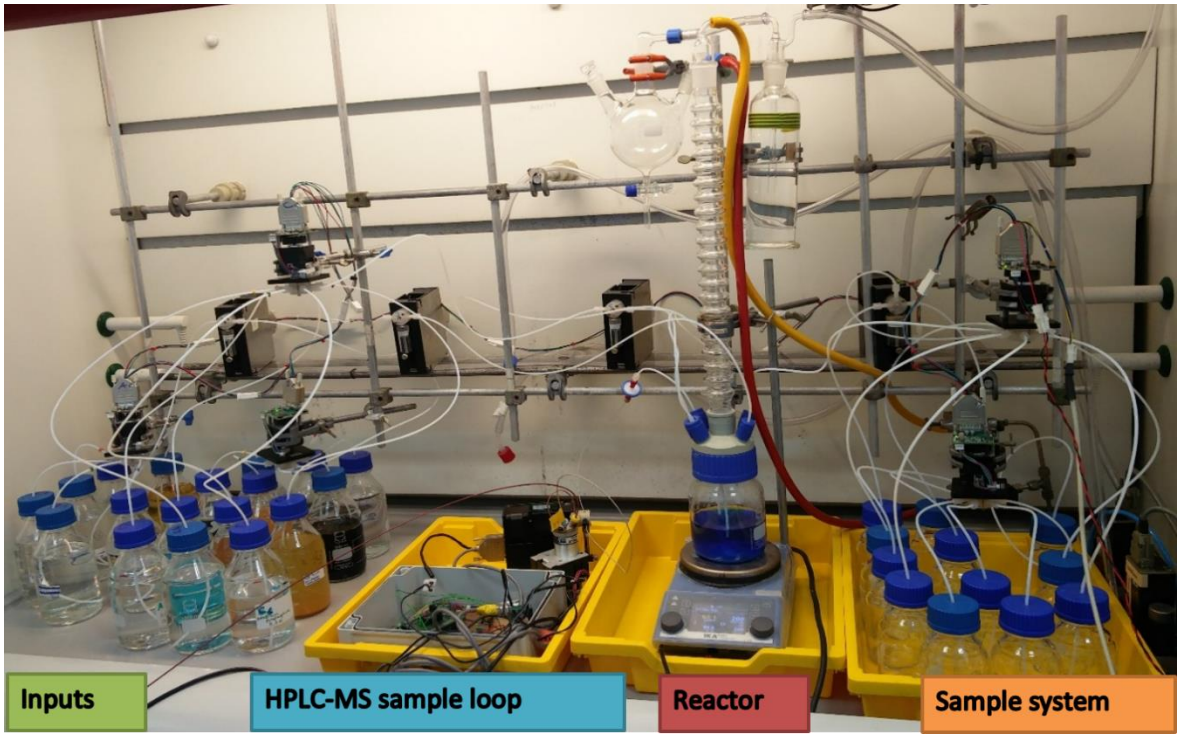


Figure 20: Platform in full build up stage.

The input bottles are on the left, connected to three 6-way distribution valves. On the shelf are four Tricontinent pumps of which the fourth is connected to the two 6-way distribution valves on the left building the offline sampling system

For the inline analytical system, which is shown in Figure 21, the HPLC-DAD system was coupled to an Advion L-series benchtop mass spec. To give us control over the complete sampling system of the HPLC, an external HPLC switching valve with a homemade sample loop was build. The Tricontinent pump was triggered by the

code, withdrawing a small amount of sample from the reaction vessel, through a 0.2 µm size nylon syringe filter. After each reaction, a cycle was completed and the sample was pushed slowly from the syringe into a piece of HPLC tubing, loading the 16 cm long sample loop, which was part of an external HPLC valve (no. 1 in Figure 21). In the default position of the HPLC, the instrument flow got flushed from the internal system through the external valve, through the column directly into the waste container. After the sample loop got loaded with fresh sample, the code triggered the system through an Arduino 2560/RAMPs combination. With triggering, the motor of the external sample valve switches and the sample is flushed together with mobile phase (as shown in Figure 21) from the external sample loop directly on the HPLC column followed by the DAD and the ESI-MS which is triggered through a contact closure from the HPLC system, ensuring no delay in the run time of both machines.

With this build up, the whole system was operational and the MS data driven algorithm was able to make input decisions leading the direction the each run was going.

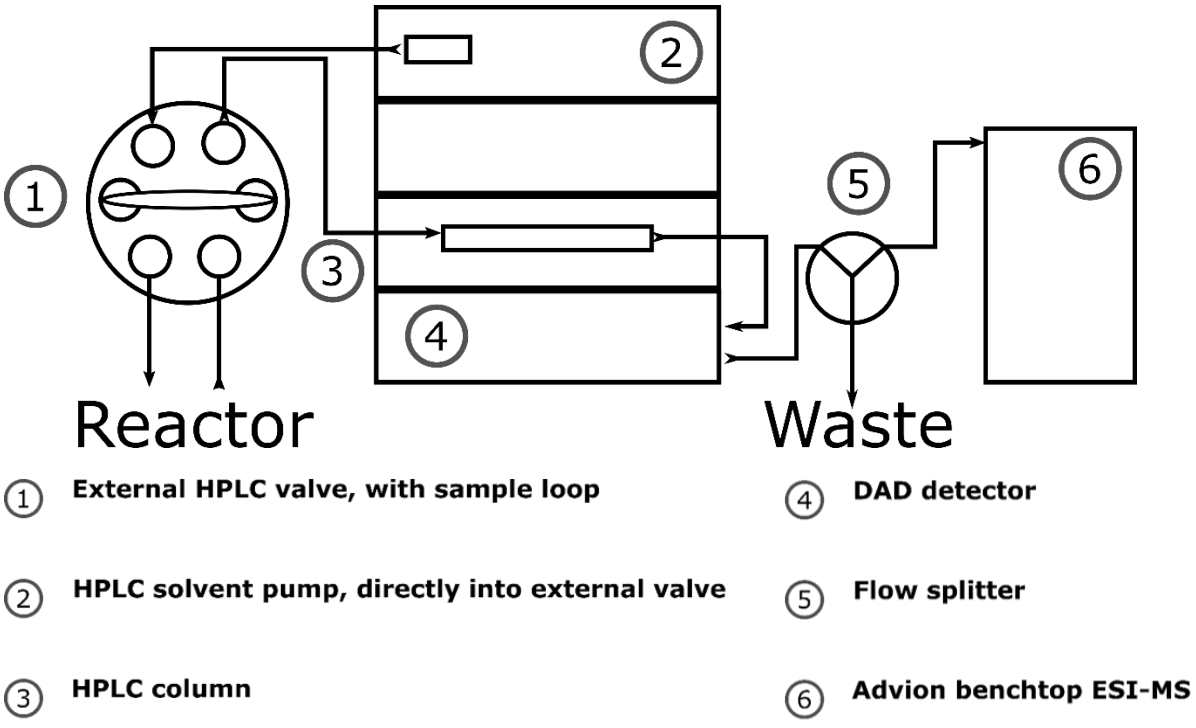


Figure 21: Build up of the inline analytical system. The external sample valve is on the left, the HPLC-DAD in the middle and the ESI-MS presented on the right

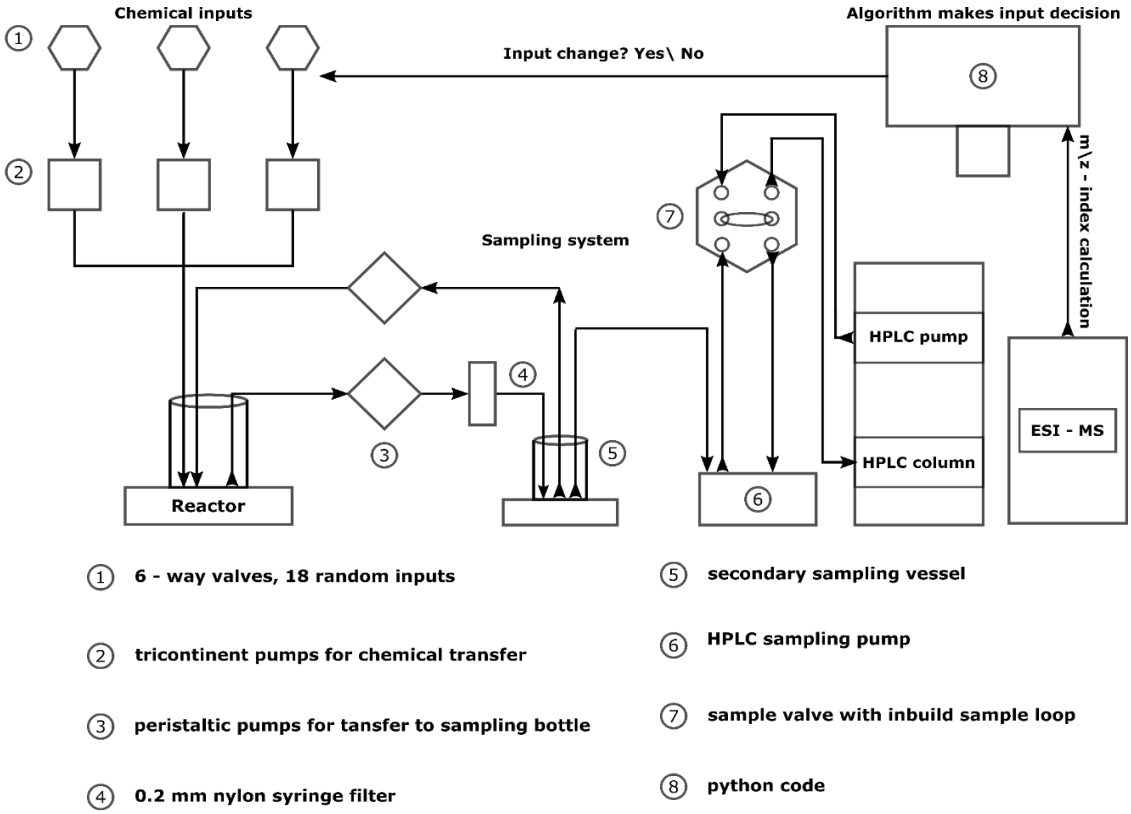


Figure 22: Platform with peristaltic pump exchange system and secondary sampling vessel

However there have still been some difficulties. The Tricontinent pump, usually known as very reliable was struggling to pull the sample from the reactor through a filter into the syringe and worse, could not handle the pressure that developed through pushing the liquid into the very narrow HPLC tubing. As a result of that, the syringe stalled and no sample was pushed into the line for measurement. A replacement was needed.

An old PerkinElmer 200 isocratic HPLC pump was unused in the lab. It was able to handle the pressure resulting from pushing liquid into a narrow peek tubing, the only issue was that it turned out to be not powerful enough to pull liquid from the reactor through a filter, into the pump. To solve this problem we separated the sample withdraw and filtering from the injection process.

An additional vessel was added to the system, as shown in Figure 22, used as a separate secondary sample vessel, with a continuous flow between the reactor and the vessel. This continuous flow was provided through two easy available and cheap peristaltic pumps model SR10/30 from Gardner Denver. The first pump pulled solution directly out of the reactor and pushing it through a syringe filter

into the secondary sample bottle. The second peristaltic pump returned with the same flow rate as the pulling pump, the sample back into the reactor. Through this method, we always had a liquid exchange and a filtered sample volume from the reactor. The PerkinElmer pump took the filtered liquid and pushed it through the sample loop. The performance of several runs on this system succeeded, though the peristaltic pumps suffered from a 24/7 use and the high amount of salty solutions made the HPLC pump struggle too. We needed a system that works 24/7 over weeks reliably. Aiming to perform long recursive runs and the least amount of human intervention.

A strong, chemically resistant peristaltic pump, made for pushing and pulling viscous and high-pressured materials seemed to be a solution, so a peristaltic pump from Vapourtec, model SF-10 was purchased. This pump was so powerful, that we abandoned the secondary sample vessel again and pulled the sample directly from the reactor through the filter, into the sample loop. We finally had a working inline sampling system.

With a working sampling system a fair amount of runs with different input composition have been executed. Nevertheless, there are always things that could get improvement. Based on our experience from the previous runs and build up stages, we went to a final effort to build the system we needed.

First of all, the 600 mL batch reactor and the sample size of 200 mL each cycle was a massive chemical waste as such an extensive amount of sample was not needed for offline analysis, not to mention, that moving that much sample liquid led to many issues with the sample pump through wear out and clogging. A 100 mL round bottom flask with a specifically tailored head was chosen as an alternative, allowing attaching screw fittings with the tubing connections and the condenser on top. The smaller sample size made the use of 250 mL glass bottles as sample container unnecessary and we changed to smaller, easier to store, 50 mL plastic centrifuge tubes for storage.

The sampling system had always had a big problem, cross contamination of sample. There was no way to clean tubes during a run and the solution was sampled into the same position through the same piece of tubing every twelfth

sample, there was always contamination present. To change this problem, a sampling system, where every sample has the exact same tubing path was needed. In that case, there would still be no mechanism to clean the sample tubing, but as the system runs recursively, traces of the previous run would be present in the current run anyway. To solve this problem an idea of a modular wheel from another group member, Daniel Salley was taken as an inspiration, and a new modular wheel based on our specific requirements was designed using Onshape (Figure 23).

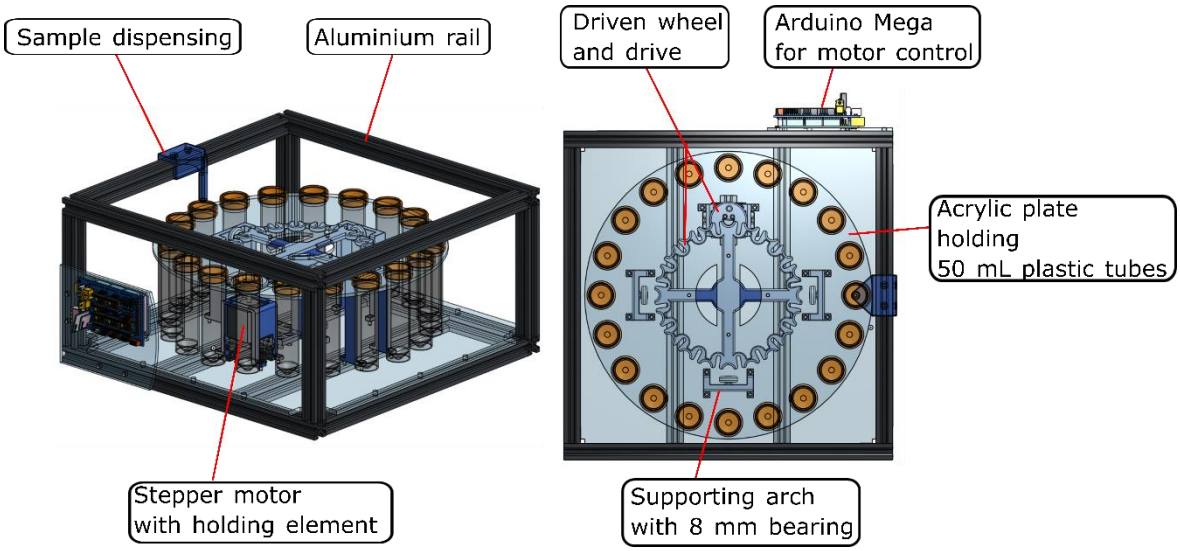


Figure 23: Design of Geneva wheel based sampling system. On the left is the system shown from the side and the picture on the right shows the wheel from the top

This system is based on a Geneva drive mechanism motion. Taking a rotation movement of a small wheel to increment the movement of a bigger wheel step by step with high accuracy. This mechanism enabled us, to move the opening of our sample tube very exact under the piece of tubing that draws the sample liquid from the syringe. The box in which the wheel system is contained is built from custom cut V-Slot aluminium rails. The column to hold the driven wheel, the drive wheel that increments the driven wheel, the stepper motor securing element, the supporting levelling arches and the solution dispensing part are all 3D printed on a Connex 500 with the translucent material RGD720. The base plate and the vial plate that is holding the 50 ml plastic sampling tubes is made from acrylic plates with a laser cutter. The levelling arches are all suited with an 8 mm ball bearing and the motor that increments the drive is controlled through an Arduino MEGA.

Every sample has always the same tubing path and it is possible to sample up to 20 samples continuously without the need to intervene by changing the sampling tubes. With this build up, as shown completely in Figure 24, several runs with different cycle length and chemical input sets have been run. It has proven to be fully reliable.

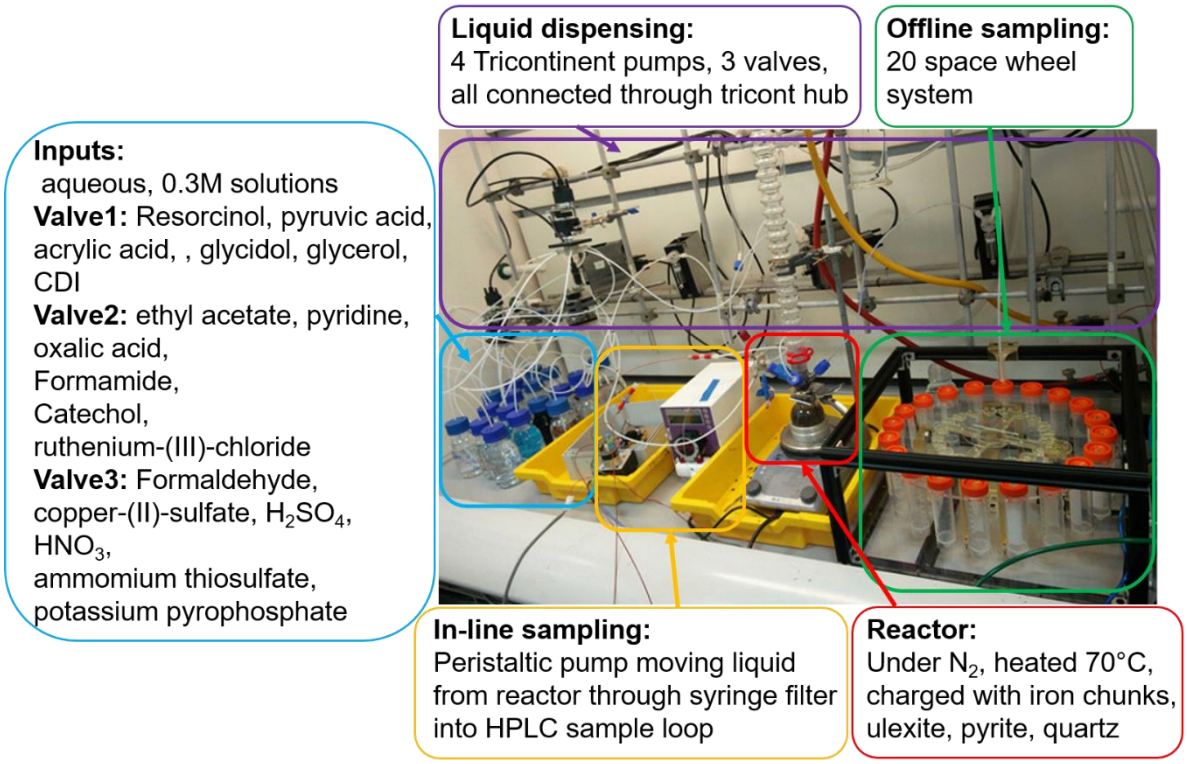


Figure 24: Platform in final build up stage

A fully automated, reliable hardware system was built. The build-up took a fair amount of effort and time and a lot of new expertise, probably unusual for the "normal" skill set of a chemist (pump maintenance/ cleaning/ repairing, 3d-design and printing, editing of code to run hardware, etc.) was acquired. In addition to that a way to build an inline system, which samples directly into a HPLC-DAD-ESI-MS system in a fully automated manner was developed, providing direct reaction feedback for the algorithm used.

The way to a fully automated, reliable system was very long and complicated. In retrospect, trying to use a barely running former long-term platform, and trying to make it work for the purpose of these experiments made the whole process of build-up longer and more complicated that it needed to be. It would have been more productive to design a system designated to do the required tasks from

scratch and build it from there, than to repurpose an old system that was developed for other objectives.

3.2 Algorithms and data

Using an automated platform to explore unconstrained abiotic mixtures brings up new technical challenges with respect to data management and analysis. With a new sample every three hours over the course of months, it is not possible to analyse every individual sample manually. Accordingly, there must be a workflow in place, which creates a balance between manually validating individual samples to ensure that the instruments are performing as expected and batch analysis of the data via algorithms. As already discussed in Chapter 2.1, the primary research objective is to handle the large volumes of data to look for system level phenomena instead of targeting specific products. This means, that cycles are compared with each other, looking for an increasing complexity, or change in general. As shown in Chapter 2.1, the inline analytical workflow is an High Performance Liquid Chromatography (HPLC) coupled to a benchtop mass spectrometer (MS). A few chosen cycles of one run, representative for all performed runs are shown in the following section, as presenting all data collected would extend the length of this work.

3.2.1 High pressure liquid chromatography

In the inline analytical system of the platform, (build up is described in Chapter 3.1) the sample is led through a home built HPLC sample loop until it is injected into a HPLC coupled to a diode array detector (DAD). This technique is useful, to get a first idea of the sample mixture and it does provide an easy means to check if the sampling system worked every cycle. The reliability of an HPLC-DAD and the ease of maintenance- compared to a mass spectrometer for example- makes it easy to use such a system in an inline analytical system. Especially when considered that the performance will not change between samples, if proper column care, such as thorough flashing of the column, the use of a guard column before the column receives the injected compounds as well as proper maintenance of the solvent lines, is performed. As the analytical method was not supposed to

narrow down the chemical input options for the system, a poroshell column was chosen, which has the advantage that it is through its EC-C18 silica bonded phase a column that can be used for a very wide product range. As we have a mixture of aqueous, organic, basic and acidic compounds as input options, it was a challenge to find a column, which would not break through the injection of a wide mixture of all sorts of chemicals.

Using a reversed phase column with a classic acetonitrile: water gradient method, which starts with a majority of the polar mobile phase, will lead to the elution of the polar products first. Followed by more non-polar species later, as the proportion of the organic mobile phase will increase the relative amount of the non-polar analytes partitioned into the mobile phase rather than interacting with the stationary phase, in this case the EC-C 18 deactivated silica surface inside the column. In Figure 24 the HPLC-DAD signal of Run G recorded at a wavelength of 215 nm, is shown. The cycles presented have been chosen as representative for each input set. It was always the first cycle of an input set and one further into the set of inputs chosen. Cycles with the same input set are presented in the same colour.

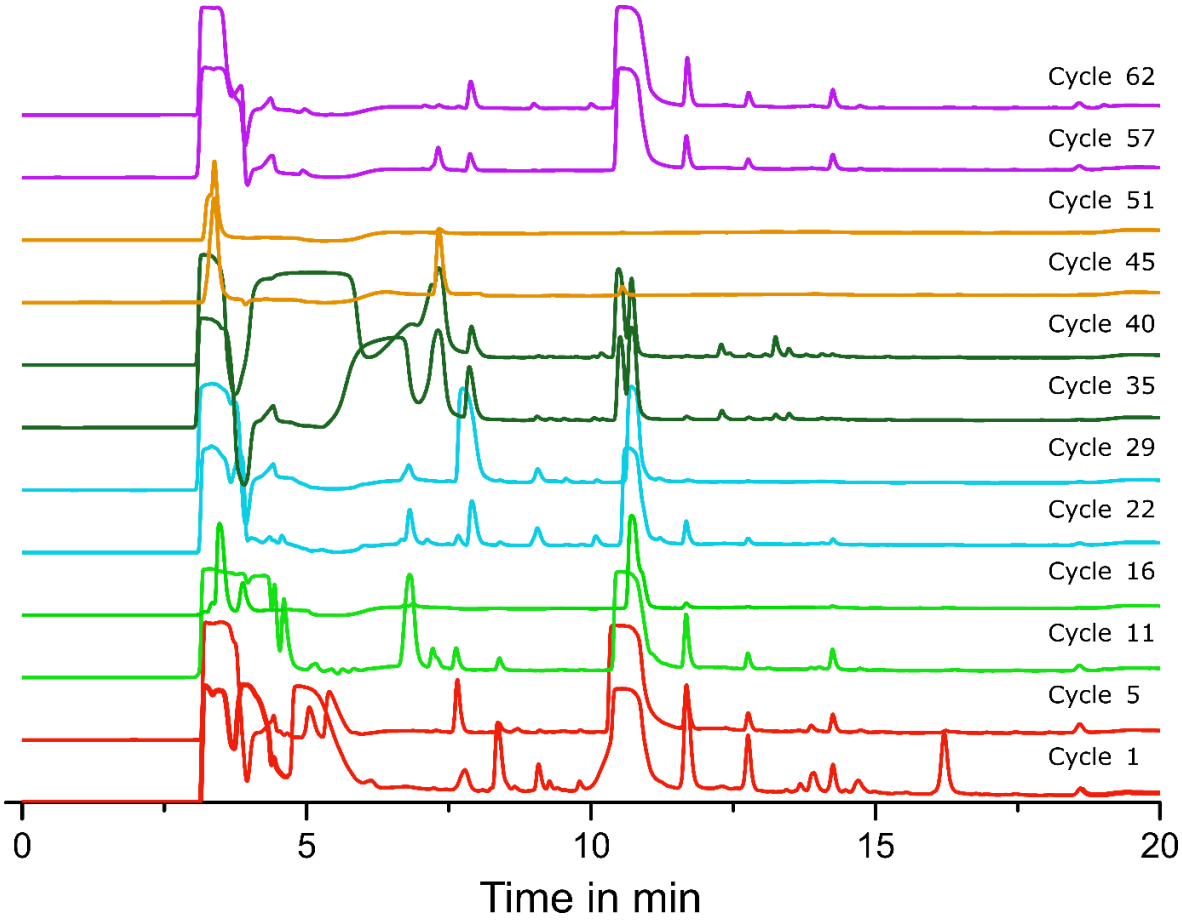


Figure 25: HPLC-DAD run overview of Run G. Presented wavelength is 215 nm. The line colour is showing how the different cycles are in context to each other. The cycles with the same line colour, had the identical input sets. The different input sets have been: Cycle 1 and 5 (oxalic acid, pyridine, catechol) (red), cycle 11 and 16 (catechol, nitric acid, pyruvic acid) (light green), cycle 22 and 29, (glycerol, ammonium thiosulfate, formamide) (blue), cycle 35 to 40 (glycidol, carbonyldimidazole, oxalic acid) (dark green), cycle 45 and 51 (ethyl acetate, pyruvic acid, ethyl acetate) (orange), cycle 57 and 62 (copper-(II)-sulfate pentahydrate, acrylic acid, glycidol) (purple)

In the presented run, the algorithm made all input decisions without human intervention. This leads to the fact that the number of cycles between different input sets varies as the input set is changed when the algorithm does not detect any change between cycles. The cycles shown in Figure 25 are the first cycles of one input set and a cycle in the middle of this set presented. The first random chosen input set shown in Figure 25 was oxalic acid, pyridine and catechol. The first cycle has more peaks than the cycle later in the input set, an explanation for that could be some mineral leaching ions or impurities, even if impurities have been eliminated through a mineral wash preparation previously to the start

of the run. In Figure 26, the HPLC-DAD run of cycle 1 and input solutions run individually is shown to represent how the comparison between the individual input solutions and the cycles has been done. The chromatograms of all input solutions are shown in Appendix Figure 1.

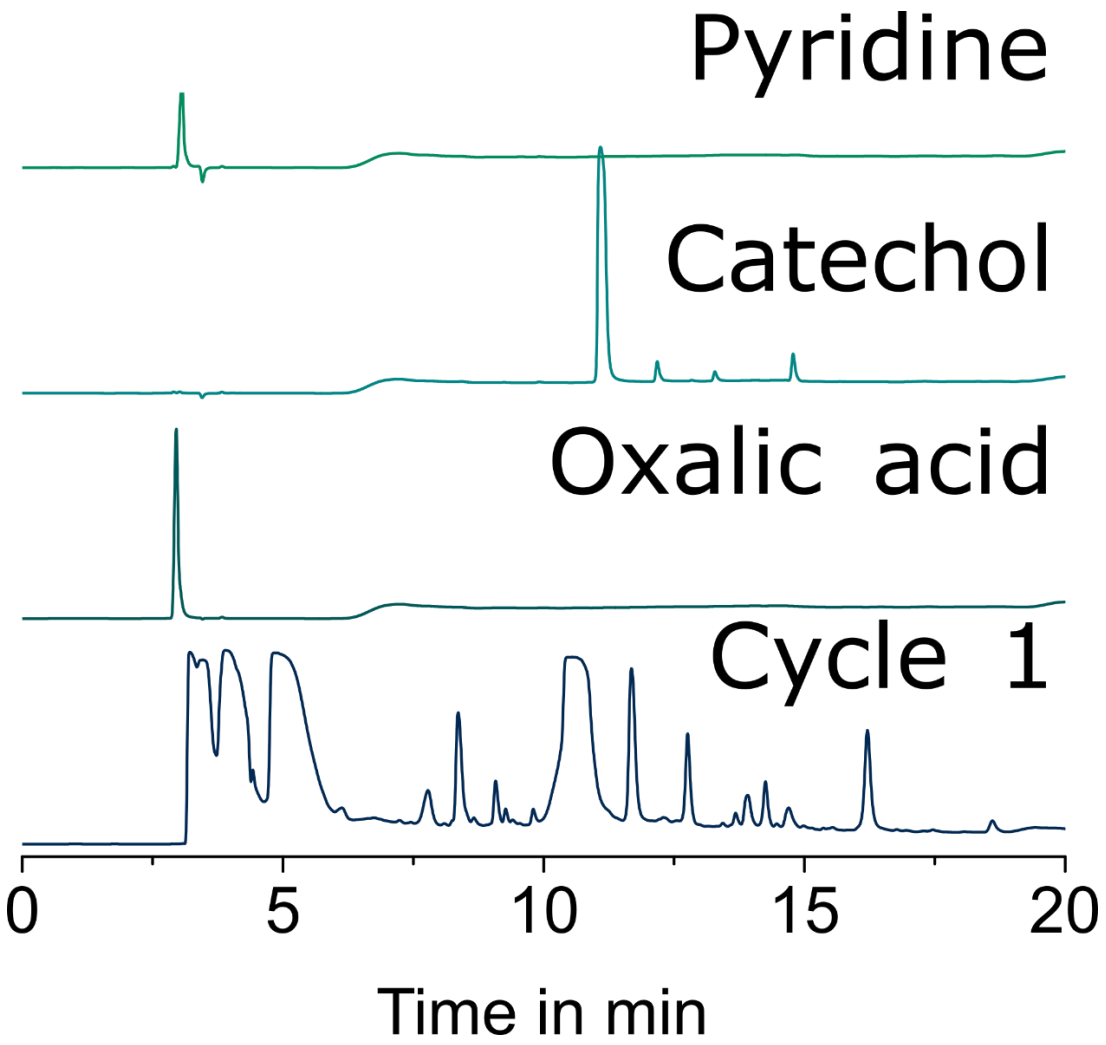


Figure 26: Run G cycle 1 compared to the input solutions used in this cycle (oxalic acid, catechol, and pyridine). All standards have been run individual through the same HPLC-DAD.

When the input solutions have been run on their own through the HPLC-DAD, oxalic acid and pyridine did elute at approximately three minutes Figure 26. This means, that these peaks will be hidden in the broad peak right in the beginning of the run. The catechol peak, which elutes at eleven minutes, can be observed in the chromatogram in Figure 25 . The next set of lines in this figure, which is shown in green, is the first cycle with the new input set composed of catechol, nitric acid and pyruvic acid. Here we have a similar situation, where nitric acid and pyruvic acid elute both at three minutes and catechol is visible at eleven minutes.

Cycle 16 consists of the same input set but the signal changed and there are less peaks visible. The three input peaks are still visible. The blue input set of Figure 25 starting on cycle 22 contains glycerol, ammonium thiosulfate and formamide. All three compounds do not have a distinct peak if run as a standard solution on its own but the peaks we can observe can come from products of this or previous cycles. Interesting is especially one peak at around seven minutes which occurred in the very first cycle but appears to have increased intensity over the cycles of the new input set. An explanation for this observation could be on the analytical side as some carry over due to not complete flushing of the column could have occurred, but based on our analytical method this option is very unlikely, it is more plausible that it is a species building up throughout the input set cycles. At cycle 35 the algorithm made another input decision and picked acrylic acid, potassium pyrophosphate and glycidol. Acrylic acid elutes at 7 minutes and we see a peak appearing in this area of Figure 25, as well as other product peaks not just in the first sample of that input set but additional in the chromatogram of cycle 51 (which is further into that set of inputs). The orange line of Figure 25 represents the chromatogram of cycle 45, which is the first sample in which the algorithm choose copper-(II)-sulphate pentahydrate, sulphuric acid and ammonium thiosulfate. There is a peak at approximately three minutes, which refers to the copper salt solution or the ammonium thiosulfate, based on the separately measured input solutions, both of which show a peak at this time. The peak at seven minutes appears to be acrylic acid from the previous set of inputs and disappears throughout the input set, we cannot observe this peak further into it at cycle 51, which is shown in yellow. The last set of inputs is glycidol, pyridine and pyruvic acid and we see a big broad peak in the beginning of the chromatogram reappearing, which is no surprise as both pyridine and pyruvic acid elute at that time, as well as a range of other previously observed peaks.

The HPLC technique that was used in this set up might not be the ideal one for the input solutions and products we want to observe, but it is the best method that was possible to automate and apply to the platform build up. A high number of the input solutions elute in the first five minutes and are packed together in one large broadened peak. If we would use exclusively HPLC-DAD as analytical method to assign our products, this would not be acceptable. However, this LC technique was not to resolve every single product in a baseline resolved peak it

was more a way to check if something changed in the reaction and to separate our product mixture prior to sending it to MS. A direct injection would lead to ion suppression issues, meaning the product signal would interfere or be suppressed by salt high amounts of salt present in the product mixture.

Comparing HPLC-DAD data and MS data is difficult as these techniques depend on completely different principles. In the DAD, every compound that is UV active at the chosen wavelength will be detected but species that are not UV active would not be observed. On the same page, with the method in use, it is not possible to even resolve the starting material signals completely from each other, based on this fact the HPLC-DAD data would not provide sufficient information about the product mixture on its own. In mass spectrometry, it depends how well the mixture ionises and if the species has a high ionisation efficiency within the product mixture. This fact was kept in mind when moving on, trying to find overlaps or differences between similar samples analysed with different techniques.

3.2.2 Electrospray mass spectrometry

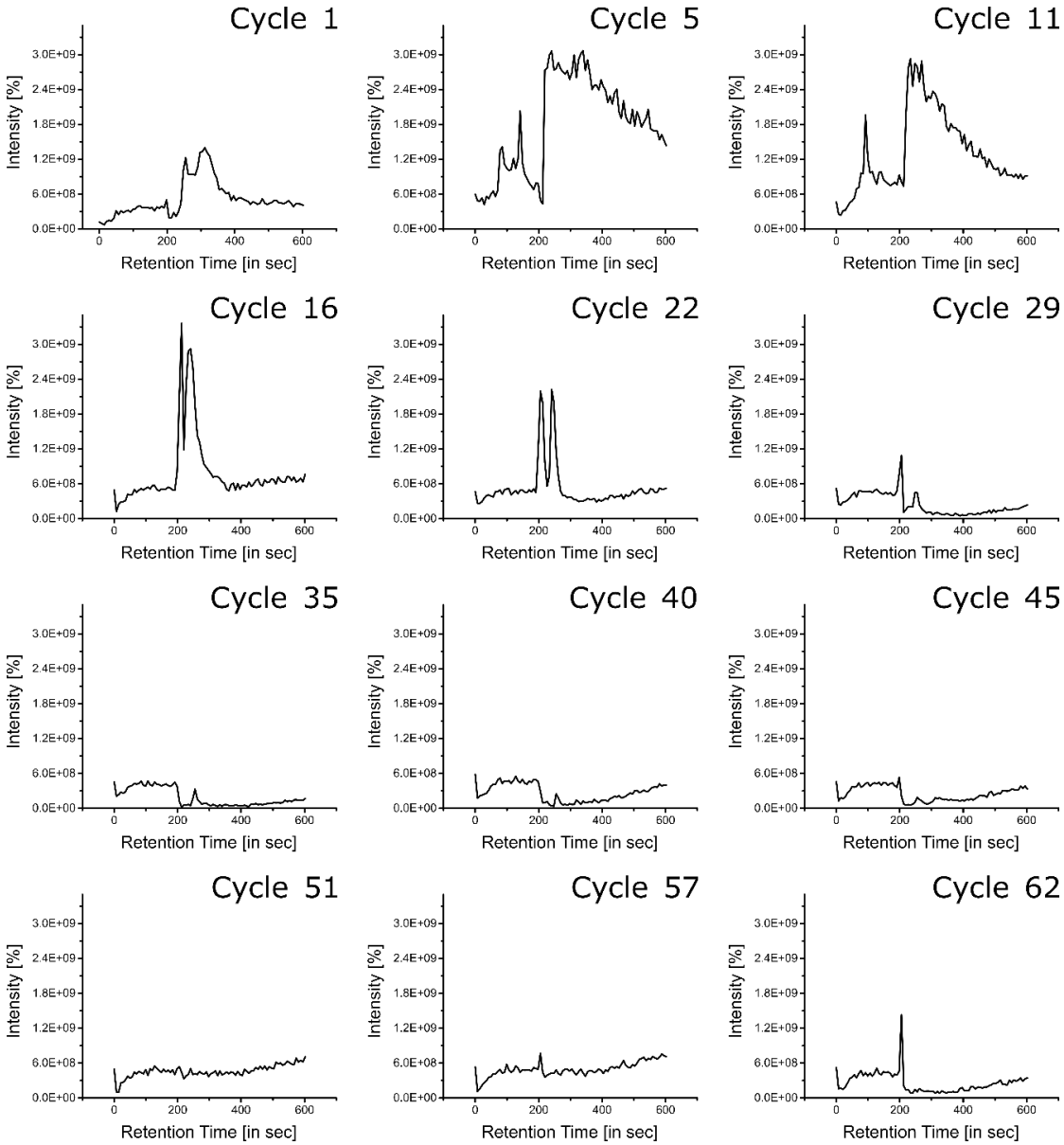


Figure 27: Total ion chromatograms of run G. The plot shows the intensity of the total signal over the retention time in seconds. The presented cycles have been chosen based on their input solutions as previously shown at Figure 25.

Having gained an overview of what (UV-active) species are in our sample, the product mixture was injected from our diode array detector into the electrospray ionisation mass spectrometer (ESI-MS). A comparison of the total ion chromatograms is shown in Figure 27. The TIC alone cannot deliver much information beyond demonstrating how much of the sample mixture is ionised by the spectrometer, but the retention times where a peak is observable can lead to interesting spectra. Spectra show the intensity against the mass over charge ratio of a point in a TIC and if desired, it is possible to extract the desired area of a

peak in the TIC into one spectrum, showing peaks of species recorded at that retention time. Spectra have been extracted for the most intense peak of each TIC and the spectra are shown in the Appendix, Appendix Figure 5, Appendix Figure 6 and Appendix Figure 7. A spectrum shows the intensity over the mass over charge ratio (m/z) of the extracted area. All peaks of the run G spectra extracted, which are shown in the Appendix that have a higher abundance than 10 % of the total abundance are listed in the following Table 2.

Table 2: Total abundance values of run G listed from extracted Advion MS spectra. The mass over charge ratios shown are based on the peaks observed.

	Cycles number											
Mass over charge ratio	**1**	**5**	**11**	**16**	**22**	**29**	**35**	**40**	**45**	**51**	**57**	**62**
39 m/z								15%	80%			
46 m/z								20%				
80 m/z								80%				
87 m/z						20%		15%				
88 m/z									20%			
106 m/z								20%				
121 m/z								15%				
138 m/z						20%						
143 m/z				45%								
145 m/z				40%					35%		20%	25%
147 m/z				20%					20%		10%	10%
151 m/z				20%								20%
155 m/z				30%	20%	10%						20%
171 m/z												
183 m/z				20%	15%							
188 m/z	80%											
196 m/z				45%	50%	20%					15%	60%
212 m/z						10%						
222 m/z						15%						
224 m/z				85%	85%	85%					25%	85%
292 m/z						10%			20%			
358 m/z						10%						
373 m/z						10%						
382 m/z						10%						
387 m/z						10%						

When the spectrum is extracted, as explained before, it relates to a range of interesting peaks. Each peak was checked if there would be a match with the mass of the starting material. If there was no match, it could be, if smaller, an ionisation fragment of the starting material and if bigger either a product species or a contaminate. For further product identification, a database search would be needed. Based on these conditions, we rather aim for a comparison of cycles looking for peaks that occur in multiple cycles. Cycle one, shows a peak in the spectra with a mass of 188 *m*/*z* visible, which could relate to a product species, as it does not match with our starting material. When looking into the same retention time in cycle five the peak at 188 *m*/*z* cannot be found again. At cycle 16 a range of different peaks can be observed as shown in Table 2. At 155 *m*/*z* there is a peak which shows a relative abundance of 30 % and shows up in the following shown cycles 22 and 29. Another peak at 183 *m*/*z* is shown in cycle 16 and 22 with decreasing intensity. This could suggest that this peak relates to a species which was made in around cycle 16 and decreased in concentration over time. A peak at 196 *m*/*z* shows an interesting development. Showing up first at cycle 16 and increasing the abundance at cycle 22 while decreasing at cycle 29. Then it is to be observed again in cycle 57 with an abundance of 15 % until it increases to 60 % in cycle 62. Both input sets of these cycles have input sets containing catechol, which leads to the suggestion that this peak relates either to the compound itself or to a product, which is made through the appearance of catechol. The same suggestion can be made about another peak observed in the spectra too. This peak is at 224 *m*/*z* and shows an relative abundance of 85% from cycle 16 to cycle 29. And similar to the peak at 196 *m*/*z* this peak at 224 *m*/*z* appears again at cycle 57 with an abundance of 25%, increasing to an abundance of 85% in cycle 62. Additional to the peaks already mentioned, cycle 29 shows a range of small peaks with a relative abundance of 10% in the mass area between 224 *m*/*z* up to 387 *m*/*z* as shown in Table 2. Cycle 40 and 45 show a range of peaks with lower masses from 39 to 121 *m*/*z* which could be ionisation fragments of the starting material or products. As stated before all observed peaks would need to be searched in additional databases or need further comparison to standard solutions of possible products to make further product assignments.

Overall when looking at Figure 27 it appears that the sensitivity of the instrument is lowered throughout the run. This is caused by the fact that sample compounds that do not ionise well in the mass spectrometer are covering the source of the instrument. Run G was run with three-hour cycles and this time was not sufficient to perform a thoroughly cleaning procedure in between cycles. For future runs, a longer cycle time or a different cleaning procedure, for example where two identical sources are going to be switched with each other could be a solution for this problem. Compared with the HPLC-DAD data, the MS data appears much noisier based on the physical properties of both techniques. MS can detect more compounds as it can measure everything that ionises, while the DAD is only limited to the wavelengths observed in the experiment (200, 215, 245, 300 nm) and if the compounds are UV active. As less resolved product peaks can be observed in the TIC of the MS and the baseline seems to vary a lot, this might be one of the reasons that not more peaks are visible in the TIC in Figure 27.

Looking into mass spectrometry with an untargeted approach is more complicated than to look at a spectra of nicely resolved compounds, but with the approach of looking into a product mixture from a system level, using algorithms and comparing data of different cycles over a run can help to fingerprint the product mixture. For a more thorough investigation of the product mixture of each specific run, a mass spectrometer of higher mass accuracy and sensitivity is needed. In Figure 28 the base peak chromatograms of run G measured on a more sensitive mass spectrometer, a Bruker MAXIS, are presented.

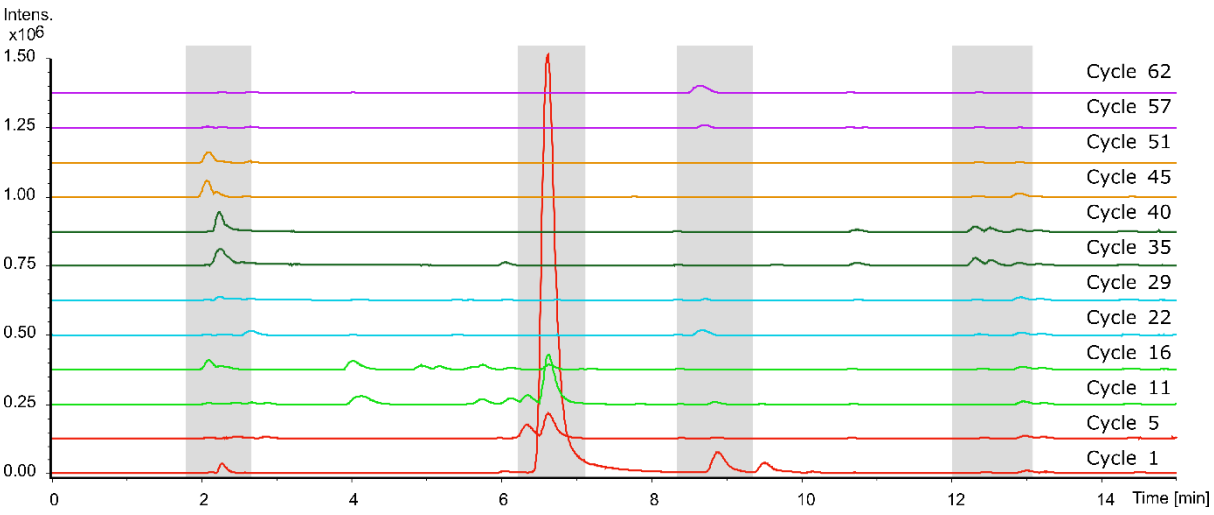


Figure 28: BPC chromatograms of offline samples measured on Bruker mass spec compared to each other.

A base peak chromatogram (BPC) presents the signal of the most intense mass in the total ion chromatogram (TIC), which usually leads to a more noise free chromatogram, as it is the case in the presented data in Figure 28. Many small peaks can be observed in the Bruker BPC. Four retention times are highlighted in grey that are further discussed. The first retention time is just after two minutes and has been already observed in the chromatogram of the Advion data in Figure 27.

For comparison, all input solutions have been run individually as standard solutions. The BPCs of the standards have been compared with the BPCs of the reaction cycles, which are our product mixtures consisting of three input solutions reacted together. Additionally to the extracted ion chromatograms (EIC) with the mass of hydrogen, hydroxide, sodium ions or water removed or added depending on the structure of the input compound have been generated and used for comparison too. The spectra of cycle 1 compared to cycle 35 for the retention time range from 2.2 to 2.5 minutes is shown in Figure 29.

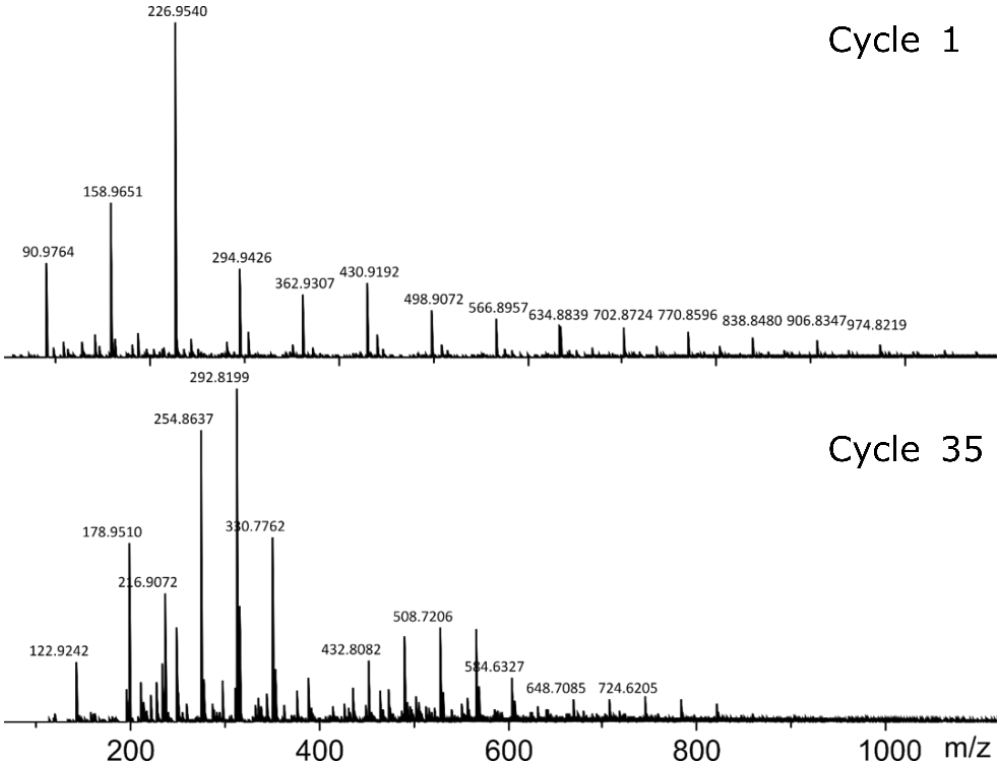


Figure 29: Spectra of Bruker chromatogram, cycle 1 and cycle 35 compared in the range from 2.2 to 2.5 minutes

Both spectra have a range of different peaks but in this case, none of them is overlapping. This is not too surprising as the cycles are quite far away from each other but it means that there is no visible trace of the first cycle in a later one. The peaks of these spectra are in the mass range from 90.97 to 974.82 *m/z*.The first peak at 90.97 *m/z* refers to the oxalic acid of that input set, the higher numbered species might be polymers or products that clustered together.

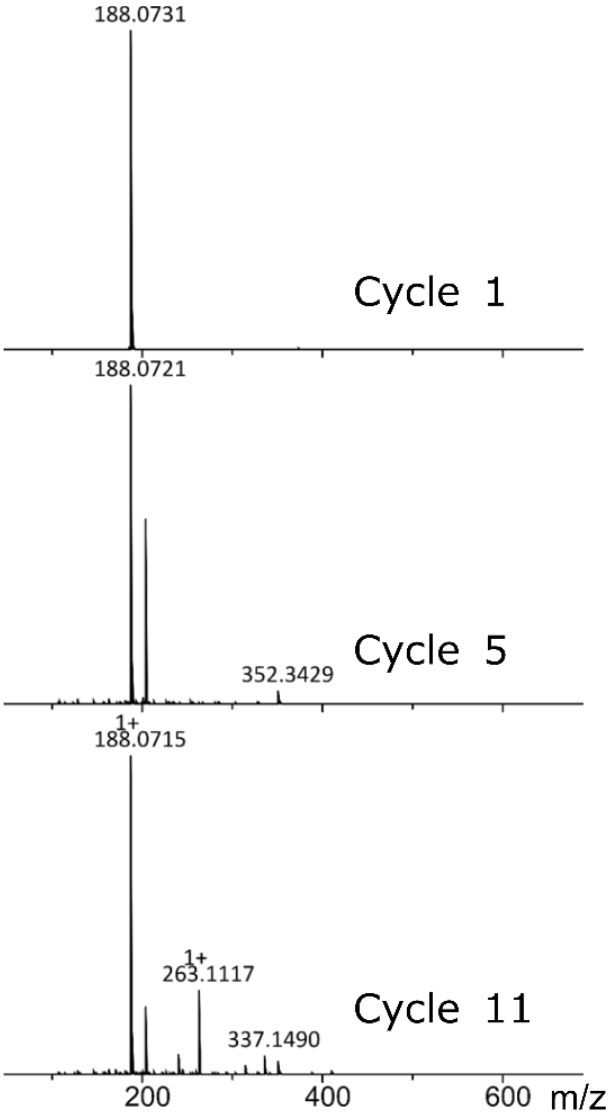


Figure 30: Spectra of chromatogram range from 6.5 to 7.2 minutes

The next peak in the chromatogram that is very prominent, especially in cycle one, as it appears to be very intense compared to all other peaks in every shown chromatogram, is in the retention time range of 6.5 to 7.2 minutes. A part of the correlating spectra of cycle 1, 5 and 11 is shown in Figure 30. There is one peak, which is at 188.07 *m/z* and visible in all three spectra. This mass does not match with any tested starting material and it is not possible to completely determine what molecule refers to this mass. This is based on the fact, that the starting materials have no obvious reactions with each other under these given conditions, but the metal ions from the minerals, for example iron from pyrite, could catalyse a wide range of different reactions. When further checked, the observed mass of 188.0731 *m/z* would refer to a species with the formula $C_{11}H_{10}NO_2$ and could lead to the hypothesis, that a pyridine molecule reacted with a catechol molecule in a condensation reaction. This reaction appears structurally and energetically very unlikely but the mass spectrometry data confirmes the calculated formula. We observe a base peak with the mass of 188.0731 *m/z* and based on the suggested formula the

theoretical mass would be 188.0716 *m/z*. This means the error is 10.315 *ppm*. Further to this, when zoomed into the spectra, another peak of an abundance of 12.5 % can be observed (Figure 31: Spectra of chromatogram range from 6.5 to 7.2 minutes of cycle 1, zoomed in the *m/z* area between 187 *m/z* to 190 *m/z*).

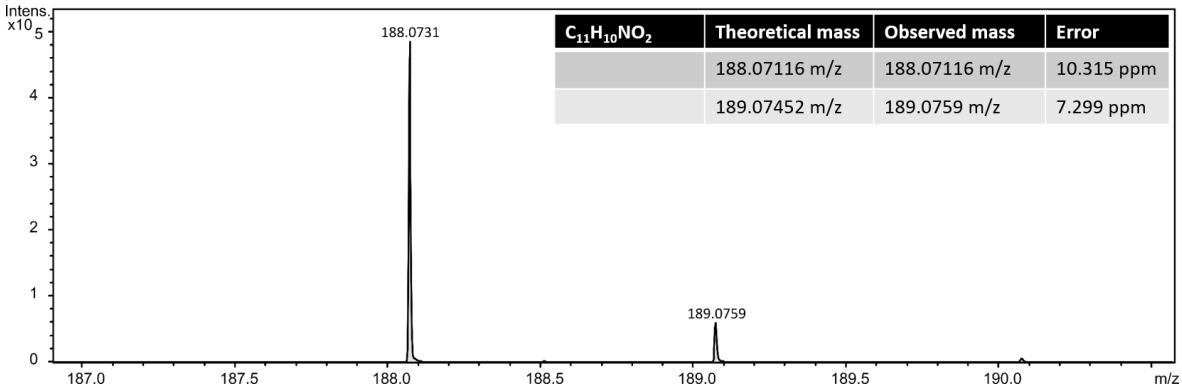


Figure 31: Spectra of chromatogram range from 6.5 to 7.2 minutes of cycle 1, zoomed in the *m/z* area between 187 *m/z* to 190 *m/z*

This peak at 189.0759 *m/z* supports the suggested formula, matching the second theoretical most abundant value, with a theoretical abundance of 11.9 % and a value of 189.07452 *m/z,* resulting in an error of 7.299 *ppm*. Both errors are low, suggesting that the data is good for the instrument used. Nevertheless without more data on this molecule or for example MS2 data, we can not give more information on this peak than the suggested formula. Interestingly when looking into futher cycles, there is a species development or a breakdown of this molecule visible. As in cycle 1 there is just one peak visible in the shown mass range, while in cycle 5 there are 2 other peaks appearing, one very close to the first one at 204.06 *m/z* and one peak at 352.34 *m/z*. Cycle 11 shows more peaks but the 188.07 *m/z* peak is still standing out and the small peak at 204.06 *m/z* from cycle 5 is present too. Two new peaks are shown at 263.11 *m/z* and 337.15 *m/z*. Neither of the mentioned peaks are matching the mass of any of the starting materials what leads to the conclusion that these peaks in Figure 30 relate to products.

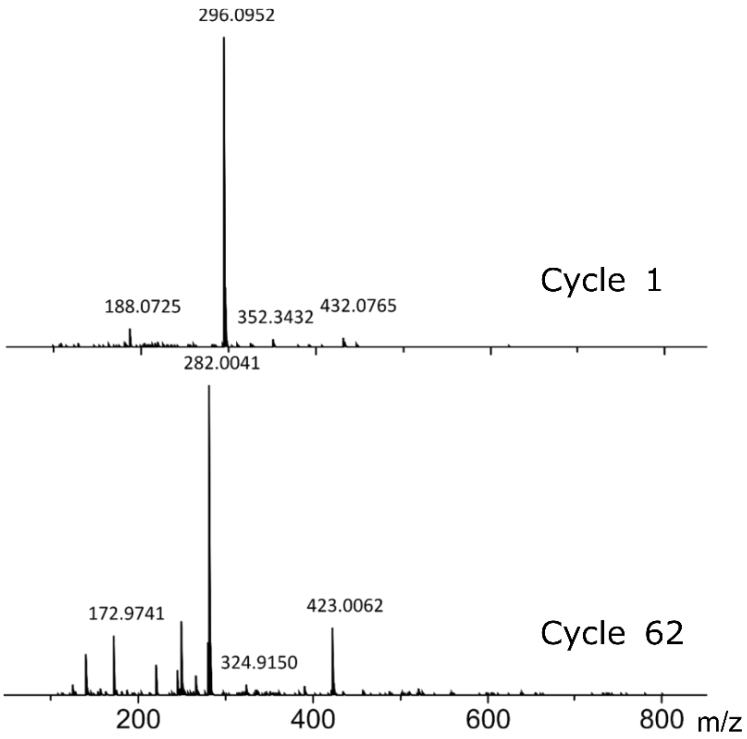


Figure 32: Spectra of Bruker chromatogram, range from 8.5 to 8.9 minutes

The next peak in the chromatogram of Figure 28 is between 8.5 to 8.9 minutes and the spectra is shown in Figure 32. This peak is interesting as it is just visible in the chromatogram of cycle 1, 22, 57 and 82.This leads to the assumption that it must be related to the presence of catechol as this was the compound that all input sets of the names cycles have in common. In cycle one we got four different peaks, we see the 188.07 *m/z* species that we already observed on a different retention time point in the spectra in Figure 30, another peak and the most outstanding peak is at 296.09 *m/z* and two other peaks at 352.34 and 432.07 *m/z*. At cycle 62, there are 4 peaks prominent, the first at 172.97 *m/z*, one at 282.0 *m/z* the next one at 324.91 *m/z* and the last one at 423.0 *m/z* but none of them is matching the peaks in the first cycle. As there are all in the same mass range, the peaks might relate to species originating from the catechol input solution.

The last peak that is highlighted in grey in Figure 28 is between 12.2 and 12.8 minutes and the spectra of cycle 5 and cycle 57 are shown in Figure 33. There is a range of peaks shown but there are two peaks in both chromatograms of specific interest as both of them are overlapping. There is one peak at 107.96 *m/z* in cycle 5 which can be found again in cycle 57 with a higher overall intensity. The second matching peak is a species at 520.26 *m/z* which decreses in intensity throughout the run. None of the peaks in the speactra relate to input solutions leading to the assumption that these are product species or contaminants.

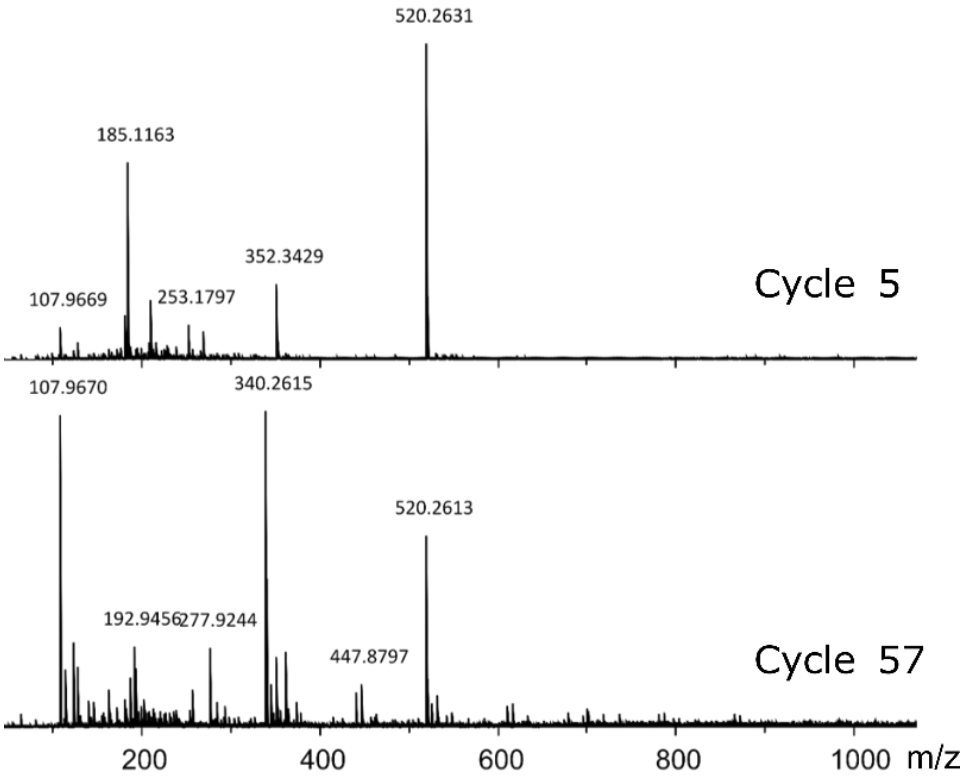


Figure 33: Spectra of Bruker chromatogram range from 12.2 to 12.8 minutes

Comparing HPLC-DAD data and MS data is difficult as these techniques depend on completely different rules and laws. In the DAD, every compound that is UV active at the chosen wavelength will be detects but species that are not UV reactive would not be observed. In the mass spectrometry, it depends how well the mixture ionises in the mass spectrometer.

It has become obvious that the comparison between two different mass spectrometers is very difficult, due to their differences in mass accuracy and sensitivity. Of course, the method is a crucial factor in every analytical process and when samples are analysed offline, on a powerful analytical instrument where every setting can be directly adjusted for the specific sample the analytical options are much bigger.

A further investigation of all found peaks and species can be done, but would require to search databases and to test these findings further with standard solution runs. As this work focused in finding an approach to fingerprint complex product mixtures, this overview of individual data analysis shall be sufficient, but determining individual products in such a product mixture is important future work to fully understand the chemistry of these reaction systems.

3.2.3 Mz index

As this platform is designed to run in a fully automated fashion, it was not just necessary to have a reliable running hardware system. One of the key points of this project was to find a way to evaluate the data of every run and be able to quantify the change differences distinguishing between two different samples of one run. A measure to assign change was needed. Another group member developed an algorithm to use in our experiment to assign the change from run to run in an early stage of this project. This evaluation was called the “mz index”. The number that this index assigned is used as a tool to compare different cycles with each other. The principle how we assign the change is shown in Figure 34.

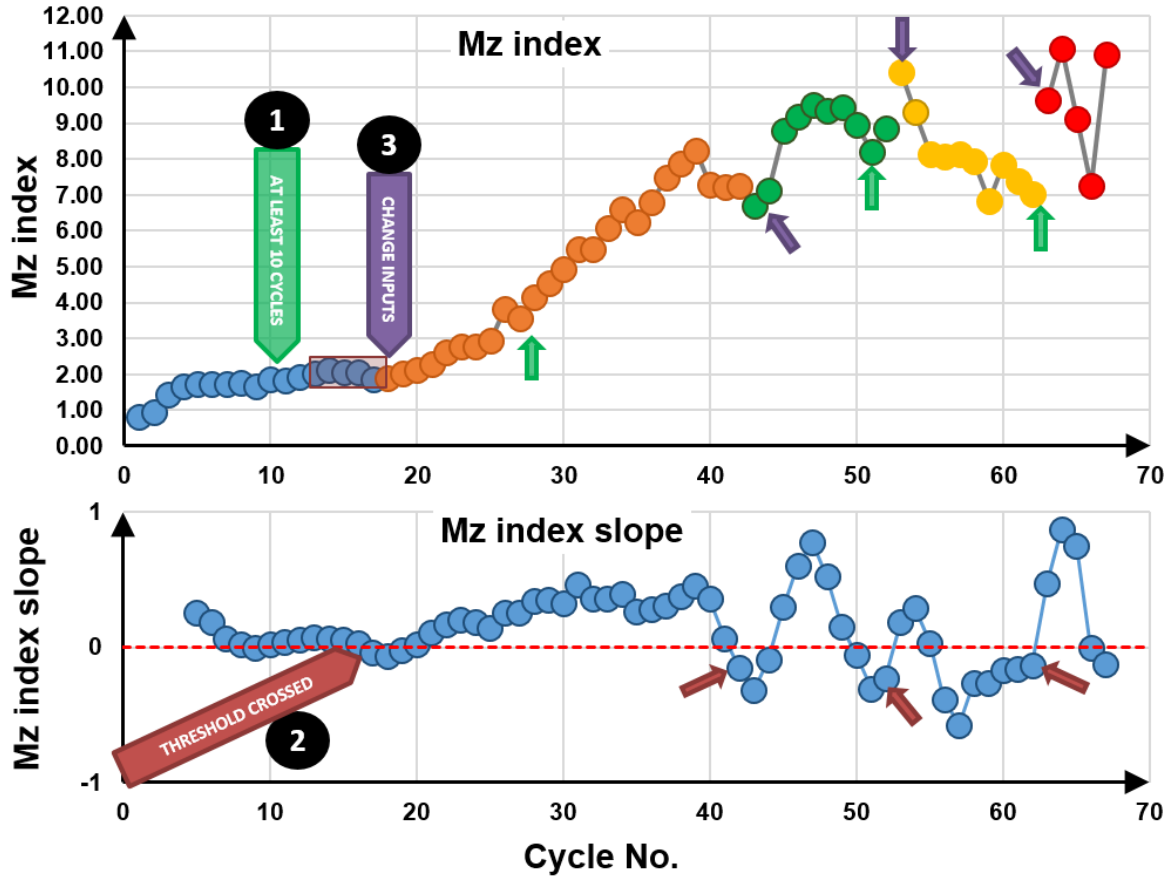


Figure 34: Concept of assigning change, visualised with real data of a run assign with the mz index. The top graph shows the mz index of each cycle with each input set in a different colour, while the graph in the bottom shows the slope between the mz index values of the cycles

The top plot shows a graph based on real data. Every point on this graph is one cycle of a platform run and every colour represents a different set of inputs. There was one set rule for the algorithm, each input decision was after the first ten cycles of the same input set. Number one in the top graph shows the point where ten cycles have been run. At this stage, the algorithm starts to calculate the slope between the different mz index values of each cycle. The slope is shown in the graph in the bottom, where again every point represents one cycle corresponding to the points in the graph above. As the corresponding point of the slope to the point at number one is above zero, meaning that the mz index is rising, the input set does not get changed immediately at that point. The algorithm continues to run with the initial input set and assigns the mz index slope for every cycle new. As soon as the slope crosses the threshold and declines, the input set is changed with the next cycle, as it is shown in number three. This procedure proceeds until the run is stopped.

As writing the code for this algorithm was not work done by the author, the code will not be presented, but its principle will be explained (Figure 35). The python code reads the netCDF files from the Advion mass spectrometer and calculates the mz index, using numpy, a python library that allows mathematical functions and the work with large data sets. For every cycle, the code extracts the total ion chromatogram of the sample. From the TIC a spectrum for each retention time point can be taken and the algorithm searches for the heaviest and lightest peak over the set threshold $1e^6$. After that, the lightest peak is subtracted from the heaviest one and the resulting number is divided through the total number of peak above threshold. The numbers of all spectra of one TIC averaged, result in the mz index for a single cycle and all values are stored in a list. This list is accessed by another python code, which calculates the slope of the mz index based on the data in the list. If the calculated slope is below zero and a set amount of cycles (depends on the run, usually a set of inputs was run for 10 cycles until a decision was made) was run with the same set of inputs the algorithm decides to change the input composition. The mz index value sets the highest and lightest peak in relation with the total number of peaks. If we have a very high heavy peak and just a few other product species, the number will be high. The more peaks are detected and the smaller the heaviest peak is located, the lower will the mz index value be. This is based on the idea to tame a combinatorial explosion and to build

a reaction system, which can produce high mass molecular species (complex molecules) that dominate the product mixture. Further, by calculating the slope between mz index values of different cycles, we can determine if the reaction changed in between cycles.

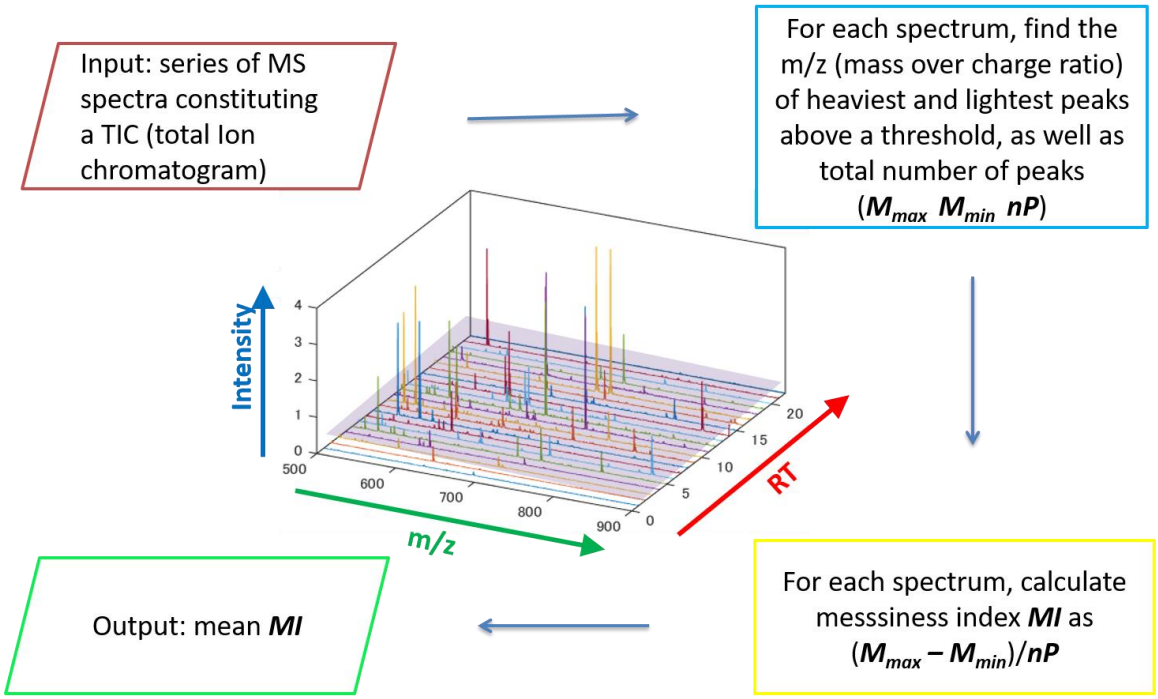


Figure 35: Schematic of stepwise description how the mz index evaluates data

With this experimental framework and the software and hardware set up, we have been able to execute a number of different automated runs. These runs have been executed over the whole duration of this project, so conditions have changed between some runs. Run A was the first successful run on the platform with the inline analytics and the decision-making working, but the mx index itself seemed to be very low. The only explanation for this fact must be the input decision of the code. Run B and C are repeats of each other (further discussed, later in this work). Run D is a repeat of run B too, but the inline analytical system failed and the data shown here is from an offline HPLC-MS run. The analysis was performed on the same instrumentation, but through a different sampling method as the samples were injected through the auto sampler of the HPLC system and not directly, from the reactor through the home build sample valve, this seems to change the data as expected. An explanation for the significant differences to all other runs and especially the runs that have been supposed to be identical could

be the fact that, in contrast to other runs, these samples have not been analysed immediately. They were stored and filtered through syringe filter before analysis.

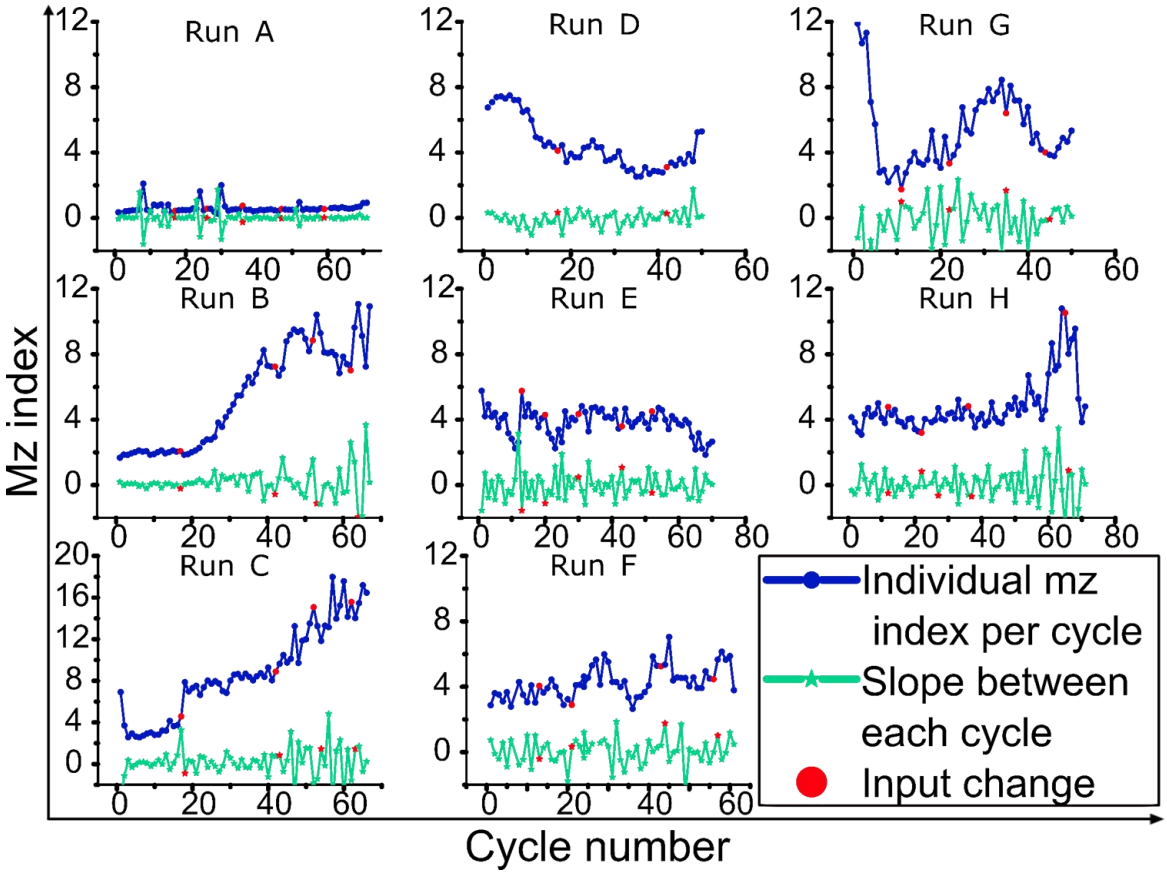


Figure 36: Selected runs and their mz index plotted against the number of cycles and the slope in-between cycles

This process does not just cause a difference to the product mixture, but may have led to the degradation of products, for example through the repeated exposure to air on several occasions in the sample preparation process. The samples that just came out of the reactor that is kept continuously under nitrogen should not touch air before analysis. Considering these facts, it is not surprising that the product mixture changed so much.

Run E used a new input set again and looks almost random at first glance. We increased the concentration of the input solutions from 0.1 M to 0.3 M (except for ruthenium chloride which is 0.01 M in both runs) up from run F and we shortened the cycle time from six to three hours up from run G. All runs from E to H have different input set and as a result, the graphs differ dramatically from each other.

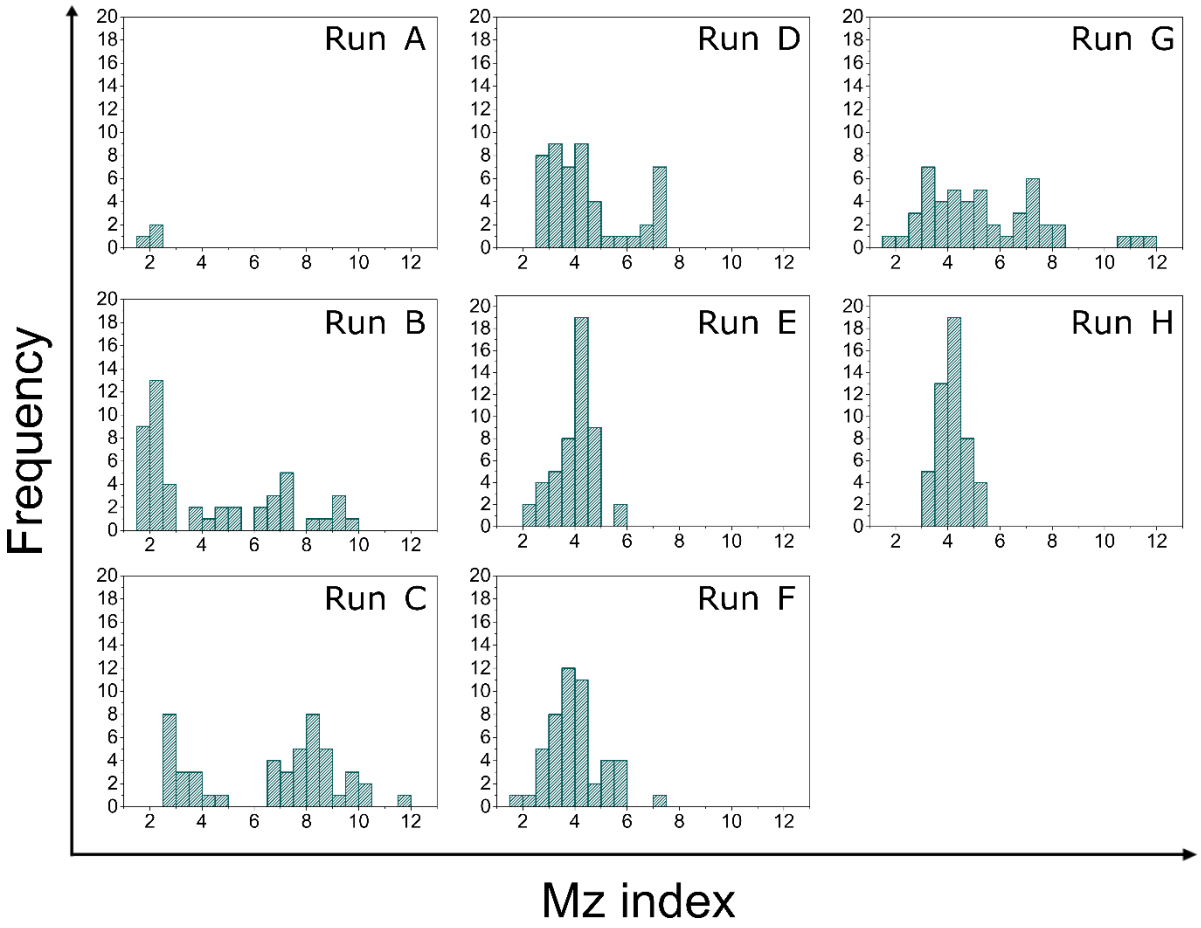


Figure 37: Histograms of different runs, evaluated with the mz index

The differences between the runs get even more obvious, when we compare the range of mz index by visualising them into histograms in Figure 37. Run A is again prominent in this comparison, as the mz index calculation seemed to be very low, compared to the other runs. We can observe as well that Run B and C are more similar to each other while the other runs have big differences in their mz index range.

It is interesting to see how much the individual runs differ from each other. Based on the decision making algorithm, how the data is assigned and that the experiment is changed, if no change between cycles is detected could have resulted in the random generation of hill-climbing plots but this is not the case. Each individual plot looks different and there is no clear pattern or trend observable, which on the same side is not surprising as the chemistry, although aqueous, differs dramatically in each run as shown in Appendix Table1. Even the reduction of the cycle time from 6 (run A to F) to 3 hours (run G and H) seem to not to make a difference in the overall comparison between cycles.

3.2.4 Repeated runs

As one of the advantages of automation in chemistry is supposed to be a better reproducibility of reactions, one of the first sets of runs executed were the repeats of a random chosen run. For this run, the algorithm was given 18 input solutions and 2 rules. Rule 1 was that the algorithm needed to wait 10 cycles until an input decision was made, rule two was that if change happens, all three inputs would be changed in a completely random manner.

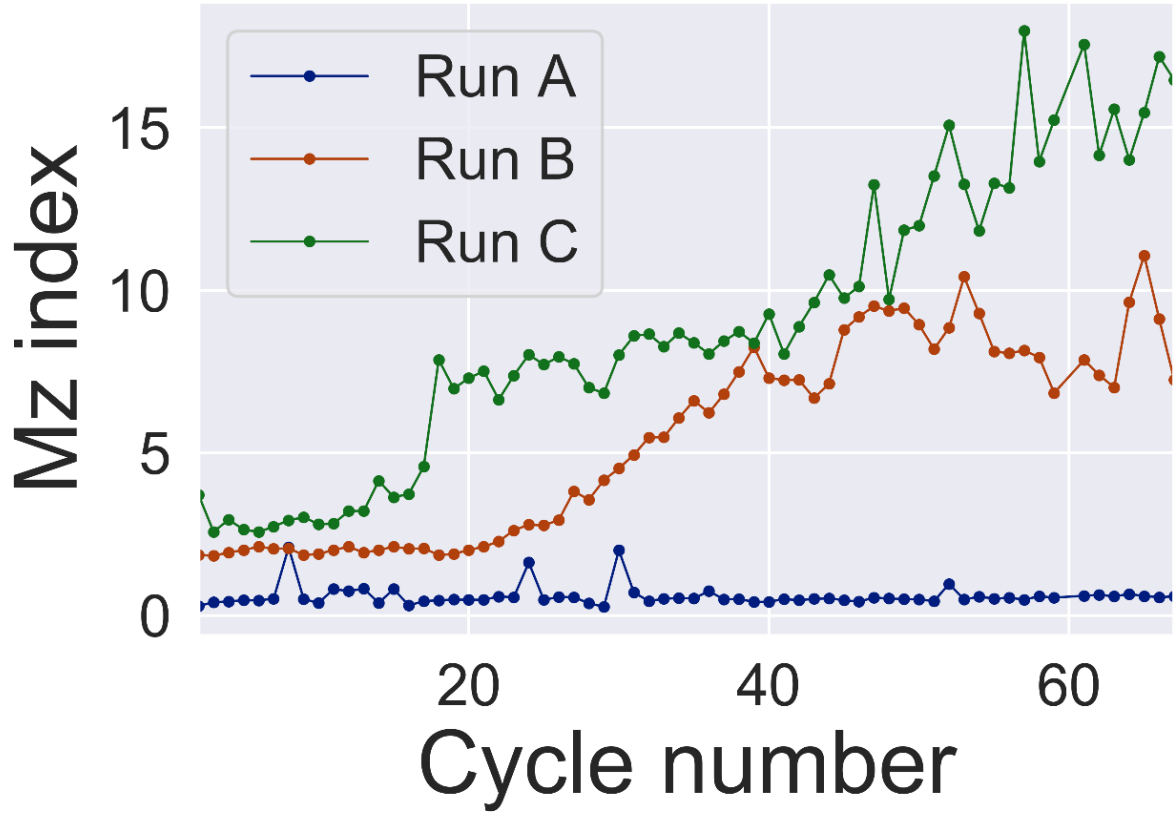


Figure 38: Three runs compared calculated with the mz index. Run B (red) is a repeat of run C (green), run A (blue) is a run with a completely different set of inputs for comparison

For all three runs in Figure 38, the reactor got charged with pre-washed minerals (ulexite, pyrite, and quartz), 2 g each in a size between 2 and 4.75 mm. The stirrer plate was set to 60°C and the stirring rate was 200 rpm, every reaction cycle was 6 hours long.

We see in Figure 38, that run B and C are following about the same mz-index trend. The index inclines almost gradually throughout the run, until the change of the slope increases up from cycle 40 and changes more rigorously with each cycle. The similarity between both runs gets more obvious when compared with another run, like run D. Run D had the same input solutions but the analysis, as mentioned before, was carried out after the run was finished and from the normal HPLC-DAD autosampler. Through this circumstance, the data looks very much different. These observations get more obvious when looking at the histograms of these specific runs in Figure 39. As mentioned above that the use of automation should improve the reproducibility of experiments, these results are not satisfactory. B and C follow the same reaction trend, but the individual mz index values differ quite dramatically from each other. This can be explained due to the conditions of the experimental set up. Because the platform allows just one reaction at a time, the performance of each experiment is months apart, which can lead to fluctuations in humidity, light and room temperature. On top of that can be slight variations in the input solutions/ mobile phases/ minerals (even if they have been careful handled and accurate recorded and the same reagent batch was used) and the fact that there are always changes in the way a specific mass spectrometry run ionises add to the differences in the mass spectrometry spectra too. All these conditions could lead to an explanation, why even the repeated runs, differ slightly from each other while following the same pattern.

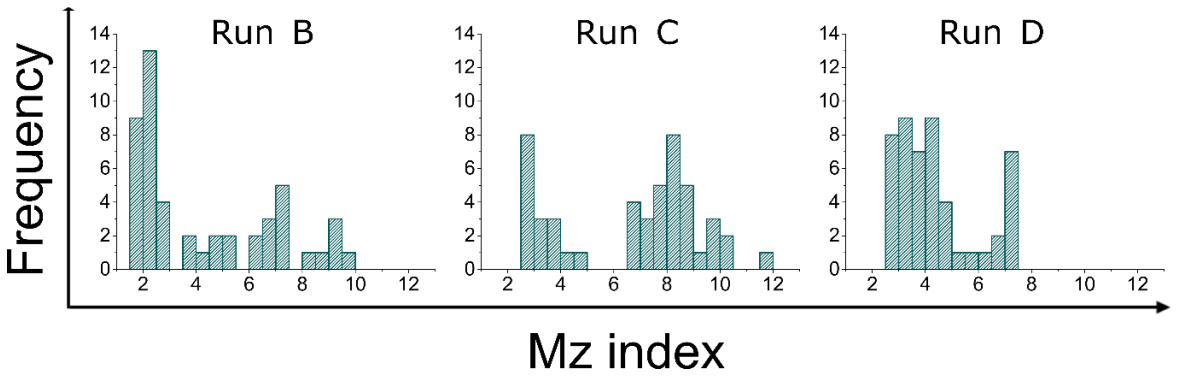


Figure 39: Histograms of the mz index of the three compared runs. Run B the initial run is on the left and the repeat, run C is in the middle. On the right is run D, which is a repeat but analysed after the run was completed, leading to major differences in the data

The mz index was an approach to assign a number to a mass spec chromatogram, but this approach might not be the most useful tool. There are some issues with

this algorithm. For instance, it only relies on a mass spectrometer, which is useful for fingerprinting so the data meaning that the accuracy of the individual peaks is limited. This means that the noise itself, which the threshold usually cuts off, might not be a massive problem and it is more necessary to not accidently cut off products with a too rigid thresholding strategy. The thresholding in the mz index code arbitrarily set and not set, based on each particular data set. Additional to this, the code just looks for masses of a spectrum, the peaks in between are not taken into the calculation and there is no relation to the intensity of each peak. This algorithm is not really a way to assign if a product mixture is “interesting”.

3.2.5 Modified mz index

Based on the issues that have been highlighted, with the initial used mz algorithm, alternative ways to evaluate the data were found. As one of the problems of the mz index, was the fact that there was no relation between the mass of the peak and its intensity, it was interesting to investigate how the data would change if the intensity would be multiplied by the mass of the corresponding peak.

What presented itself as an easy task turned out to be a slightly more challenging undertaking, as there have been two bottlenecks that made the editing of existing code difficult. For once the previously written code that was used to assign the online data was not modular but the more challenging problem was the fact that the Advion mass spec limits its data output to two different file types. The choice was between using the format the Advion software is using, “datx” or the more ominous “cdf” format. After many different approaches to open either of these files through code and to get the data into a format that made sense while comparing it to the actual data in the Advion software to check if anything made sense, it proved as a rather overly complicated task. Another group member contacted Advion and it was possible to get an API key that could open the datx file. With this API key, one was able to convert the datx file into a csv file with a row for retention time, the m/z, and the intensity saved next to each other, if run on a Windows 32-bit machine.

After it was possible to access the data, the thresholding procedure needed to improve. Several different data manipulations have been tested and evaluated and a useful way for thresholding was found. The best way for thresholding turned out to take the median of the intensities added to half of a standard deviation unit, see Figure 40. This way of thresholding was chosen as it enables the code to capture every single peak of a spectrum, while filtering the lowest amount of small peaks over noise. This was important, as we wanted to take species into our calculation that are too low in abundance to be over a set threshold, but are different from starting material and noise. In the first part of the figure, all required libraries are imported. The function starts with reading the csv data into a data frame. In the second part of the function, the threshold of each intensity is calculated and added as a column to the dataframe. Dataframes are objects to store tabular data, specific for the data analysis toolkit Pandas, which can be used in Python. This thresholding strategy is used for every described code below.

```
import numpy as np
from numpy import array
from numpy import matrix
import pandas as pd
import matplotlib.pyplot as plt


def get_df(df_file):
    #opens csv file and reads it in as a df
    my_df = pd.read_csv(df_file)
    list(my_df.columns.values)
    #set header right
    my_df.columns = ["rts", "mz", "int"]

    #adds another column to the dataframe with the intensity substracted by our set threshold
    #our threshold is the average threshold with the standard deviation divided by two added
    threshold = ((my_df.loc[:, "int"].median()) +  (my_df.loc[:, "int"].std()) / 2)
    thresh_df = my_df["int"] - threshold
    #add thresholding column
    my_df["threshold"] = thresh_df
    my_df[my_df < 0] = 0


    #this adds a column to the dataframe where we multiply mz mass with the thresholded intensity
    mz = my_df["mz"]
    t_int = my_df["threshold"]
    multi = mz * t_int
    my_df["multi"] = multi
    return(my_df)
```

Figure 40: First part of the modified mz index code. The first block shows the libraries imported, the function shown here opens a csv file and adds a column with the calculated threshold to the dataframe. In the last part the threshold column is multiplied with the mass

The last part of Figure 40 shows how the mass is set in relation with the intensity of the peak. Another column “multi” is added to the data frame in which the mass of each peak is multiplied with the thresholded intensity of each peak. Instead of using just the mass value in the mz index before, we are looking for the heaviest and lowest peak and are saving the corresponding multiplied value of this particular peak into two separate list. After that, the number of peaks is determined for the whole spectrum and this number is stored for each retention time group (representing the spectra in the dataframe) in an additional list. With these three lists, it is possible to calculate the modified mz index, see Figure 41.

```
def mz_index_calculation(my_df):
    #this calculates the actual mz index value
    list_max_number = max_number(my_df)
    list_min_number = min_number(my_df)
    peak_count = peak_counter(my_df)

    #first substract the minimum number from the maximum for every value in both lists
    a = matrix(list_max_number)
    b = matrix(list_min_number)
    sub = a - b
    #we convert the substraction list into an array that we can devide later
    sub_array = array(sub)
    #turns peak count into an array
    peak_array = array(peak_count)
    #divides the substracted min and max values through the number of peaks
    mz_list = np.divide(sub_array, peak_array)
    #takes the average of the total mz calculation
    mz_value = np.average(mz_list)
    return(mz_value)
```

Figure 41: Function to calculate the modified mz index, using the python libraries numpy arrays and panda data frames

To compare both evaluation methods, the mass spec data from the inline-measured samples is rerun with the modified version of the mz index code (Figure 41). The comparison is to be made carefully, as not just the relation between peaks and intensity is changed but on top of that, the thresholding. The values of the plotted data changed completely, so the comparison of these two data sets is limited to the observation of “data trends” and can be seen in Figure 42.

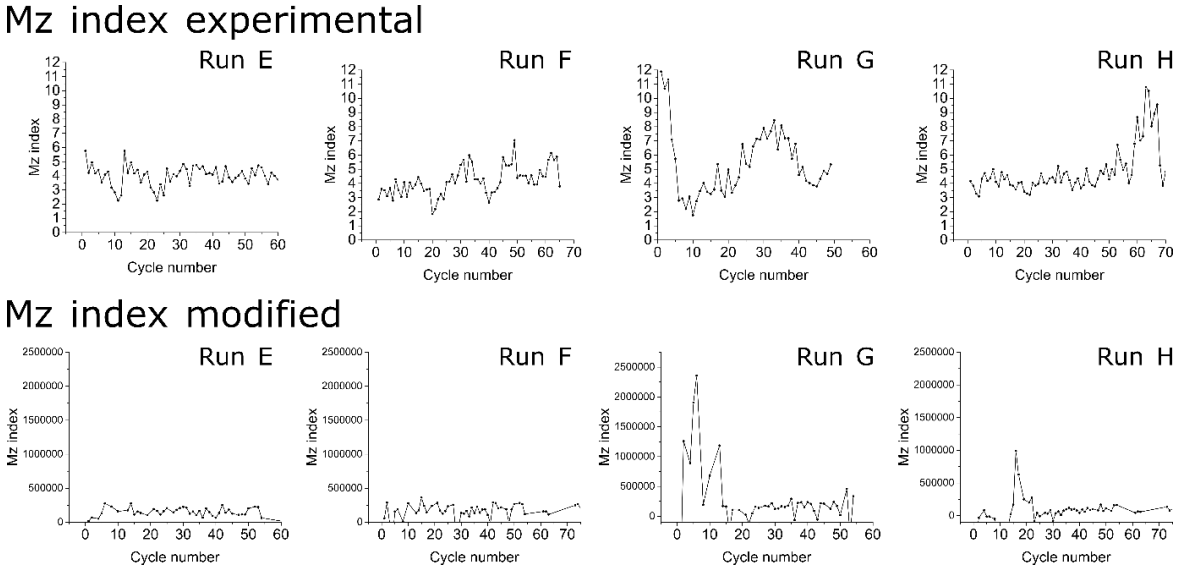


Figure 42: Data evaluated with experimental mz index and modified version, applied on same data

The plots presented in Figure 42, do not show an immediate trend, which would be possible to observe. In general, the modified mz index seemed to have lowered the complexity or variation of specific cycles. In the experimental mz index comparison, all runs are in a similar scale, as when a similar scale is applied to the modified mz index, one run, run G stands out the most. This correlates with the experimental mz index, as the experimental mz index starts very high, until it drops around cycle 10. The modified mz index does has a rise in the beginning but the pattern of being high and dropping around cycle 10 can be observed in this case too. Run E and F are looking fairly similar in both calculations and run H seemed to be turned over its vertical axis as there is a rise visible at the end in the experimental mz index and with the modified version, a rise is possible to observe around cycle 20, which drops again afterwards. The algorithm would have clearly made different input choices than the experimental algorithm, but if this algorithm would make more sense than the mx index that was in use is questionable.

3.2.6 Weight by intensity index

To find an alternative way in evaluating the mass spec data, different approaches to calculate the data have been tried. One of them follows a strikingly simple approach. The aim of any algorithm in this experiment is to create a number, which gives the opportunity to evaluate a cycle in comparison to following cycles. As generally the aim is to look into a complex mixture, it should be possible to observe if the number of peaks or the mass of compounds increases. Based on this idea, an evaluation method based on the weight and the intensity of each peak was developed. In this calculation, the aim is to get a number z, which is the sum of all intensities multiplied by their mass value over each spectrum.

$$z = \sum I_p * \frac{m}{z}_p .$$

When this value is high, the sample has a higher amount of larger products, with a stronger signal.

As just multiplying every value would lead to very high numbers, the intensity is normalised before multiplied with the mass value (Figure 43), similar to the multiplication step in the calculation for the modified mz index (Figure 40).

```
def normalise(my_df):
    #we are iterating through every retention time group
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    #add another line into our df with normalised intensities ip, currently just with zeros stored
    my_df["normed_I"] = np.zeros(len(my_df["int"]))
    #want to get intensities for each spectra normalized by the total intensity for that spectra
    #(e.g. the total intensity for each rt should be = 1)
    for i in range(n_rt):
        #iterates through every retention time group
        rt_df = my_df[my_df["rts"]==unique_rts[i]]
        #sums up all intensities in a rt group
        T = sum(rt_df["int"])
        #divides the sum of all intensities T through each intensity in our df
        rt_df.loc[:,"normed_I"] = rt_df.loc[:,"int"]/T
        #adds or rt_df into our main df as our normed_I column
        my_df[my_df["rts"]==unique_rts[i]] = rt_df
    return(my_df)

def multiply(my_df):
    # add multiplied mz with threshold intensity
    mz = my_df["mz"]
    int_norm = my_df["normed_I"]

    multi_normed = mz * int_norm
    my_df["multi_normed"] = multi_normed
    return(my_df)

def sum_it_up(my_df):
    #sums the multi column and returns the multi sum
    total_sum = my_df["multi_normed"].sum()
    print(total_sum)
    return(total_sum)
```

Figure 43: Functions for the calculation of the normalised mass over intensity value

In the last part of Figure 43, the sum of all normed and multiplied values of the cycle is taken and saved to a list. The experimental inline Advion data is rerun with the new calculation and the data is presented in Figure 44. In this figure, the weight by intensity value of run E to H is shown. As all of these runs have different input sets, there is not much to compare in between these runs but when the data is compared with the previous algorithms in Figure 42, some differences are visible. As these are different calculations, the values on the y-axis on every plot are incomparable, therefore is the comparison limited to a description of the shape based on the cycle number.

Interestingly run E differs the most from the previous calculated mz index values. The weight by intensity values are compared to the other plots calculated with the same algorithm much higher, which leads to an elevation of the whole graph in the plot while the values are in the same data range as all other calculated values in data compared with the mz index in Figure 42. Run F looks similar as in

the modified mz index. While the experimental mz index has a clear rise between cycle 30 to 40 and again at approximately, cycle 50. The weight by intensity calculated values and the modified mz index values have several small peaks but no clear rise throughout the run. The weight by intensity run G is more similar to the experimental mz index calculation, showing a rise right in the beginning, but the rise later in the experimental mz index calculation, visible in Figure 42 is not observable in the values shown in Figure 44.

Run H differs significant from the mz index calculations. There is no massive peak observable but the values show a rise around cycle 30, which is not visible in the data previously calculated. An overlap with the experimental mz index is that another peak between cycle 60 and 70 is visible, which is a very distinct peak in the graph of the experimental mz index of run H shown in Figure 42.

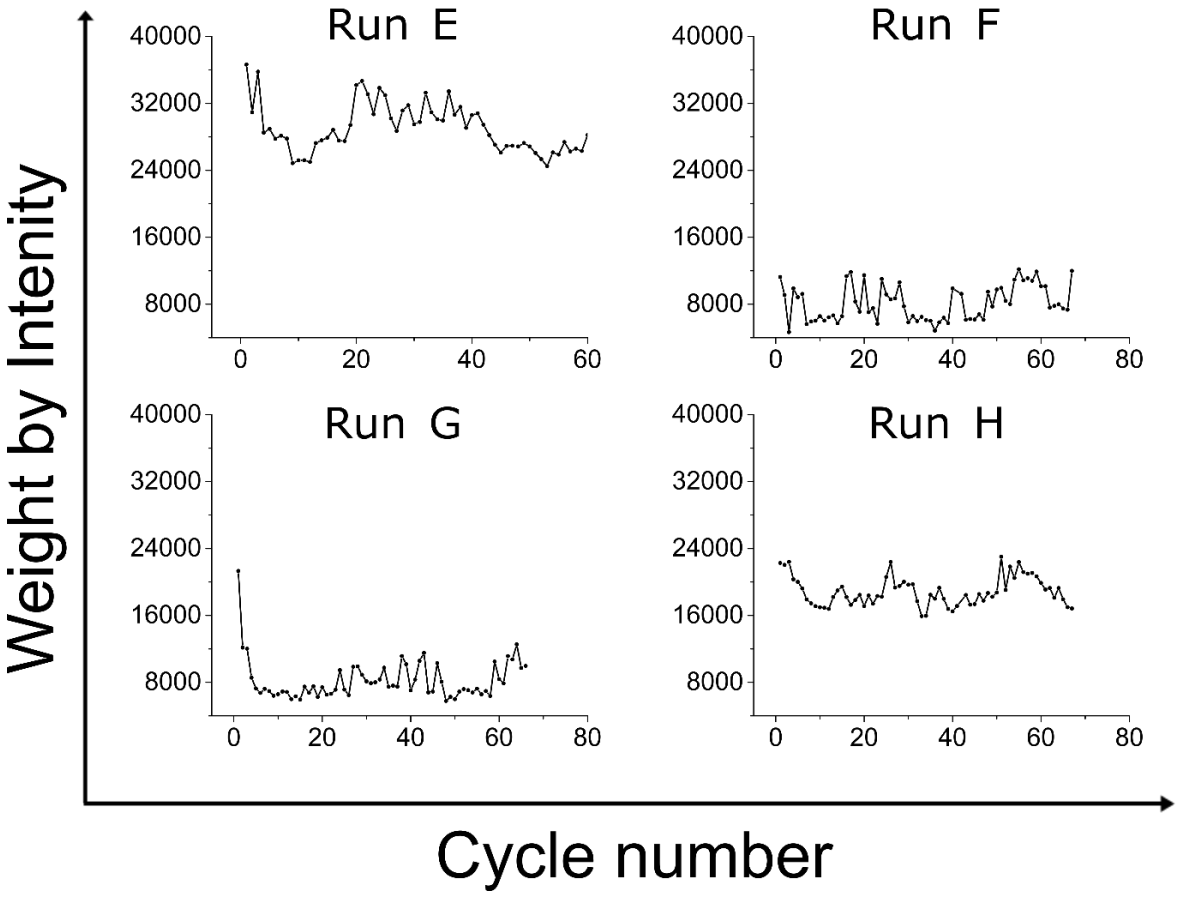


Figure 44: Data evaluated with the weight by intensity calculation.

The changes in the data based on a different mathematical approach have been interesting so the approach was taken another step further to another new algorithmic idea.

3.2.7 Information entropy value

In this approach, a code is developed to compare the cycles of a run based on their information entropy. Entropy is often defined as a value for the disorder of a system, but in this case, it is used to describe the information content in our system[94]. The information entropy of a spectrum is defined as:

$$S = -\sum_{p} i_p * \ln(i_p).$$

Where $i_p = \frac{I_p}{T}$, and $T = \sum_p I_p$, i_p is the intensity of peak p normalized to the total intensity of the spectra. This leads to a value which will be lower when the sample has fewer, larger peaks and higher when the sample has many peaks of comparable size. The code starts in a similar manner to the previous ones by transferring the data into a pandas dataframe and setting the threshold for each intensity value. The intensity values are being normalised but this time not multiplied with the mass values. How the entropy value is calculated is shown in Figure 45.

```
def calculate_entropy(my_df):
    #we are iterating through every retention time group
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    #this adds a column to the dataframe that has the ln of ip stored
    my_df["log_ip"] = np.log(my_df["normed_I"])
    #this adds a column that has the ln of ip multiplied to ip stored
    my_df["entropy"] = my_df["log_ip"] * my_df["normed_I"]
    #create an empty list
    entropy = []
    #we are iterating through every retention time group again
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    for i in range(n_rt):
        #iterates through every retention time group
        rt_df = my_df[my_df["rts"]==unique_rts[i]]
        #sums up the entropy for a spectra by summing it up for a retention time group and
        #stores it into our entropy list
        S = sum(rt_df["entropy"])
        entropy.append(S)
    #takes the average of the entropy of all spectras
    total = statistics.mean(entropy)
    print(total)
    return(total)
```

Figure 45: Function to calculate the entropy value of Advion data

This function iterates through each retention time group and adds a column of the natural logarithm of each individual normed intensity to the data frame. This value

is then multiplied with the value of the normed intensity. As the value desired is the entropy over a full spectrum, the code iterates through the data frame again, summing up all entropy values for each individual retention time group, resolving in the entropy value for each individual spectra. To generate a through the run comparable number, the average of the entropy of all spectra is taken and returned as entropy value for the individual cycle. The resulting data is shown in Figure 46.

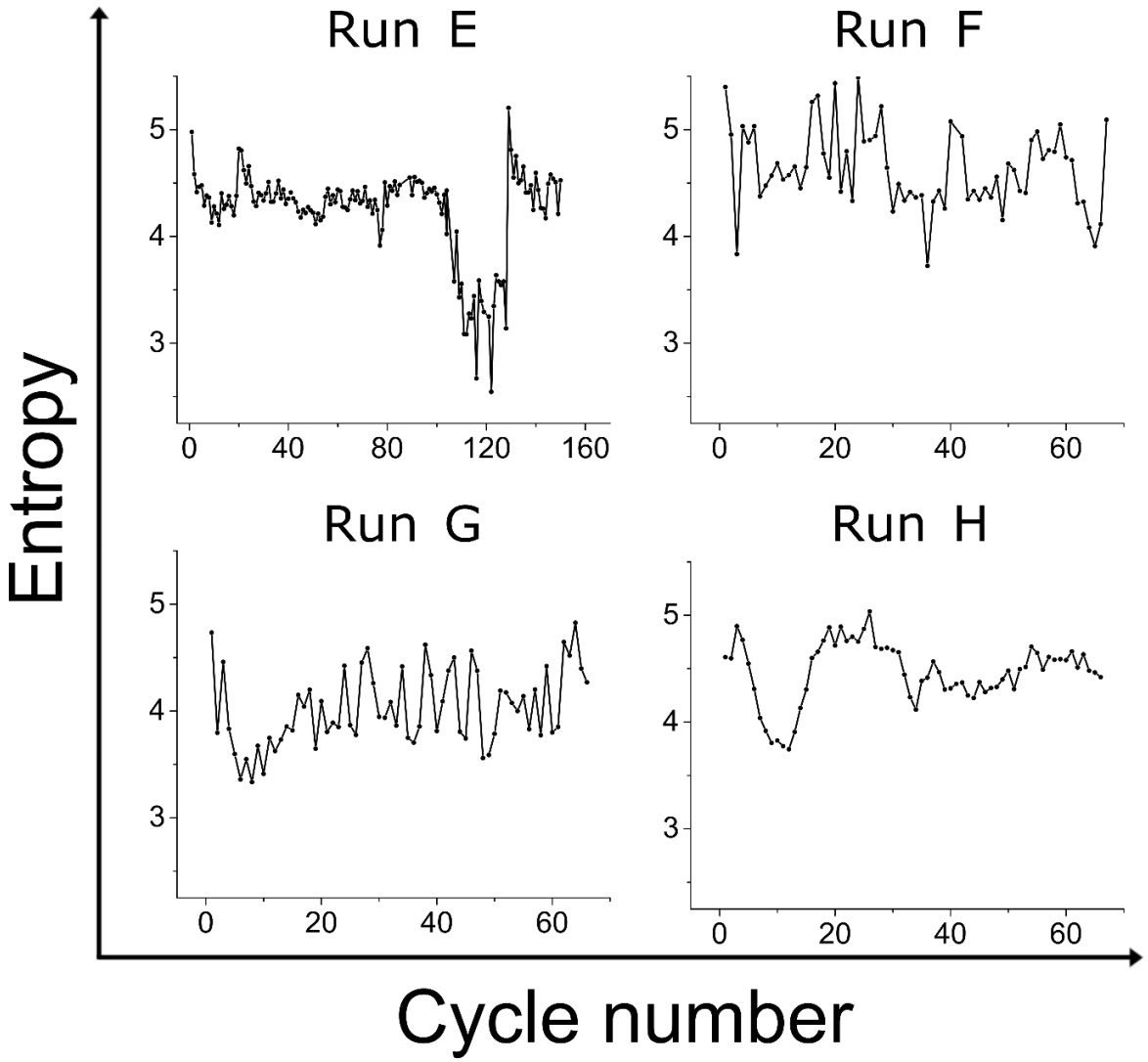


Figure 46: Data evaluated with the information entropy measure

As before, the data will be compared with the mz index used in the experiment, as this was the code of which the input decisions have been based on. The graph for the entropy in run E in Figure 46 is moderate throughout the first 100 cycles and declines steep after that until it rises again around cycle 130 to cycle 150. The graph has no similarities to the mz index calculates graph in Figure 42 though.

Different to the entropy graph of run F that follows approximately a similar pattern than the graph of the mz index calculation but looks almost random. Both figures show a quite disorganised pattern but both plots rise between cycle 20 and 40. The run G plots differ heavily from each other. The mz index graph in Figure 42 is high in the beginning, descends until cycle 10 and follows a curve after that. The information entropy graph in Figure 46 of run G shows random distribution of points and the value is going rapidly up and down between cycles. Run H is different, as the entropy plot declines until cycle 10, rises after that until it reaches cycle 20 to cycle 55, where it starts to follow another more moderate curve. The mz index plot of run H in Figure 42 shows a moderate course until the index rises rapidly after cycle 60.

If we compare the algorithmically produced data of each run individually, we will compare the modified mz index, the weight by intensity and the information entropy as the same thresholding strategy was used in all 3 calculations. It turns out, the different algorithms give almost complimentary information about the individual runs. While run E has the highest weight by intensity value in comparison with the other 3 runs, which leads to the idea that this run has many product species of high mass value and high abundance, the information entropy of this run shows a drop around cycle 120. This means that in this range of cycles, rather than having a high amount of many product species we got a few very large peaks, which is plausible based on the weight by intensity value, as this value can not distinguish between a high amount of peaks and a few intense peaks, in opposite to the information entropy value. On the same side, when considering the mz index, it suggests a number of larger peaks as if there would just be one dominant peak, this value would be high too. For run F, the weight by intensity index is relatively low throughout the run, this would lead to the idea, that there was a lower amount of species and the overall abundance of the signal was lower too. This explains the high information entropy, as it suggests many peaks of comparable size, which do not contradict the weight by intensity value. The mz index of run F is low, which shows that there is no dominant species of high mass, but rather a higher amount of peaks of lower abundance. The interpretation of run G appears more complex. The mz index value shows a high rise in the beginning of a run, suggesting an abundant dominant species in the beginning, which breaks down throughout the run. This can be further validated by the weight by intensity

value, as this value shows a rise in the begin of the run. The information entropy, shows a drop in the same area, suggesting a fewer number of peaks but not such a clear trend as when calculated with the other two values. Run H shows a peak in the area between cycle 10 and 20, which suggests the build-up of a few species of higher mass, which break down into more species afterwards. This rise is not clear to read in the weight by intensity value as this value shows a small rise at that point but a more dramatic rise later in the run, in which the mz value does not show a rise. On the other hand, the information entropy shows a drop, suggesting that there are rather few larger species, like the mz index suggested. It is interesting to see that observations based on the different algorithms can, if handled carefully, build on each other. On the other it is important to state, that the algorithms alone are just able to give ideas of the product distribution in a sample and if a cycle contains dominant product species. For more information about the exact chemical composition and the reactions, which did occur in the system, a more extensive analysis is necessary.

Looking at LC-MS data from a system prospective provides new insights but also leads to new problems. In particular, the possibility of overlooking interesting effects. It is very difficult to find a sensible approach, which helps to assign and describe the bulk properties of the data, and to highlight which samples are the most promising for further manual analysis. Looking at mass spec data might already be too narrow of an analytical technique. A possible alternative might be to find a way to assign a HPLC-DAD spectrum and compare it to the MS data for algorithmic analysis. As just one algorithm presented here was experimentally tested, it would be interesting to see how alternative decision making processes, for example the information entropy algorithm, would perform making experimental input decisions and what input set and decision ratio that would lead to. I will leave this topic for future experiments.

4 Conclusions and Future Work

An automated platform was successfully developed, built, and can now be used for a range of future experiments. Through the modularity of the build, it is quite easy to add or change elements based on the desired chemical set that is to be explored. Experiments for the near future on the platform are already planned and are going to investigate how complex mixtures are able to constrain each other. For this, two different chemical input sets are run on the platform in alternating recursive reaction cycles. This is two chemistries which might not be expected to interact which each other on the first glance, and the aim is to study how the reaction sets are "reacting" in the presence of one another and if there might be a chemistry which will dominate. Chemical sets which could be tested in this project are amino acid condensation reactions (which lead to the build-up of peptides) crossed over with the acidic citric acid cycle chemistry, an aldol condensation or the formose reaction.

An attempt to find a useful algorithm to assign data was made. Nonetheless, based on the data presented so far and with the data that the calculation would be fed, none of the algorithms are able to assign whether the reaction is "interesting" on their own. Interesting data could be, to give one example of many, the discovery of a dominant species, which can "survive" several cycles and in the best case even a change of input solutions, which might indicate an autocatalytic behaviour of the system. Further development is needed to algorithmically explore phenomena in complex reaction mixtures, but the results presented herein provide an important first step. For future work, it might be more effective to have an algorithm assigning the data, which is extensively tested before use, to be able to precisely predict when the threshold between dead and alive matter is crossed, and rather concentrating on the factor of complexity of that reaction system than just to compare different samples with each other.

If there would be another platform built in the future, it would be of high interest to concentrate on an origin, chemical space exploration, and high liquid throughput system. The fact that the current platform has just one single reactor, and a quite high amount of liquid per sample and cycle, makes the system very slow. As there are not enough indications in which chemical direction the

exploration should go, it is important to be able to screen a high amount of small samples, rather than have a few big samples. On the same side, having the ability to run several reaction at the same time would lead to the option to run every sample in triplicate which is just better analytical practice, this is especially beneficial when reactions are run in cycles for months, as reaction repeats would just not be feasible through time constrains.

The options in the field of origins of life and automation of chemistry research are almost unlimited, and it will be exciting to see where both fields are going in the future.

5 Experimental

5.1 General

All materials and solvents were purchased from commercial sources (Sigma Aldrich, Fischer Scientific and Alfa Aesar) unless otherwise stated and used without any further purification.

Ulexite and quartz were obtained from Richard Tayler Minerals, Cobham, Surrey, England and crushed in a Mad Mining Rock Crusher with a Solid Steel Frit.

5.1.1 Chemical input Preparation

All input solutions have been prepared on demand in HPLC-grade water as followed:

Table 3: List of chemical inputs and their reagent amounts in 250 mL solutions.

Input	Chemical	Volume in mL	Molar concentration in M	Mass in g	Volume in mL
1	resorcinol	250	0.1	2.75	
2	pyruvic acid	250	0.1		1.74
3	acrylic acid	250	0.1		1.71
4	glycidol (2,3-epoxy-1-propanol)	250	0.1		1.66
5	glycerol	250	0.1		1.83
6	carbonyldiimidazole	250	0.1	4.05	
7	ethyl acetate	250	0.1		1.98
8	pyridine	250	0.1		2.02
9	oxalic acid	250	0.1	2.25	
10	formamide	250	0.1		1
11	catechol	250	0.1	2.75	
12	ruthenium-(III)-chloride hydrate	250	0.01	0.519	
13	formaldehyde	250	0.1		1.86
14	copper-(II)-sulfate pentahydrate	250	0.1	6.24	
15	sulfuric acid	250	0.1		1.3
16	nitric acid	250	0.1		1.04
17	ammomium thiosulfate	250	0.1	3.71	
18	potassium pyrophosphate	250	0.1	8.26	

5.1.2 Mineral wash workflow

All minerals used have been sieved to a size between 2 - 4.75 mm. The minerals have been boiled and stirred in HPLC-grade water for 2 hours and continuously rinsed with fresh HPLC-grade water until the solution in touch with the minerals remained clear. After that, the minerals have been dried and directly transferred to the reactor.

5.1.3 Mobile phase preparation

For platform HPLC analysis, 0.1 % of sodium formate solution was added to HPLC-grade water or HPLC-grade acetonitrile and the solution was sonicated for an hour before set up on the instrument. For high resolution HPLC-MS, the procedure was similar, but LC-MS grade water and acetonitrile have been used.

5.2 Instrumentation

5.2.1 Platform High Pressure Liquid Chromatography (HPLC-DAD)

Gradient HPLC analysis was performed on an Agilent 1260 Series (Agilent Technologies) equipped with a quaternary pump (G1311B) and a diode array detector (DAD) (G1315D). The sample was injected from the sample loop on an Agilent Infinity Lab Poroshell 120 Eclipse EC-C18 UHPLC Guard 3.0 x 5 mm guard column witch was connected to an Agilent Poroshell 120, 120 EC-C18, 4.6 x 150 mm column, kept in a column compartment (G1316A) with a controlled temperature of 30 °C. The method used was a gradient method with 95% 0.1% formic acid added to HPLC grade water and 5% 0.1% formic acid added to HPLC grade acetonitrile (MeCN). Over 10 minutes, the organic (MeCN) flow was increased to 40%, after another 5 minutes it was at 50% MeCN. After 20 minutes a flow of 100% organic mobile phase was reached. After that, the mobile phase was switched back to the initial 95% water and 5% acetonitrile and a 20 minutes flow was maintained for column cleaning. The flow rate through the whole run was 0.5 mL/min while a 0.2 ml/min flow was maintained in-between runs. Elution was

detected by UV (λ = 200, 215, 245, 300) and samples were run for 40 minutes in total. The performance of the HPLC column was checked with a caffeine standard solution based on the directions in DIN 20481, on a regular basis.

5.2.2 Benchtop Electrospray- Ionisation Mass Spectrometry (ESI-MS)

The ESI-MS analysis was performed on an Benchtop MS: expression^{L} CMS system from Advion. After the sample passed the HPlC- DAD, it went through a split valve resulting into a flow of 0.2 mL/min injected to the ESI system. The mass spec was run in positive and negative mode, switching between both modes during analysis with a switching speed of 50 ms. The positive mode turned out to be more useful and the results of this mode only have been feed to the algorithm. The *m*/*z* Range was set from 10 to 2,000 *m*/*z*. The scan time was 3,345 ms and the scan speed was set to 595 *m*/*z*/sec. The ion source parameters have been as followed: capillary temperature 300 V, capillary voltage 120 V, source voltage offset 20, source voltage span 30 and the source gas temperature 200 °C. All settings have been similar in positive and negative mode, except the ESI voltage, which was 3,500 V for the positive mode and 2,500 in the negative mode. A calibration was performed regularly with a MS tuning mix (Agilent). Data was was analysed using the Advion Data Express software.

5.2.3 Electrospray- Ionisation Mass Spectrometry (ESI-MS)

The sample was run through a DIONEX Ultimate 3000 series HPLC-DAD set up with a RS (rapid separation) pump. Injected from the RS autosampler (WPS-3000 (T) RS) on an Agilent Infinity Lab Poroshell 120 Eclipse EC-C18 UHPLC Guard 3.0 x 5 mm guard column witch was connected to an Agilent Poroshell 120, 120 EC-C18, 4.6 x 150 mm column, kept in a column compartment (TCC-3000SD) with a controlled temperature of 30 °C. The method used was a gradient method with 0.1 % formic acid added to LC-MS grade water and 0.1 % formic acid added to LC-MS grade acetonitrile (MeCN). The run started with 100 % aqueous phase. Over 4 minutes, the organic (MeCN) flow was increased to 10 %, after another 12

minutes it was at 70% MeCN. After 19 minutes runtime, a flow of 100 % organic mobile phase was reached. After that, the mobile phase was switched back to the initial 100 % water. The flow rate through the whole run was 0.7 mL/min and the total runtime was 26 minutes. The HPLC was connected to a Bruker MaXis Impact quadrupole time-of-flight mass spectrometer with an electrospray source, operating exclusively in positive mode. The instrument was calibrated with a sodium acetate standard solution before each run. Samples were introduced into the MS at a dry gas temperature of 220 °C. The ion polarity for all MS scans recorded was positive, with the voltage of the capillary tip set at 4800 V, end plate offset at − 500 V, funnel 1 RF at 400 Vpp, funnel 2 RF at 400 Vpp, hexapole RF at 100 Vpp, ion energy 5.0 eV, collision energy at 5 eV, collision cell RF at 200 Vpp, transfer time at 100.0 µs, and the pre-pulse storage time at 3.0 µs. The mass range was set to 50 – 2000 m/z. Data was analysed using the Bruker DataAnalysis v4.1 software suite.

Bibliography

1. C. Patterson, “Age of Meteorites and the Earth,” Geochimica et Cosmochimica Acta, 1956, 230-237
2. A.Bonanno, H. Schlattl, and L. Paterno`, “The Age of the Sun and the Relativistic Corrections in the EOS”, Astronomy & Astrophysics, 2002, 1115-1118
3. C. H. Lineweaver, “A Younger Age for the Universe,” Science, 1999, 1503-1507
4. N. Kitadai and S. Maruyama, “Origins of Building Blocks of Life: A Review,” Geoscience Frontiers, 2018, 1117-1153
5. E. Schrödinger, “What Is Life? Mind and Matter,” American Journal of Physical Anthropology, 1946, 103-104
6. S. I. Walker, “Origins of Life: A Problem for Physics, a Key Issues Review,” Reports on Progress in Physics, 2017, 092601
7. D. E. Koshland, “The Seven Pillars of Life,” in The Nature of Life: Classical and Contemporary Perspectives from Philosophy and Science, 2010, 307-309
8. C. Sagan, “Definitions of Life,” in The Nature of Life: Classical and Contemporary Perspectives from Philosophy and Science, 2010, 303-306
9. C. Darwin, On the Origin of Species by Means of Natural Selection, D. Appleton and Company, 1859
10. S. A. Benner, “Defining Life,” Astrobiology, 2010, 1021-1030
11. K. Ruiz-Mirazo, C. Briones, and A. De La Escosura, “Prebiotic Systems Chemistry: New Perspectives for the Origins of life,” Chemical Reviews 114, 2011, 285-366
12. S. I. Walker and P. C.W. Davies, “The Algorithmic Origins of Life,” Journal of the Royal Society Interface, 2012, 20120869
13. G. F. Joyce, “Booting up Life,” Nature, 2002, 278-279
14. P. H. Rampelotto, “PANSPERMIA: A PROMISING FIELD OF RESEARCH,” Astrobiology Science Conference, 2010
15. A. Mann, “Bashing Holes in the Tale of Earth’s Troubled Youth,” Nature, 2018, 393-395
16. C. H. Lineweaver and M. D. Norman, “The Bombardment History of the Moon and the Origin of Life on Earth,” CD,(Ed) National Space Society Of, 2009
17. P. H. Rampelotto, “Resistance of Microorganisms to Extreme Environmental Conditions and Its Contribution to Astrobiology,” Sustainability, 2010, 1602-1623
18. F. H.C. Crick and L. E. Orgel, “Directed Panspermia,” Icarus, 1973, 341-346
19. A. I. Oparin, “Evolution of the Concepts of the Origin of Life, 1924-1974,” Origins of Life, 1976, 3-8
20. J. B. S. Haldane, “The Orgin of Life,” Rationalist Annual, 1929, 735-739

21. D. Trail, E. B. Watson, and N. D. Tailby, “The Oxidation State of Hadean Magmas and Implications for Early Earth’s Atmosphere,” Nature, 2011, 79-82
22. S. L. Miller, “A Production of Amino Acids under Possible Primitive Earth Conditions,” Science, 1953, 528-529
23. S. L. Miller, “The Mechanism of Synthesis of Amino Acids by Electric Discharges,” BBA - Biochimica et Biophysica Acta, 1957, 480-489
24. J. L. Bada, “New Insights into Prebiotic Chemistry from Stanley Miller’s Spark Discharge Experiments,” Chemical Society Reviews, 2013, 2186
25. E. T. Parker et al., “Primordial Synthesis of Amines and Amino Acids in a 1958 Miller H2S-Rich Spark Discharge Experiment,” Proceedings of the National Academy of Sciences 108, 2011, 5526-5531
26. J. Oró, “Mechanism of Synthesis of Adenine from Hydrogen Cyanide under Possible Primitive Earth Conditions,” Nature, 1961, 1193-1194
27. A. N. Simonov et al., “The Nature of Autocatalysis in the Butlerov Reaction,” Kinetics and Catalysis, 2007, 245-254
28. R. Breslow, “On the Mechanism of the Formose Reaction,” Tetrahedron Letters, 1959, 22-26
29. C. Reid and L. E. Orgel, “Synthesis of Sugars in Potentially Prebiotic Conditions,” Nature , 1967, 455-455
30. R. Dahm, “Friedrich Miescher and the Discovery of DNA,” Developmental Biology, 2005, 274-288
31. J.D. Watson and F.H.C. Crick, “Reprint: Molecular Structure of Nucleic Acids,” Annals of Internal Medicine, 2013, 581
32. L. Tymoczko, J. L. Berg, J. M. Stryer, Biochemistry. 5th Edition, Section 26.4 Important Derivatives of Cholesterol Include Bile Salts and Steroid Hormones, 2002
33. J. D. Sutherland, “The Origin of Life - Out of the Blue,” Angewandte Chemie International Edition, 2016, 104-121
34. C. G. Kurland, “The RNA Dreamtime,” BioEssays, 2010, 866-871
35. G. F. Joyce, “The Antiquity of RNA-Based Evolution”, Nature, 2002, 214-221
36. S. Altman, “Enzymatic Cleavage of RNA by RNA (Nobel Lecture),” Angewandte Chemie International Edition, 1990, 749-758
37. T. R. Cech, “A Model for the RNA-Catalyzed Replication of RNA.,” Proceedings of the National Academy of Sciences, 1986, 4360-4363
38. B. Zhang and T. R. Cech, “Peptide Bond Formation by in Vitro Selected Ribozymes,” Nature, 1997, 96-100
39. P. J. Unrau and D. P. Bartel, “RNA-Catalysed Nucleotide Synthesis,” Nature, 1998, 260-263
40. W. K. Johnston et al., “RNA-Catalyzed RNA Polymerization: Accurate and General RNA-Templated Primer Extension,” Science, 2001, 1319-1325
41. S. A. Benner, “Paradoxes in the Origin of Life,” Origins of Life and Evolution of Biospheres, 2014, 339-343

42. M. W. Powner, B. Gerland, and J. D. Sutherland, “Synthesis of Activated Pyrimidine Ribonucleotides in Prebiotically Plausible Conditions,” Nature, 2009, 239-242
43. J. S. Teichert, F. M. Kruse, O. Trapp, “Direct Prebiotic Pathway to DNA Nucleosides”, Angewandte Chemie International Edition, 2019, 9944-9947
44. G. Wächtershäuser, “Before Enzymes and Templates: Theory of Surface Metabolism.,” Microbiological Reviews, 1988, 452–484
45. C. Huber and G. Wächtershäuser, “α-Hydroxy and α-Amino Acids under Possible Hadean, Volcanic Origin-of-Life Conditions,” Science, 2006, 630-632
46. C. Huber, “Peptides by Activation of Amino Acids with CO on (Ni,Fe)S Surfaces: Implications for the Origin of Life,” Science, 1998, 670-672
47. D. S. Kelley et al., “An Off-Axis Hydrothermal Vent Field near the Mid-Atlantic Ridge at 30° N,” Nature, 2001, 145-149
48. John B. Corliss et al., “Submarine Thermal Springs on the Galápagos Rift,” Science, 1979, 1073-1083
49. W. Martin et al., “Hydrothermal Vents and the Origin of Life.,” Nature Reviews. Microbiology , 2008, 805-814
50. M. Preiner et al., “Serpentinization: Connecting Geochemistry, Ancient Metabolism and Industrial Hydrogenation,” Life, 2018, 41
51. S. J. Varma et al., “Metals Enable a Non-Enzymatic Acetyl CoA Pathway,” BioRxiv, 2017, 235523
52. K. B. Muchowska, S. J. Varma, and J. Moran, “Synthesis and Breakdown of Universal Metabolic Precursors Promoted by Iron,” Nature, 2019, 104
53. K. B. Muchowska, E. Chevallot-Beroux, and J. Moran, “Recreating Ancient Metabolic Pathways before Enzymes,” Bioorganic and Medicinal Chemistry, 2019, 2292-2297
54. D. M. Lorenz, A. Jeng, and M. W. Deem, “The Emergence of Modularity in Biological Systems.,” Physics of Life Reviews, 2011, 129-160
55. L. Cronin, “Reaction: A New Genesis for Origins Research?,” Chem, 2017, 601-603
56. P. W. Anderson, “More Is Different.,” Science, 1972, 393-396
57. P. J. T. Morris, The Matter Factory: A History of the Chemistry Laboratory, 2016
58. K. Olsen, “The First 110 Years of Laboratory Automation: Technologies, Applications, and the Creative Scientist,” Journal of Laboratory Automation 17, 2012, 469-480
59. R. B. Merrifield, “Automated Synthesis of Peptides,” Science, 1965, 178-85
60. R. B. Merrifield, “Solid Phase Peptide Synthesis. I. The Synthesis of a Tetrapeptide,” Journal of the American Chemical Society, 1963, 2149-2154
61. R. B. Merrifield, John Morrow Stewart, and Nils Jernberg, “Instrument for Automated Synthesis of Peptides,” Analytical Chemistry, 1966, 1947-51
62. B. Gutte and R. B. Merrifield, “The Total Synthesis of an Enzyme with Ribonuclease A Activity,” Journal of the American Chemical Society, 1969, 501-2

63. R. B. Merrifield “Automated Synthesis of Peptides.”, Hypotensive Peptides, 1966, 1947-51
64. H. Fleischer, K. Thurrow, Automation Solutions for Analytical Measurements: Concepts and Applications, 2017
65. H. R. Watkins and S. Palkin, “Automatic Devices for Extracting Alkaloidal Solutions*,” The Journal of the American Pharmaceutical Association, 1912, 612-614
66. J. J. Lingane, “Automatic Potentiometric Titrations,” Analytical Chemistry, 1948, 33-39
67. S. M. Martin, “Automatic Starter for Chromatograms,” Analytical Chemistry, 1958, 1890-1890
68. A. G. Godfrey, T. Masquelin, and H. Hemmerle, “A Remote-Controlled Adaptive Medchem Lab: An Innovative Approach to Enable Drug Discovery in the 21st Century,” Drug Discovery Today, 2013, 795-802
69. G Alvarado-Urbina et al., “Automated Systems of Gene Fragments,” Science, 1981, 270-4
70. H. Gobind Khorana, Some Recent Developments in the Chemistry of Phosphate Esters of Biological Interest, John Wiley & Sons, Inc, 1961
71. R. L. Letsinger and K. K. Ogilvie, “Nucleotide Chemistry. XIII. Synthesis of Oligothymidylates via Phosphotriester Intermediates,” Journal of the American Chemical Society, 2005, 773-786
72. R. L. Letsinger et al., “Phosphite Coupling Procedure for Generating Internucleotide Links,” Journal of the American Chemical Society, 1975, 3278-3279
73. M. D. Matteucci and M. H. Caruthers, “The Synthesis of Oligodeoxyprimidines on a Polymer Support,” Tetrahedron Letters, 1980, 719-722
74. S. J. Horvath et al., “An Automated DNA Synthesizer Employing Deoxynucleoside 3′-Phosphoramidites,” Methods in Enzymology, 1987, 314-326
75. J. M. S. Bartlett and D. Stirling, 226-PCR Protocols, 2nd Edition, vol. 226, 2003
76. R. Dessy, “Robots in the Laboratory: Part 1,” Analytical Chemistry 55, 1983
77. A. R. Frisbee et al., “Robotic Orchestration of Organic Reactions: Yield Optimization via an Automated System with Operator-Specified Reaction Sequences,” Journal of the American Chemical Society, 1984, 7143-7145
78. D. F. Emiabata-Smith, Derek L. Crookes, and Martin R. Owen, “A Practical Approach to Accelerated Process Screening and Optimisation,” Organic Process Research and Development 3, 1999, 281-288
79. O. J. Plante, E. R. Palmacci, and P. H. Seeberger, “Automated Solid-Phase Synthesis of Oligosaccharides,” Science, 2001, 1523-1527
80. J. M. Fréchet and Conrad Schuerch, “Solid-Phase Synthesis of Oligosaccharides. II. Steric Control by C-6 Substituents in Glucoside Syntheses,” Journal of the American Chemical Society, 1972, 604-609
81. P. H. Seeberger, “Automated Carbohydrate Synthesis to Drive Chemical Glycomics,” Chemical Communications, 2003, 1115-1121

82. K. Naresh et al., “Pushing the Limits of Automated Glycan Assembly: Synthesis of a 50mer Polymannoside,” Chemical Communications 53, 2017, 9085-9088
83. M. B. Plutschack et al., “The Hitchhiker’s Guide to Flow Chemistry”, Chemical Reviews 117, 2017, 11796-11893
84. T. Takahashi et al., “Development and Application of a Solution-Phase Automated Synthesizer, ‘ChemKonzert,’” Chemical & Pharmaceutical Bulletin 58, 2010
85. J. Li et al., “Automated Process,” Organic Synthesis 347, 2015, 1221-1226
86. E. P. Gillis and Martin D. Burke, “A Simple and Modular Strategy for Small Molecule Synthesis: Iterative Suzuki-Miyaura Coupling of B-Protected Haloboronic Acid Building Blocks,” Journal of the American Chemical Society, 2007, 6716-6717
87. J. W. Lehmann, D. J. Blair, and M. D. Burke, “Toward Generalized Iterative Small Molecule Synthesis,” Nat Rev Chem, 2018, 436-440
88. R. Service, “Billion-Dollar Project Would Synthesize Hundreds of Thousands of Molecules in Search of New Medicines,” Science, 2017, 1073
89. K. Sanderson, “AUTOMATION : CHEMISTRY SHOOTS FOR THE MOON,” Nature, 2019
90. J. M. Granda et al., “Controlling an Organic Synthesis Robot with Machine Learning to Search for New Reactivity,” Nature, 2018, 67-69
91. D. Caramelli et al., “Networking Chemical Robots for Reaction Multitasking,” Nature Communications, 2018, 3406
92. S. Steiner et al., “Organic Synthesis in a Modular Robotic System Driven by a Chemical Programming Language,” Science, 2019, 363
93. S. M. Marshall, Alastair R. G. Murray, and Leroy Cronin, “A Probabilistic Framework for Identifying Biosignatures Using Pathway Complexity,” Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 375, 2017, 342
94. C. E Shannon and warren weaver, “The Mathematical Theory Of communication,” THE UNIVERSITY OF ILLINOIS PRESS, 1949

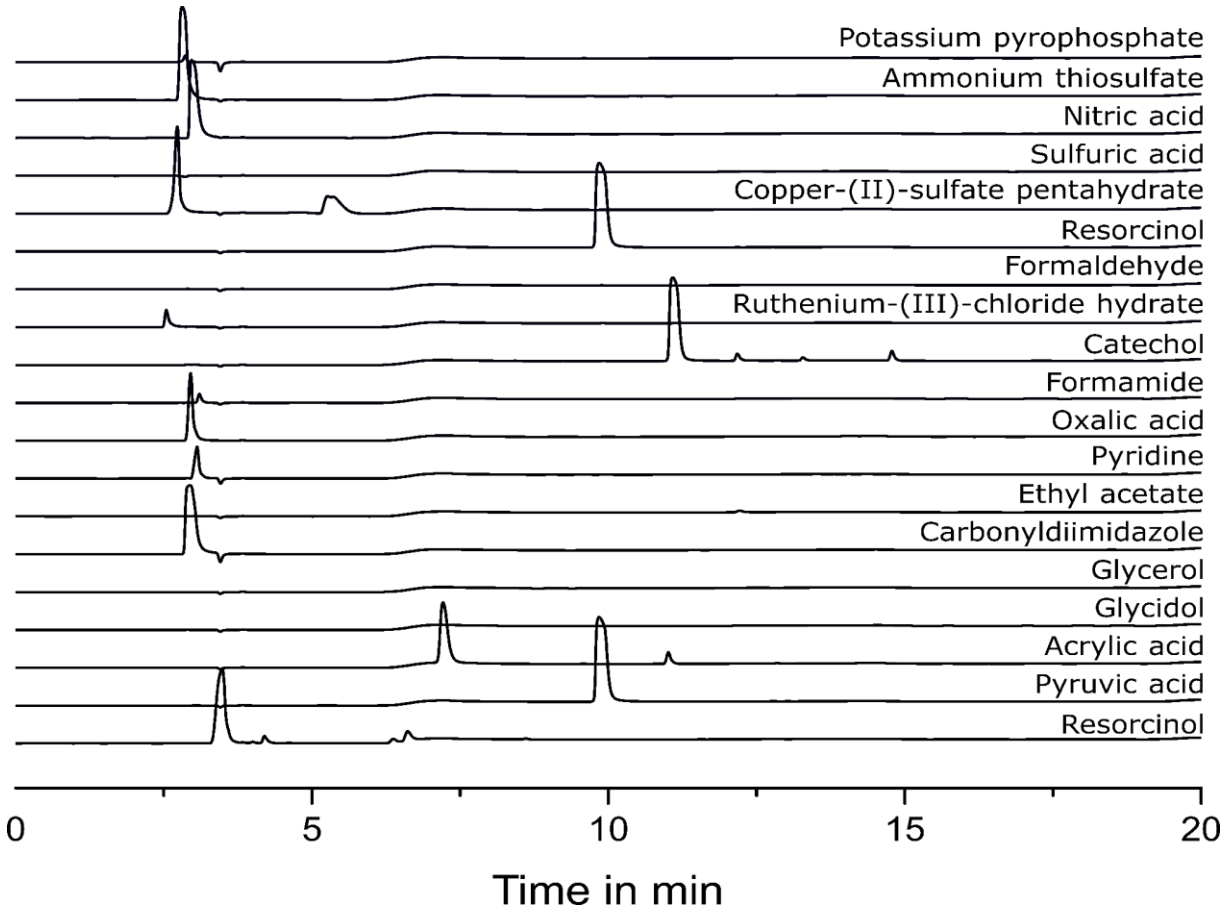
Appendix

Appendix Table1: List of executed runs, their cycle number and their input solution in order of addition to reactor

RUN	CYCLE NUMBER	INPUT 1	INPUT 2	INPUT 3
A	1 to 17	resorcinol	pyruvic acid	ammonium thiosulfate
	18 to 26	copper-(II)-sulfate pentahydrate	oxalic acid	nitric acid
	27 to 36	copper-(II)-sulfate pentahydrate	oxalic acid	carbonyldiimidazole
	37 to 47	pyridine	ruthenium-(III)-chloride hydrate	ammonium thiosulfate
	48 to 59	pyruvic acid	resorcinol	catechol
	60 to 78	ammonium thiosulfate	carbonyldiimidazole	formamide
	79 to 88	pyruvic acid	formaldehyde	glycidol
	89 to 98	formaldehyde	acrylic acid	sulfuric acid
	99 to 110	carbonyldimidazole	pyridine	ruthenium-(III)-chloride hydrate
	111 to 113	ruthenium-(III)-chloride hydrate	sulfuric acid	ethyl acetate
B/ C/ D	1 to 17	acrylic acid	potassium pyrophosphate	carbonyldiimidazole
	18 to 42	ethyl acetate	formamide	formaldehyde
	43 to 52	ruthenium-(III)-chloride hydrate	pyridine	copper-(II)-sulfate pentahydrate
	53 to 62	resorcinol	pyridine	formamide
	63 to 67	potassium pyrophosphate	ethyl acetate	formaldehyde
E	1 to 13	glycidol	pyruvic acid	sulfuric acid
	14 to 20	glycidol	resorcinol	catechol
	21 to 30	oxalic acid	sulfuric acid	acrylic acid
	31 to 43	sulfuric acid	potassium pyrophosphate	acrylic acid

	44 to 52	oxalic acid	formaldehyde	sulfuric acid
	53 to 70	acrylic acid	pyridine	sulfuric acid
	71 to 80	ammonium thiosulfate	oxalic acid	sulfuric acid
	81 to 92	ruthenium-(III)-chloride hydrate	carbonyldiimidazole	formamide
	93 to 100	ruthenium-(III)-chloride hydrate	potassium pyrophosphate	ethyl acetate
	101 to 112	nitric acid	oxalic acid	sulfuric acid
	113 to 125	ruthenium-(III)-chloride hydrate	potassium pyrophosphate	ruthenium-(III)-chloride hydrate
	126 to 129	ruthenium-(III)-chloride hydrate	potassium pyrophosphate	catechol
	130 to 160	ruthenium-(III)-chloride hydrate	potassium pyrophosphate	resorcinol
F	1 to 1	pyruvic acid	ethyl acetate	oxalic acid
	13 to 20	acrylic acid	copper-(II)-sulfate pentahydrate	sulfuric acid
	21 to 43	glycidol	carbonyldiimidazole	oxalic acid
	44 to 56	ethyl acetate	pyruvic acid	ethyl acetate
	57 to 71	copper-(II)-sulfate pentahydrate	acrylic acid	glycidol
G	1 to 10	oxalic acid	pyridine	catechol
	11 to 21	catechol	nitric acid	pyruvic acid
	22 to 34	glycerol	ammonium thiosulfate	formamide
	35 to 44	acrylic acid	potassium pyrophosphate	glycidol
	45 to 56	copper-(II)-sulfate pentahydrate	sulfuric acid	ammonium thiosulfate
	57 to 67	glycerol	ammonium thiosulfate	catechol
	68 to 76	glycidol	pyridine	pyruvic acid
H	1 to 12	glycidol	ammonium thiosulfate	carbonyldimidazole
	13 to 22	glycidol	formaldehyde	glycidol

	23 to 36	ammonium thiosulfate	copper-(II)-sulfate pentahydrate	catechol
	37 to 65	sulfuric acid	resorcinol	pyruvic acid
	66 to 80	sulfuric acid	ruthenium-(III)-chloride hydrate	carbonyldiimidazole
	81 to 90	oxalic acid	glycerol	resorcinol
	91 to 96	potassium pyrophosphate	catechol	resorcinol



Appendix Figure 1: Input solutions run on their own on HPLC-DAD

```
def get_df(df_file):
    #opens csv file and reads it in as a df
    my_df = pd.read_csv(df_file)
    list(my_df.columns.values)
    #set header right
    my_df.columns = ["rts", "mz", "int"]

    #adds another column to the dataframe with the intensity substracted by our set threshold
    #our threshold is the average threshold with the standard deviation divided by two added
    threshold = ((my_df.loc[:, "int"].median()) +  (my_df.loc[:, "int"].std()) / 2)
    thresh_df = my_df["int"] - threshold
    #add thresholding column
    my_df["threshold"] = thresh_df
    my_df[my_df < 0] = 0


    #this adds a column to the dataframe where we multiply mz mass with the thresholded intensity
    mz = my_df["mz"]
    t_int = my_df["threshold"]
    multi = mz * t_int
    my_df["multi"] = multi
    return(my_df)

def min_number(my_df):
    #this makes a list with the minimum multiplied value of every spectra for that run
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    list_min_number = []
    for i in range(n_rt):
        new_df = my_df[my_df["rts"]==unique_rts[i]]
        for min_number in new_df["multi"]:
            if min_number == 0:
                pass
            else:
                list_min_number.append(min_number)
                break
    return(list_min_number)

def max_number(my_df):
    #this makes a list with the maximum multiplied value of every spectra for that run
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    list_max_number = []
    for i in range(n_rt):
        new_df = my_df[my_df["rts"]==unique_rts[i]]
        rev_df = new_df.reindex(index=new_df.index[::-1])
        for max_number in rev_df["multi"]:
            if max_number == 0:
                pass
            else:
                list_max_number.append(max_number)
                break
    return(list_max_number)
def peak_counter(my_df):
    #this makes a list of the number of peaks of every spectra for that run
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    peak_count = []
    peaks = 0
    for i in range(n_rt):
        new_df = my_df[my_df["rts"]==unique_rts[i]]
        for number in new_df["threshold"]:
            if number > 0:
                peaks += 1
        peak_count.append(peaks)
    return(peak_count)

def mz_index_calculation(my_df):
    #this calculates the actual mz index value
    list_max_number = max_number(my_df)
    list_min_number = min_number(my_df)
    peak_count = peak_counter(my_df)

    #first substract the minimum number from the maximum for every value in both lists
    a = matrix(list_max_number)
    b = matrix(list_min_number)
    sub = a - b
    #we convert the substraction list into an array that we can devide later
    sub_array = array(sub)
    #turns peak count into an array
    peak_array = array(peak_count)
    #divides the substracted min and max values through the number of peaks
    mz_list = np.divide(sub_array, peak_array)
    #takes the average of the total mz calculation
    mz_value = np.average(mz_list)
    return(mz_value)

def main():

    my_df = get_df("Machine_R19__1_Scan1_is1.csv")
    mz_value = mz_index_calculation(my_df)
    print(mz_value)

if __name__ =="__main__":
    main()
```

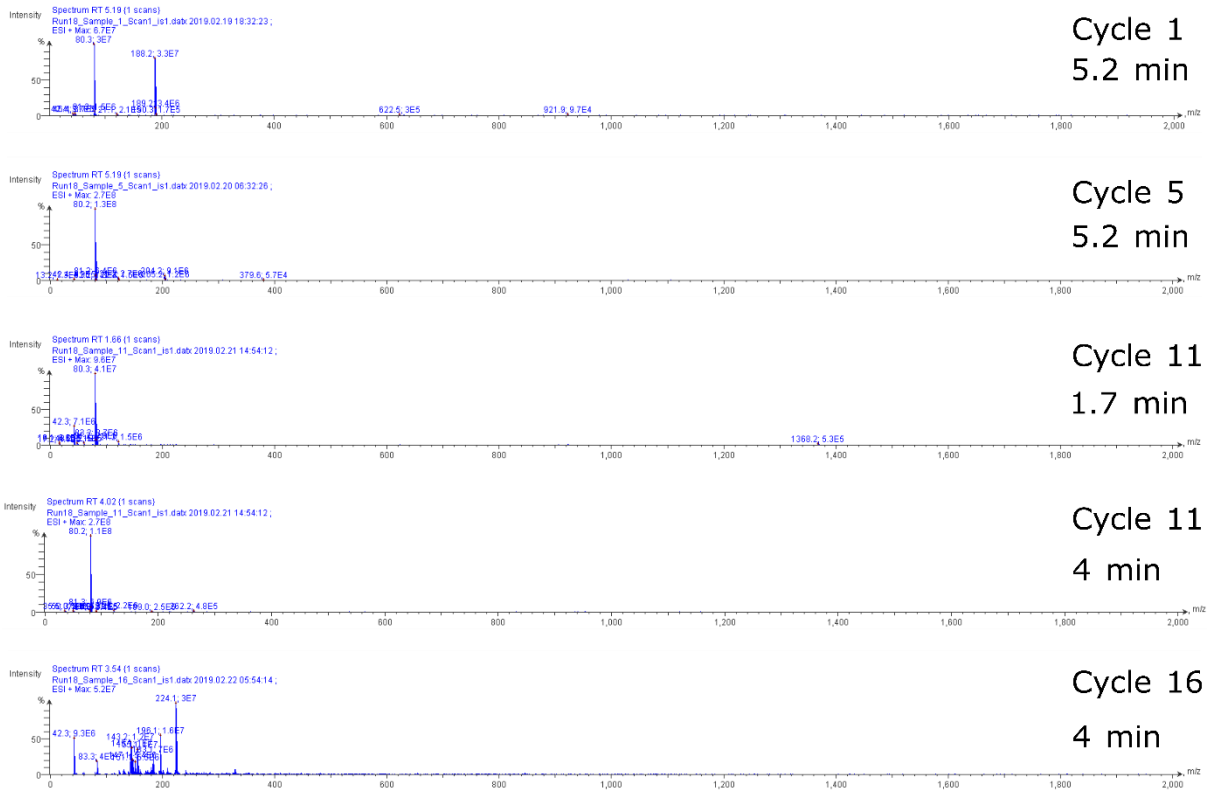
Appendix Figure 2: Code for modified mz index

```
import numpy as np
from numpy import array
from numpy import matrix
import pandas as pd
import statistics
import csv
import os
import filetools
import inspect
import matplotlib.pyplot as plt
%matplotlib inline

#this opens the folder with our csv files
HERE = os.path.dirname(os.path.abspath(inspect.getfile(inspect.currentframe())))

def get_files_to_run(directory):
    return [file for file in filetools.list_files(directory) if ".csv" in file]

def get_df(df_file):
    #opens specfific csv file and reads it in as a df
    my_df = pd.read_csv(df_file)
    list(my_df.columns.values)
    #set header right
    my_df.columns = ["rts", "mz", "int"]


    threshold = ((my_df.loc[:, "int"].median()) +  (my_df.loc[:, "int"].std()) / 2)

    thresh_df = my_df["int"] - threshold
    #add thresholding column
    my_df["threshold"] = thresh_df
    my_df[my_df < 0] = 0
    return(my_df)

def normalise(my_df):
    #we are iterating through every retention time group
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    #add another line into our df with normalised intensities ip, currently just with zeros stored
    my_df["normed_I"] = np.zeros(len(my_df["int"]))
    #want to get intensities for each spectra normalized by the total intensity for that spectra
    #(e.g. the total intensity for each rt should be = 1)
    for i in range(n_rt):
        #iterates through every retention time group
        rt_df = my_df[my_df["rts"]==unique_rts[i]]
        #sums up all intensities in a rt group
        T = sum(rt_df["int"])
        #divides the sum of all intensities T through each intensity in our df
        rt_df.loc[:,"normed_I"] = rt_df.loc[:,"int"]/T
        #adds or rt_df into our main df as our normed_I column
        my_df[my_df["rts"]==unique_rts[i]] = rt_df
    return(my_df)

def multiply(my_df):
    # add multiplied mz with threshold intensity
    mz = my_df["mz"]
    int_norm = my_df["normed_I"]

    multi_normed = mz * int_norm
    my_df["multi_normed"] = multi_normed
    return(my_df)

def sum_it_up(my_df):
    #sums the multi column and returns the multi sum
    total_sum = my_df["multi_normed"].sum()
    print(total_sum)
    return(total_sum)

def output_csv(total_sum):
    output = total_sum
    return(output)


def save_to_csv(filename, output):
    #should output the mz_value and the filename in a csv file
    with open(filename, "w") as csvFile:
        outputwriter = csv.writer(csvFile, delimiter=",")
        outputwriter.writerow(["File", "Mass_Intensity_Sum"])
        for row in output:
            outputwriter.writerow(row)
def main():
    files_to_run = get_files_to_run(os.path.join(HERE, "input_csv"))
    out_rows = []
    for csv_file in files_to_run:
        my_df = get_df(csv_file)
        my_df = normalise(my_df)
        my_df = multiply(my_df)
        total_sum = sum_it_up(my_df)
        out_rows.append([csv_file, total_sum])
    save_to_csv("Output_weight_by_normed_intensity.csv", out_rows)
    print("calculation finished")

if __name__ =="__main__":
    main()
```

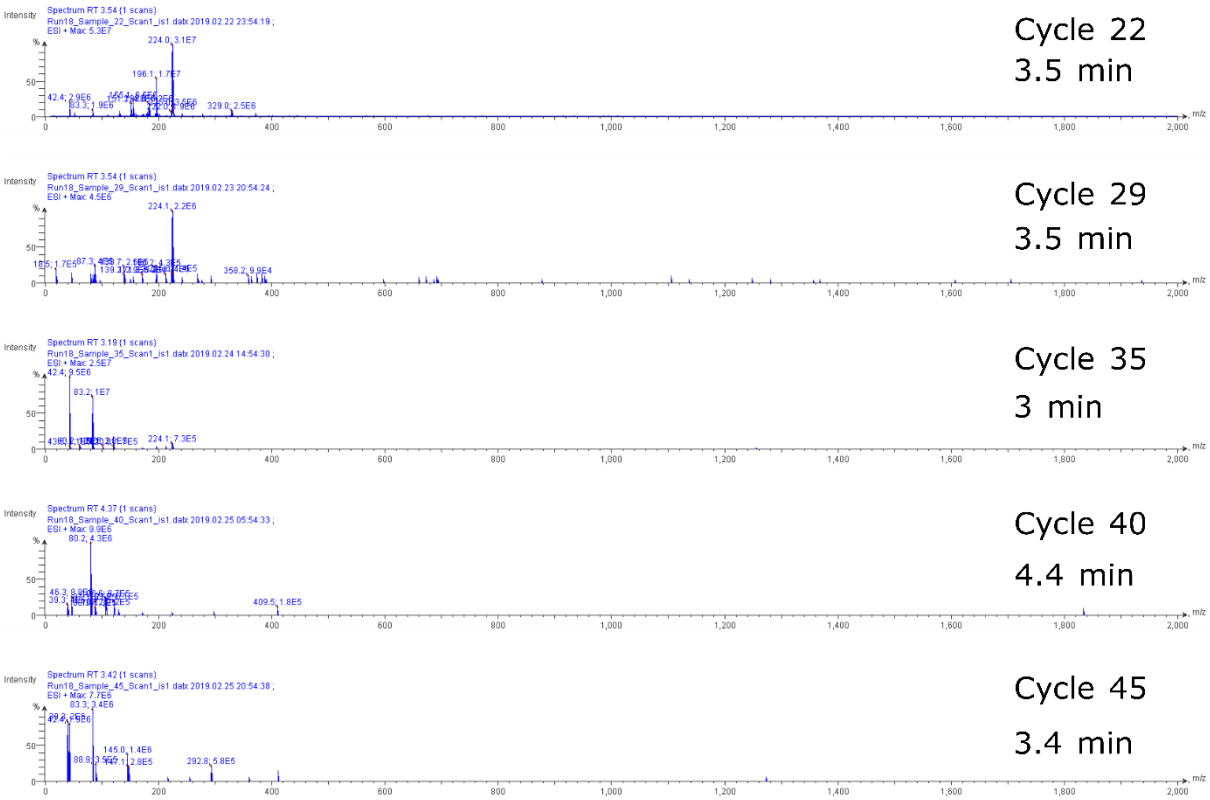
Appendix Figure 3: Code for normalised weight by intensity

```
import numpy as np
from numpy import array
from numpy import matrix
import pandas as pd
import statistics
import csv
import os
import filetools
import inspect
import matplotlib.pyplot as plt
%matplotlib inline

#this opens the folder with our csv files
HERE = os.path.dirname(os.path.abspath(inspect.getfile(inspect.currentframe())))

def get_files_to_run(directory):
    #returns list of filenames in file folder
    return [file for file in filetools.list_files(directory) if ".csv" in file]

def get_df(df_file):
    #opens specfific csv file and reads it in as a dataframe
    my_df = pd.read_csv(df_file)
    list(my_df.columns.values)
    #set header right
    my_df.columns = ["rts", "mz", "int"]

    #setting the threshold
    threshold = ((my_df.loc[:, "int"].median()) +  (my_df.loc[:, "int"].std()) / 2)
    #creating a new dataframe for the thresholded values
    thresh_df = my_df["int"] - threshold
    #add thresholding column
    my_df["threshold"] = thresh_df
    my_df[my_df < 0] = 0
    return(my_df)

def normalise(my_df):
    #we are iterating through every retention time group and sort the df based on
    #retention times
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    #we add another line into our df with normalised intensities (ip), currently just with zeros stored
    my_df["normed_I"] = np.zeros(len(my_df["int"]))
    #We want to get intensities for each spectra normalized by the total intensity for that spectra (e.g. the total intensity for each rt should  = 1)
    for i in range(n_rt):
        #iterates through every retention time group
        rt_df = my_df[my_df["rts"]==unique_rts[i]]
        #sums up all intensities in a rt group
        T = sum(rt_df["int"])
        #divides the sum of all intensities T through each intensity in our df
        rt_df.loc[:,"normed_I"] = rt_df.loc[:,"int"]/T
        #adds or rt_df into our main df as our normed_I column
        my_df[my_df["rts"]==unique_rts[i]] = rt_df
    return(my_df)

def calculate_entropy(my_df):
    #we are iterating through every retention time group
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    #this adds a column to the dataframe that has the ln of ip stored
    my_df["log_ip"] = np.log(my_df["normed_I"])
    #this adds a column that has the ln of ip multiplied to ip stored
    my_df["entropy"] = my_df["log_ip"] * my_df["normed_I"]
    #create an empty list
    entropy = []
    #we are iterating through every retention time group again
    unique_rts = sorted(list(set(my_df["rts"])))
    n_rt = len(unique_rts)
    for i in range(n_rt):
        #iterates through every retention time group
        rt_df = my_df[my_df["rts"]==unique_rts[i]]
        #sums up the entropy for a spectra by summing it up for a retention time group and
        #stores it into our entropy list
        S = sum(rt_df["entropy"])
        entropy.append(S)
    #takes the average of the entropy of all spectras
    total = statistics.mean(entropy)
    print(total)
    return(total)
def output_csv(total):
    #just writes our output as output
    output = total
    return(output)

def save_to_csv(filename, output):
    #outputs the mz_value and the filename in a csv file
    with open(filename, "w") as csvFile:
        outputwriter = csv.writer(csvFile, delimiter=",")
        outputwriter.writerow(["File", "Entropy"])
        for row in output:
            outputwriter.writerow(row)

def main():
    #calls all our functions above
    files_to_run = get_files_to_run(os.path.join(HERE, "input_csv"))
    out_rows = []
    for csv_file in files_to_run:
        my_df = get_df(csv_file)
        norm_data = normalise(my_df)
        total = calculate_entropy(my_df)
        out_rows.append([csv_file, total])
    save_to_csv("Output_Entropy.csv", out_rows)
    print("calculation finished")

if __name__ =="__main__":
    main()
```

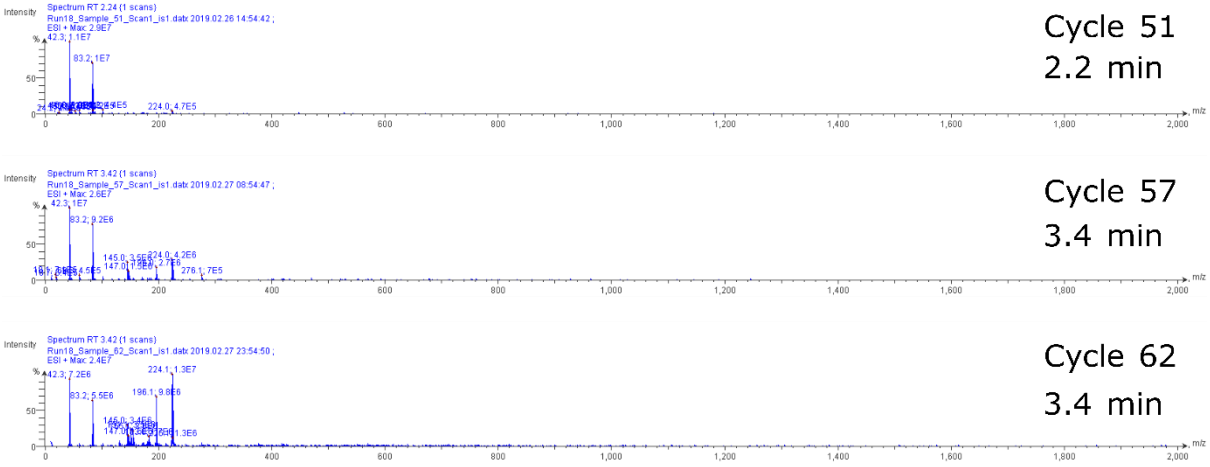
Appendix Figure 4: Code for entropy



Appendix Figure 5: Chosen spectra of run G. Shown in this figure is cycle 1 at 5.2 min, cycle 5 at 5.2 min, cycle 11 at 1.7 min and 4 min and cycle 16 at 4 min. The peaks at 42 m/z and 80 m/z are related to the formic acid mobile phase and can be seen in blank spectra too.



Appendix Figure 6: Chosen spectra of run G. Shown in this figure is cycle 22 at 3.5 min, cycle 29 at 3.5 min, cycle 35 at 3 min, cycle 40 at 4.4 min and cycle 45 at 3.4 min. The peaks at 42 m/z and 80 m/z are related to the formic acid mobile phase and can be seen in blank spectra too.



Appendix Figure 7: Chosen spectra of run G. Shown in this figure is cycle 51 at 2.2 min, cycle 57 at 3.4 min and cycle 62 at 3.4 min. The peaks at 42 m/z and 80 m/z are related to the formic acid mobile phase and can be seen in blank spectra too.