Bachelor Thesis

Use of a ligand containing a pyrazole functionality and two carboxylate groups to construct new MOFs for applications in gas storage

Marton Stefano

July 2023

Abbreviations

DikDiEst Dimethyl 4,4′-malonyldibenzoate

DikDiAc 4,4′-malonyldibenzoic acid

PyrDiEst Methyl 4,4′-(1H-pyrazole-3,5-diyl)dibenzoate

PyrDiAc Dimethyl 4,4′-malonyldibenzoate

UnsDiEst Methyl 4-[(1E)-3-[4-(methoxycarbonyl) phenyl]-3-oxoprop-1-en-1-yl]benzoate

DikDiCN Dimethyl 4,4′-malonyldibenzoate

Contents

List of Figures

Summary

CHAPTER 1: INTRODUCTION

Over the past 150 year supramolecular chemistry has grown at an exponential rate, with the field expanding to embrace a wide range of applications, in what can be the described as a new chemical space. This chapter aims to provide a brief historical account of the most notable development in supramolecular chemistry, focusing on MOFs and 3D periodic structures as well as their implementation in gas adsorption and electrochemistry applications.

CHAPTER 2: RESULT AND DISCUSSION

The construction of 3D periodic solid structure oriented towards specific gas adsorption remain a challenging synthetic problem for chemists, several problems related to their development and use must be accounted. In this chapter will be reported the advancement in the synthesis of a pyrazole based ligand and the utilization of a 1,3-diketones ligand in electrochemistry oriented application.

CHAPTER 3: EXPERIMENTAL SECTION

Several synthesis and characterization method as been applied, both in the synthesis of pyrazole based ligand and 1,3-diketones MOF electrochemistry exploration, in order to achieve the results discussed earlier. They will be analyzed in great detail in this chapter.

CHAPTER 4: TOOL DEVELOPMENT

During the time spent working on these wide range topics the need of small and reliable analytical tools raised. Being able to analyze and draw a comparison between obtained results in a fast manner has been very important. Across this chapter will be presented a tiny python implementation for fast data analysis and graphs creation based on matplotlib.

1 Introduction

1.1 Supramolecular Chemistry, why?

Supramolecular chemistry can be classified as the branch of chemistry concerned about the interplay between designed molecular assemblies and intermolecular bonds, or more colloquially referred to as "chemistry beyond the molecule"[?].

The discipline focuses on the design and synthesis of molecular architectures by relying on the complementary recognition, and subsequent assembly, of well-defined subunits. The products of complementary synthesis, the so-called "supermolecules" are sustained by non-covalent interactions such as hydrogen bonding, halogen bonding, coordination forces and $\pi - \pi$ interactions.

The emergence of supramolecular chemistry has directly influenced how efficiently chemists can design and synthesize desired frameworks. The development and application of the bottom up approach is widely successful, owing to the non-covalent forces that dictate structural and morphological properties, while producing structures that were previously inaccessible.

1.1.1 Host-Guest Chemistry

In supramolecular chemistry, host–guest chemistry describes complexes that are composed of two or more molecules or ions that are held together in unique structural relationships by forces other than those of full covalent bonds. Host–guest chemistry encompasses the idea of molecular recognition and interactions through non-covalent bonding. Non-covalent bonding is critical in maintaining the 3D structure of large molecules.

Host-guest interaction has raised dramatic attention since it was discovered. It is an important field because many biological processes require the host-guest interaction, and it can be useful in material designs.

1.1.2 Metal-Organic Frameworks

Metal-Organic Frameworks represent since 1995 [?] an exciting and rapidly growing area of research within the field of supramolecular chemistry.

MOFs' chemistry is a specific type of supramolecular chemistry that involve the coordination of metal ions with organic ligands to form highly porous and crystalline materials with a unique structure.

MOFs are obtained from the reaction between an organic linker and inorganic species, such as clusters or metal ions. The metal ions act as nodes that are connected by the organic ligands to create a three-dimensional framework. By understanding and tweaking the features of this two components within the material, it is possible to predict and tune the chemical and physical properties of the framework.

This offers a wide range of tunable properties that depend both from the inorganic joints and the organic linkers. By looking closer at the organic linker, it is possible to imagine a tuning of the

shape (e.g. ligand coordination angles), size (e.g. dimension of the ligand as expansion or shrink-age) but also, functionality (e.g. decoration of the pores depending on the sub-stituent of the organic unit) of the MOF.

The self assembly of these components, typically done in solution, generate a linkage extended in a 3D crystalline framework that is characterized by large pores filled of solvent. Interesting, this rigid structure can stand vacuum: after removal of the solvent molecules the structure usually remains intact.

The resulting structure has a large internal surface area and can adsorb gases and other molecules with high efficiency, making MOFs useful in a variety of applications, such as gas storage, catalysis, drug delivery and molecule harvesting.

1.1.3   Carboxylate–Based Metal–Organic Frameworks

Carboxylate ligands are tipically obtained from the acid-base reaction of a carboxilic acid with an appropriate base, their electronic structure determine the properties that follows.
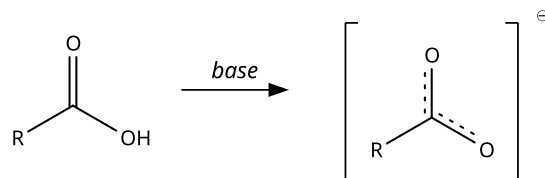
Figure 1.1: Carboxilate structure

The carboxylate oxygen atoms can coordinate with metal ions through their lone pairs of electrons, forming metal-carboxylate coordination bonds. The metal-carboxylate bonds formed by carboxylate ligands are generally strong due to the high electronegativity of oxygen and the polarizability of the carboxylate group. The binding strength can be influenced by the nature of the metal ion, the electronic properties of the ligand, and the coordination geometry. Strong metal-carboxylate bonds contribute to the stability of the MOF framework.

The carboxylate group that can serve in multiple coordination mode although the chelating coordination mode is typically adopted: the ligand coordinates to the metal ion through both oxygen atoms.

Figure 1.2: Carboxilate coordination modes

Their geometric arrangement and flexibility can vary. The length and conformation of the carbon backbone connecting the carboxylate groups can affect the spatial arrangement of the ligand and rigidity or flexibility of the ligand can impact the packing arrangement and the resulting MOF

structure.

Carboxylate ligands can also incorporate various functional groups, such as aromatic rings, electron-withdrawing or electron-donating groups, and functional moieties (e.g., -OH, -NH2). These functional groups can impart additional reactivity and chemical properties to the MOF, enabling specific applications. The presence of functional groups can also affect the hydrophobic or hydrophilic nature of the MOF.

The solubility of carboxylate ligands plays a crucial role in MOF synthesis. Some ligands are readily soluble in common solvents, facilitating the preparation of precursor solutions and the synthesis of MOFs. However, certain carboxylate ligands may exhibit limited solubility, necessitating the use of specialized solvent systems or modifications in reaction conditions to ensure proper ligand incorporation and MOF formation.

The divalent metal carboxylate based frameworks MOF-5 and HKUST-1 are examples of prototypical MOF materials and triggered a huge growth in the field of metal-organic frameworks.

1.1.4   1,3–Diketones–Based Metal–Organic Frameworks

Classical β-diketones and related ligands have been studied for more than a century and their ability to give rise to a rich and interesting coordination and supramolecular chemistry is well documented[**?**]. The widespread use of β-diketones structure as chelating ligand in coordination chemistry and the stability of its complexes with a number of metal ions originate from its chemical properties, determined by the keto–enol tautomerism.
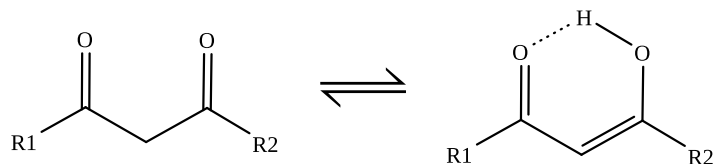

Figure 1.3: β-diketones keto-entol tautomerism

1,3-diketones as ligands

They act under appropriate conditions as uninegative chelating donors, capable of stabilizing mononuclear or polynuclear complexes. One of the reason of such stabilization capability is the aromatic character of the chelate ring.
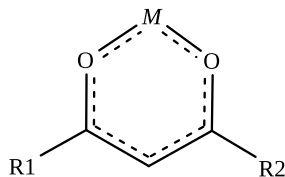

Figure 1.4: Coordination mode of β-diketones

1,3-diketones MOFs examples

1.1.5 Azoles-Based Metal-Organic Framerworks

Heterocycles are a remarkable class of compounds in terms of the biological, chemical and technological roles they play. Azoles, in particular, owe their importance to the fact that they occur in natural and synthetic molecules, but also to their fascinating coordinating chemistry, which has allowed their use in a wide range of applications.

Although azoles are mostly known as bases (i.e. pyridines), the five-membered azoles, including imidazole, pyrazole, triazole, and tetrazole can also be deprotonated to form the corresponding azolate anion. The possibility of generating this azolate anions not only it allows all N atoms to coordinate with metal ions in many different coordination modes, but also further increases the basicity of these donor sites. Consequently, azolate ligands can generate coordination compounds with a particularly high thermal and chemical stability which is one of the most important issues for practical applications of coordination polymers. For these reasones azolates have been widely used as bridging ligands for coordination polymers before the beginning of the past decade.

Pyrazole Ring
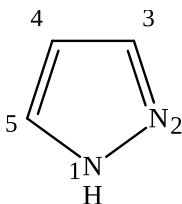
Figure 1.5: Pyrazole structure

Simple 1H-pyrazoles have a Lewis acid N-H group at N_1 and a Lewis basic pyridine N-donor N_2 directly adjacent to each other. Thus, electronically, these heterocycles are both σ-donor and π-acceptor ligands. In addition, they are planar and non-bulky ligands. For these reasons, the pyrazole ring is one of the easiest and most flexible N-donor heterocycles to incorporate into larger polydentate ligand structures.

Pyrazole and only a handful of its derivatives are commercially available, but by using a variety of synthetic approaches it is possible to create substituted pyrazole rings with a wide range of substituents on their carbon atoms.

Pyrazolate as ligands

Pyrazolate have been found to bind metals in a variety of coordination modes and can also act as bridging ligands. After deprotonation of the N-H pyrazole, the obtained pyrazolate anion can be classified to act mainly in three different coordination modes:

- mono-dentate mode
- exo-bidentate $(\eta_1 - \eta_1)$
- endo-bidentate (η_2)

This coordination ability is controlled by the nature of the metal ion and the substituents on the pyrazole ring, infact the substituents at the 3 and 5 positions (shown in 1.5 can modify the steric
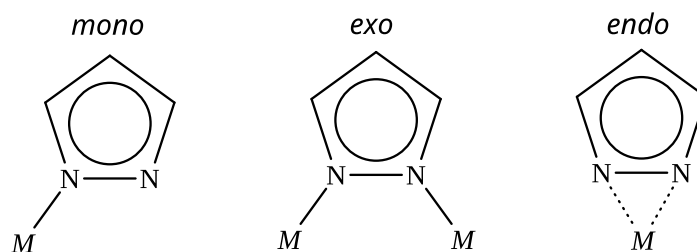
Figure 1.6: Pyrazole coordination modes

properties, whereas substituents at the 4-position can mainly change the electronic properties

Pyrazolates MOFs examples

MOF-303 and its derivates are an excellent example of pyrazole based MOFs with real applications as highly water-permeable[?] and water harvesting[?] capable materials.

They crystallizes in the monoclinic space group and its rodlike SBU consists of cistrans-alternating corned-shared AlO6 octahedra. The MOF synthesis is scalable in water.
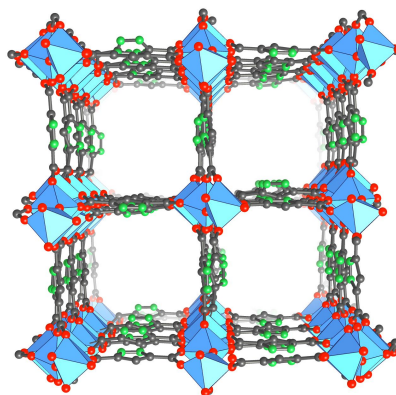
Figure 1.7: MOF-303 structure

2 Result and discussion

2.1 Pyrazole Ligand Development

2.1.1 Starting Materials

While several ligands were synthesized and utilized within the research group led by Professor L. Carlucci, my specific contribution focused exclusively on one of these ligands, that is based upon an already studied one. The structure of the starting material, 4,4'-malonyldibenzoic, and the retrosynthetic approch to it is reported.
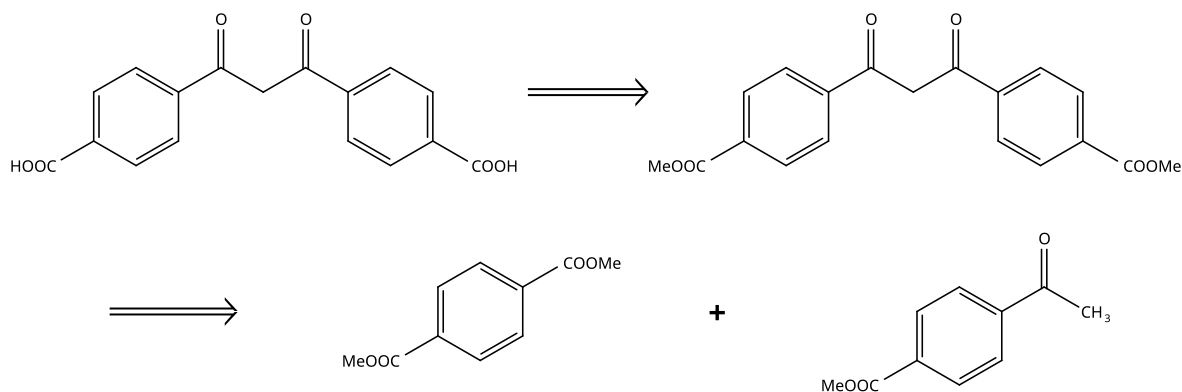
Figure 2.1: 4,4'-malonyldibenzoic acid Retrosynthesis Approach

The synthesis of the ligand is based on the cross-Claisen condensation followed by the hydrolisis of methylesters groups. The reaction was conducted with excellent yields and purity, resulting in the production of tens of grams of the compound. The reagents required for the synthesis are readily available commercially, facilitating the accessibility and reproducibility of the process.

The same synthetic approach has been carried out over a wide range of analogue compound, resulting tipically in high yield scalable reaction, as an example the ligand that will be analyzed in the section 2.2 has been approached with the same strategies.

The detailed information on the synthesis and charactherization of 4,4'-malonyldibenzoic are reported in the section 3.

2.1.2 Pyrazole Ligand Retrosynthesis

Starting from the 1,3-diketone diester intermediate, obtained from the cross Claisen condensation, the apparently easier way to procede with the pyrazole formation is using hydrazine, of which a brief reaction mechanism is reported.
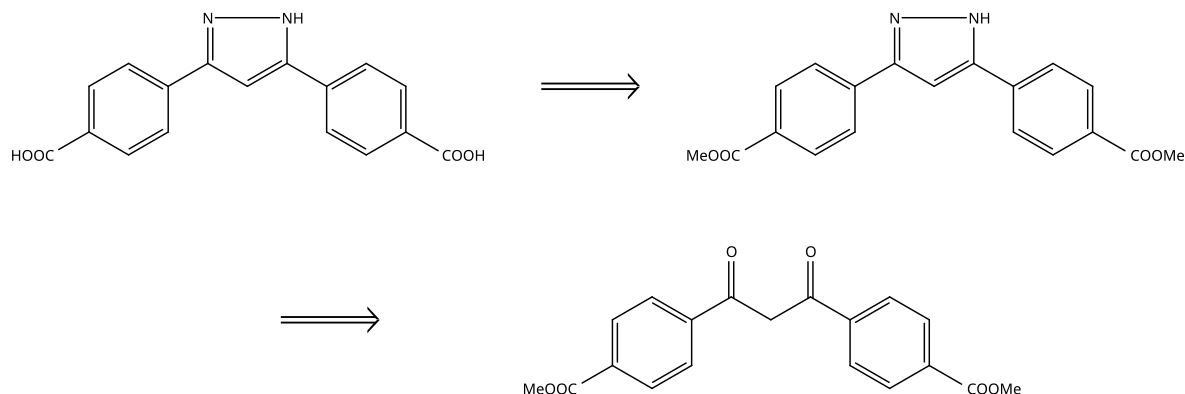
Figure 2.2: Pyrazole Ligand Retrosynthesis

Typically the reaction reaches completion in ethanol under reflux conditions with an excess of hydrazine within a few hours withouth any catalyzer: in this specific case several problems has been encountered and analyzed.
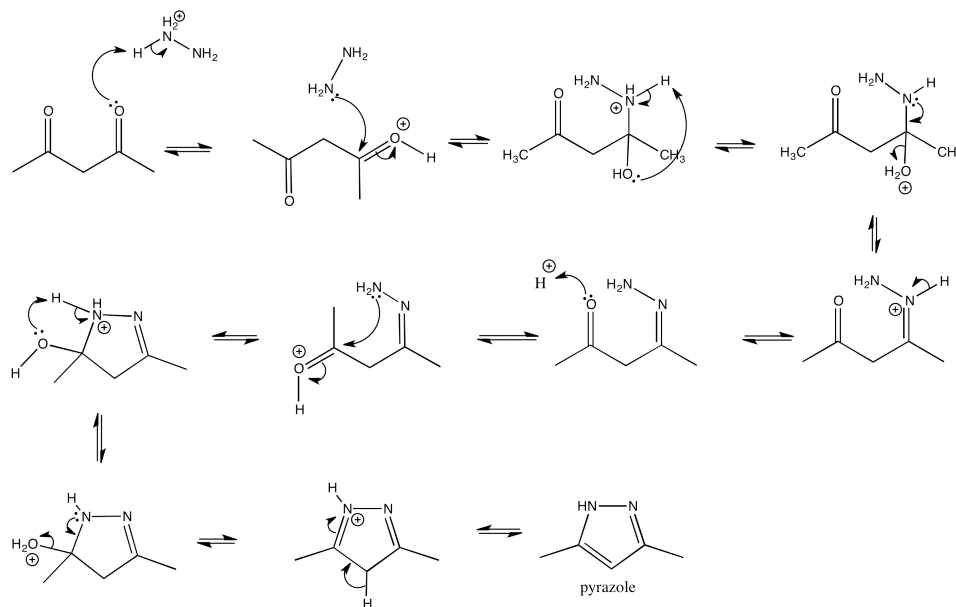
Figure 2.3: Pyrazole Formation Mechanism

After hydrazine addition the diester intermediate must undergo hydrolisis to obtain the corrisponding dibenzoic acid. The conditions for this step have also been analyzed.

2.1.3 Pyrazole Diester Formation

During the iterative process of analysing the reaction condition, certain key characteristics of the starting reagent and the product of interest were crucial. The diketone intermediate possesses a characteristic straw-yellow colour, and also exhibits photoluminescence in the yellow-green range easily observed with a UV lamp. The pyrazole product is whitish in colour and returns intense blue emission.

Numerous variables influencing the course of the reaction were evaluated, the results of which are given below. The use of IR spectroscopy and NMR structural analysis was instrumental in recognising the products formed and also the mixtures left over from failed reactions, especially in the first attempts.
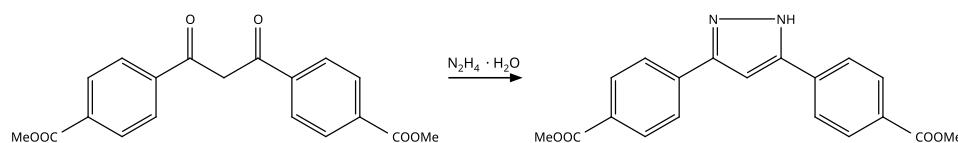
Figure 2.4: Pyrazole Formation

Solvent Choice and Reagent Solubility

Dimethyl 4,4'-malonyldibenzoate is poorly soluble in most organic solvents, even after careful grinding and sonication. Different alternatives has been evaluated, noting that the solvent choice was highly restricted from the fact that 65% hydrazine in water has been initially used.

Given the relatively low polarity of the ester functionalities, it would be beneficial to use a less polar solvent that still allows for the formation of a single phase with hydrazine water solution, while avoiding reactions with it.

The so low solubility can be attributed to several characteristic of the compound that is an highly simmetric diester. The correlation between solubility and symmetry of a molecule can vary depending on various factors, including the specific nature of the molecule and the solvent in which it is dissolved. While symmetry can influence certain aspects of a molecule's behavior, such as its physical properties, there is no general rule or direct correlation between solubility and symmetry. For example, symmetric molecules may have a more compact shape, which could reduce the surface area available for interaction with the solvent molecules. This can potentially lead to lower solubility if the solvent relies on surface interactions for dissolution. On the other hand, highly symmetrical molecules may have fewer polar or charged groups, which can reduce their ability to form favorable interactions with polar solvents. Other factors, such as temperature, pressure, and the presence of specific functional groups, can also significantly impact solubility. Experimental data and detailed analysis of the specific compound of interest are often required to accurately determine its solubility characteristics.

For these reasons the use of different solvent has been investigated, involving different approch for product recovery subsequently to solvent properties.

Reaction conducted in EtOH gives below 30% yield and overnight reaction time is needed to observe appreciable product formation. Nevertheless the solvent was easily removable by evaporation and the product can be easily purified afterward.

The use of THF as solvent gives no noticeable advantages in solubility or product recovery, providing lower yield than former conditions.

DMSO on the other hand provided higher yield of approximately 50%, at the cost of highly time

expensive and difficult recovering, as the product itself it highly soluble in the solvent. High diluition in water of the reaction mixture was carried out, followed by several centrifugation phases. This process was considered not reproducible and difficult to carry out especially on higher scale reaction. The valuable aspect of DMSO use are high reactant solubility and the possibility to achieve higher reaction temperature.

Trying to avoid extreme water diluition and centrifugation as recovery option DMF was tested as solvent, noting the fact that DMF can be evaporated, although the process is not quite as easy as with ethanol. The solubility of the reactant is not as great as in DMSO but better than EtOH or THF and higher temperature is possible too. Despite the promising outlook the result are not as expected and the observed yield is not higher than 30%. In addition, a non-negligible amount of DMF remains inside the flask following evaporation, making the purification process rather unreliable.

In light on the obtained information, it is worth to make a few assessments and adjustments.

The solubility of the reactant is clearly a key point in the formation of the product of interest, which is why DMSO or DMF should be used, noting also that the reaction seems to benefit from higher temperatures. However, using high-boiling solvents complicates the process of isolation and purification of the product in a way that is difficult to manage, making the synthesis process poorly reproducible.

Some example of similar problem has been found in literature [**?**], where using fluorinated derivatives of EtOH and iPrOH allowed to greatly improve the yield and in the specific case the regioselective of the hydrazine addition reaction on 1,3 non-symmetrical diketones. Owing to their unique properties (high hydrogen bonding donor ability, low nucleophilicity, high ionizing power and ability to solvate water), fluorinated alcohols, hexafluoroisopropanol (HFIP) and trifluoroethanol (TFE), has been shown to modify the course of reactions when they are used as solvents, allowing reactions, which usually require the use of added reagents or metal catalysts to be carried out under neutral and mild conditions.

Reagents proportions

Various values of the molar ratio between 1,3-diketone and hydrazine were tested after analysing the information available in the literature. The reaction was carried out in the presence of 2.5, 5 and 10 equivalents of hydrazine in relation to dimethyl 4,4'-malonyldibenzoate, using EtOH as solvent.

Molar Ration / n $(N_2H_4 \cdot H_2O)$ n$(4,4'-malonyldibenzoate)^{-1}$	Yield
2.5	-
5	25 %
10	6 %

Table I: Hydrazine Molar Ration

Reaction Time

As the starting intermediate is poorly soluble in most of the solvents used, the reaction inevitably takes place in heterophase, which could lead to longer reaction times than expected. For this reason, different reaction times were evaluated by following the course of the reaction in TLC and assessing the composition of the product obtained.

With a reaction time of about 3 h, no product formation was observed: it is necessary to let the mixture react overnight, however, no significant improvement in yield is observed for reaction times

longer than 24 h. Given that the reaction proceeds by a series of equilibrium steps, the experimental result is consistent with the reaction mechanism.

Reagent Choice

By further analyzing the reaction mechanism of the addition of hydrazine to 1,3-diketone, it became clearer that the water loss steps could be the reason for the problems associated with the low yield of the reaction. The use of hydrazine in aqueous solution would theoretically be detrimental in order to shift the equilibrium toward product formation, so the use of pure monohydrate hydrazine was verified. Reverifying the reaction trends using pure hydrazine, no significant changes in yield values were observed.
Supporting the experimentally obtained conclusion is the fact that several other addition reactions of hydrazine to 1,3 diketones have been conducted with good yields using aqueous solution of hydrazine monohydrate

Catalyst and Other Reagents

To increase the yield of the reaction, the use of catalysts and other chemicals has been considered. Small quantities of TEA and Et_4NOH has been tested in order to enhance the solubility of reagent in the used solvent, however the bases effect are detrimental on the reaction mechanism. No product formation has been observed using Et_4NOH and no noticeable difference in solubility has been observed. Use of sulfuric acid -¿ mechanism,dehydratation

Purification Methodes

The ideal purification method should be selective, efficient, scalable with minor side effects. Keeping in mind this preamble some alternative regarding the purification step has benn evaluated.

1. Gravity Chromatographic Column, Hex:AcOEt 75:25

   Through this process small quantities high purity product has been obtained. The operations itself is not time efficient and takes up large amount of solvent, leaving behind a considerable quantity of product of interest.
   Backing up this consideration, several examples of chromatographic column purification of similar pyrazole compounds are reported in the literature, but none of them use considerable amount of products.

2. Recrystallization from EtOH

   Taking special care to hot filter on Teflon filter any residual unreacted reagents reactants and conducting controlled cooling, a product of good purity is obtained in a reasonable time and using a minimum amount of solvent.
   Recrystallization from EtOH results frequently in the literature [?] as a purification method for pyrazole compounds.

3. Soxhlet with EtOH

   The technique appears to be of good efficiency in terms of purity but the amount of solvent and time required are excessive. The long time in contact with EtOH partially causes the dissolution of some of the impurities. It might be appropriate on large amounts of product where crystallization becomes less efficient.

Recrystallization in EtOH was found to be the easiest to perform and was the one that was used after this evaluation for all tests when deemed necessary.

2.1.4 Alternative Pyrazole Retrosynthesis

Considering the problem encountered in exploring the synthesis of this ligand, another synthetic route could be followed, using a croton condensation and then a conjugate addition to the newly formed $\alpha - \beta$ unsaturated compound [**?**].
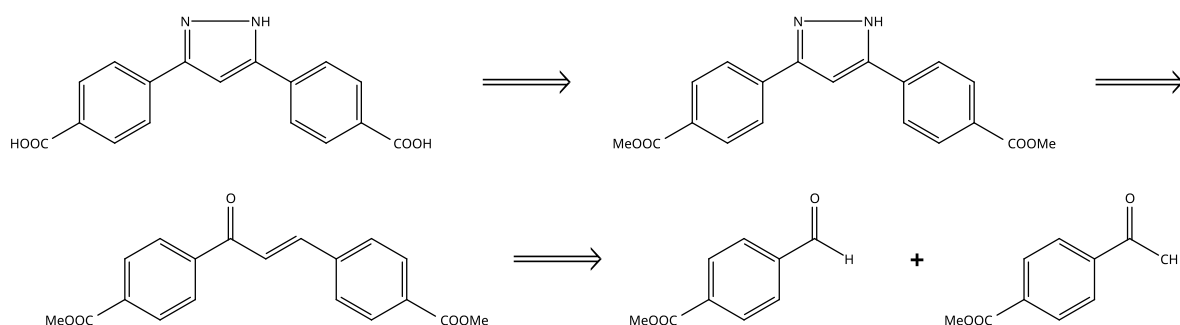
Figure 2.5: Pyrazole Ligand Alternative Retrosynthesis

The $\alpha - \beta$ unsaturated diester intermediate, is asymmetric and carry an olefinic group that should ensure good solubility in less polar organic solvents, allowing to bypass the key problem of solubility encountered with dimethyl 4,4'-malonyldibenzoate.

2.1.5 Unsaturaded Diester Formation

In order to evaluate the alternative synthetic pathway, with the hope of resolving the series of criticalities encountered, the conditions of formation of the first product reported in the retrosynthesis were evaluated.
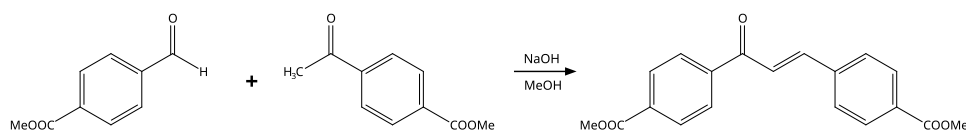
Figure 2.6: Unsaturated Compound Formation

The reagents required for the reaction are readily available, and the reaction conditions used follow those classically applied in croton condensation reactions. Some of the variables that can influence the reaction have been analyzed, based on the reagents properties.

Solvent

As for the solvent MeOH, it seems the easiest choice to avoid transesterification. Moreover, at the end of the reactions performed, the solvent takes on a yellowish colour that could be associated with the fraction of self-condensation product formed that would then be soluble in MeOH. The condensation product, on the other hand, precipitates quantitatively. This is very useful to avoid

any purification steps; any alternative solvent should retain this characteristic.

On the basis of the observations, all reactions were carried out in MeOH and the effect of the solvent has not been adequately investigated.

Reaction Temperature and Reagent Addition

The ketone could give rise to self-condensation leading to the formation of an unwanted product, reducing the yield and necessitating a purification step. However, the self-condensation reaction is kinetically disadvantageous compared to cross-condensation. For these reasons, a slow addition of the NaOH solution and an ice bath was used.

It would also be possible to exploit a slow addition of ketone to a dispersion of NaOH and aldehyde, however this is impractical as the ketone is not soluble in the typical amounts of solvent used and there is a risk of losing some of the reagent in the addition step.

Under conditions such as room temperature, 4h reaction time and 1:1 molar ratio of ketone to aldehyde, a yield of 81% was achieved. This, although satisfactory, can be further optimised.

Reaction Time

With regard to the optimal reaction time, some tests were carried out. An increase in reaction time leads to a significant increase in yield without any downside. The results obtained are shown below:

Time / h	Yield
4	81 %
15	89 %

Table II: Reaction time comparison

A longer reaction time leads to a significantly higher yield. Since the reaction takes place at room temperature and there are no special requirements in terms of equipment, there is no reason not to exploit this factor to achieve a higher yield.

Reagent Proportion

In order to optimise the reaction yield, it was thought that increasing the ratio of aldehyde to ketone might have a positive effect, as it theoretically reduces the possibility of a self-condensation reaction, given the higher aldehyde ratio. This possibility was tested with a 1.2 molar ratio of aldehyde to ketone, on an overnight reaction time. The results are shown below.

Aldehyde Molar Ration	Yield
1	89 %
1.2	94 %

Table III: Molar ratio comparison

It is evident that the effect of the slightly higher ratio is positive, but the effects on the reaction yield of even higher molar ratios have not been investigated: it might be interesting to evaluate this in the future.

2.1.6 Hydrazine addition to unsaturated precursor
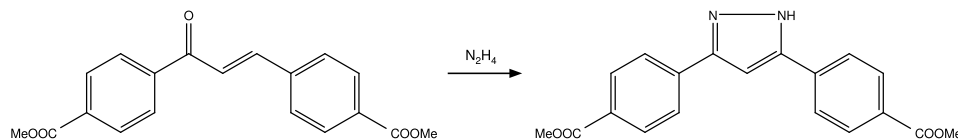
Figure 2.7: Hydrazine reaction with unsaturated compound

This step is the crucial one to evaluate the effectiveness of the alternative synthesis, given the difficulties encountered in forming the desired product from diketone.

The reaction mechanism does not differ substantially from that reported for diketone, however several of the difficulties encountered in the previous synthetic pattern, particularly the solubility of the initial compound, should not be encountered in this situation. The mechanism is reported below.

The conditions and results obtained for the formation of the pyrazole ring are evaluated, which include different solvents, catalysts, reaction times and the use of the microwave reactor. The conditions used were chosen from those reported in the literature [?], verifying the effectiveness of each procedure and modifying what was reported as needed.

Acetic Acid

According to the article in the literature, this synthetic route should be the simplest and most effective one, capable of achieving high reaction yields.
The reaction is conducted in acetic acid in the presence of hydrazine, with temperatures and reaction times that may vary depending on the substrate. At the end of the reaction, the product should precipitate spontaneously and the same should be easily recoverable by filtration.

However, on a practical level it was not optimal and the results obtained are discouraging. Neither product nor intermediate formation was observed, which reportedly should have been possible to isolate. No effects were observed even by extending reaction times and raising temperatures. The obtained results are reported below and the detailed experimental information are in the chapter 3.

ID	Temperature [K]	Time [h]	Yield
A	RT	1	-
B	363	3	-
C	393	48	-

Table IV: Acetic acid reaction yields

The results obtained did not meet expectations; although the reagent dissolved easily at higher temperatures than in test B, no product was recovered. In addition, investigation of the composition of the mixture were made difficult by the large amount of acetic acid present. Cooling the mixture succeeded in observing extensive precipitation of what was later identified with difficulty as the initial reagent.

Methanol with acid and basic catalysis

In addition to the conditions used with acetic acid, which proved to be a failure with this initial reagent, conditions including the use of methanol as a solvent in the presence of acidic and basic catalysts, such as hydrochloric acid and pyridine, are reported in the literature.

In relation to the use of hydrochloric acid as a catalyst the results obtained are poor. The addition of hydrazine to the solution of hydrochloric acid in methanol causes a rather violent reaction, subsequently the initial reactant does not solubilize. In the reaction time no change is observed. No product formation is observed, and the mixture does not develop blue photoluminescence. These reaction conditions were then not further investigated.

Unlike what is found in conditions of acid catalysis, the use of pyridine as a catalyst initially seems to lead to better results. The initial reactant completely solubilized and the mixture was left to react under reflux.

Ethanol

As a result of the difficulties encountered previously, several tests were carried out in ethanol alone, at varying temperatures and reaction times, maintaining the molar ratio costant.

The progress of the reaction was checked at room temperature, in the presence of 3 moles of hydrazine per moles of initial reagent, overnight, under strong stirring. The formation of crystalline product, with blue photolumiscence, was observed. What apparently looked like it might be the desired product, upon coupled NMR analysis such as COSY and NOESY turned out to be the hydrazonic intermediate whose structure is given below.
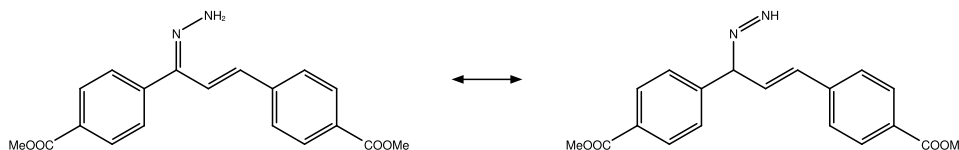
Figure 2.8: Intermediate compound structure

Indeed, within the literature it was reported that the intermediate could be obtained without proceeding to pyrazole ring closure. By raising the temperature, which should have a positive effect in this case, no change was observed in the product formed despite several hours of reaction. However, the crystallinity of the precipitate formed appears to the eye to be of a higher degree. The structure of the compound identified as an intermediate was again identified using NMR.

Microwaves

Microwave reactors have emerged as powerful tools in the field of chemistry [**?**], offering numerous advantages over traditional heating methods.

They are able to heat reaction mixtures very quickly, promoting mixing. They also provide granular control of temperature curves and easily allow working at pressures above atmospheric, reaching temperatures above the boiling temperature of the solvent itself at SATP. The amount of solvent used is small, it is not challenging to run reactions with small amounts, and the whole process is energy efficient.
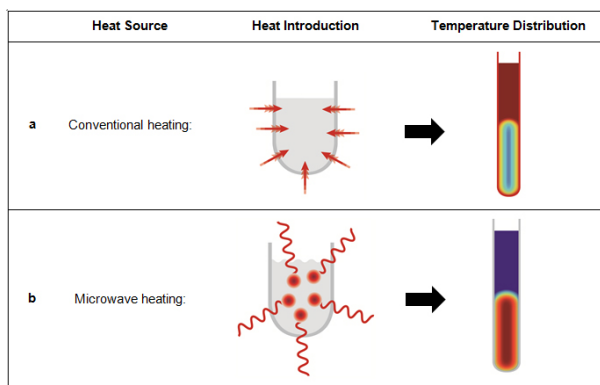
Figure 2.9: A good representation of the fundamental difference between traditional heating methods and microwaves, where the heat actually comes from the "inside".

Especially with regard to cyclization reactions and the formation of pyrazolines, several examples are available in the literature [**?**]. The high temperature and pressure make the reactions more selective by providing higher yields; in addition, reaction times are greatly reduced from hours to tens of minutes.

2.1.7   Pyrazole Ester Hydrolisis
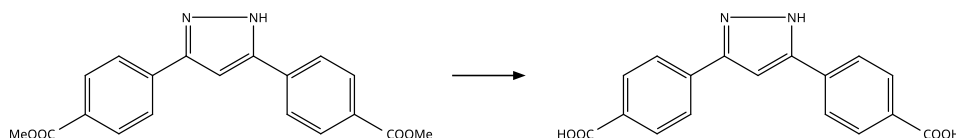
Figure 2.10: Pyrazole Diester Hydrolisis

The removal of methyl esters is a standard practice in organic chemistry, and no specific problems were encountered during this step of the synthesis.Two synthetic routes were analyzed. Initially, removal with LiOH in THF:H_2O 1:1 was used, but the yields of the different reactions were not highly reproducible, perhaps because of the precariousness of the different impurities resulting from the different methods of pyrazole synthesis.
Instead, removal with NaOH in MeOH:H_2O 1:1 could give better results, with higher yield and cleaner processing.
It is important to note, however, that the results obtained refer to small amounts of reagent, often containing impurities, this due to the large diffculty in obtaining the ester. Thus, this aspect of the synthesis will have to be finalized when the ester synthesis is optimized.

mancano rese e risultati ottenuti

2.1.8   Conclusion, consideration and further development

2.2 Electrochemistry

2.2.1 Why electrochemistry and cyclic voltammetry?

Cyclic voltammetry (CV) is a technique used in the analysis of organic and inorganic compounds [**?**]. It provides information about reduction and oxidation processes of molecular species, the stability, the conversion and storage of energy and several other interesting properties. CV is also invaluable to study electron transfer-initiated chemical reactions, which includes catalysis. CV can be used to estimate the electron affinity and ionization energy of test compounds in a much cheaper and faster way compared to other high vacuum methods. It is important in electrochemistry because it allows to study the electron transfer processes that are at the center of the reactivity of inorganic complexes. The aforementioned parameters correlate with the energy levels of highest occupied molecular orbital (HOMO) and lowest unoccupied molecular orbital (LUMO).

When cyclic voltammetry (CV) is combined with or ultraviolet-visible and near-infrared (UV-Vis-NIR) spectroscopies, it provides useful information such as electron affinity, ionization potential, band-gap energies, the type of charge carriers, and degradation information [**?**]. Analizing the CV experiment data is possible to determine the band-gap value that can be later evaluted against the one obtained from the spectroscopic UV data.

Ligands and Background

Electrochemistry exploration has been carried out on already studied compounds [**?**], that have been already synthetized and studied. Structure and synthesis of ligand and monomers are given below, details of reaction conditions are given in section 3.
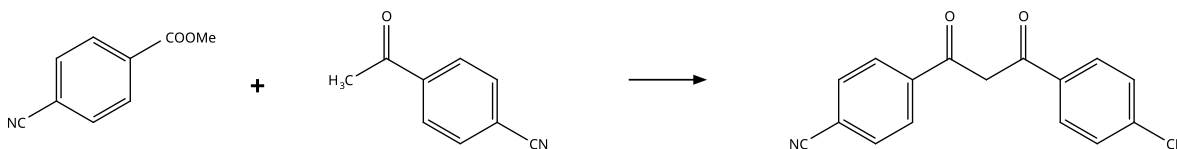
Figure 2.11: 4-[3-(4-cyanophenyl)-3-oxopropanoyl]benzonitrile

mancano risultati ottenuti

3 Experimental section

3.1 Dimethyl 4,4'-malonyldibenzoate

3.1.1 Synthesis

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
methyl 4-acetylbenzoate	3609-53-8	178.19	4.400	24.69	1.20
dimethyl terephthalate	120-61-6	194.19	4.000	20.60	1.00
NaH, mineral dispersion	7646-69-7	23.998	1.132	47.17	2.30

The reaction is carried out under N_2 atmosphere, the reactant are carefully dried beforehand.
In a flask, 1.132 g of NaH dispersion in mineral oil are washed with anhydrous THF (15 mL x 2).
4.0 g of methyl 4-acetyl benzoate and 4.4 g of dimethyl terephthalate are dissolved in 45 mL of anhydrous THF, and then NaH suspension in THF is added.
The mixture is heated to reflux overnight.
Evaporation of the solvent under vacuum gives a dark brown compound. The solid is taken up in water and acidified with HCl 20% in an ice bath. Taking care to leave it to react for a few hours and checking the pH to make sure it is acid. More water is added if mixing is particularly difficult. The raw product is recovered as a yellow solid by filtration with a Buchner funnel. To eliminate the impurities of the remaining reagents the solid is washed with chloroform and with diethyl ether. The solid is lastly collected on a Buchner funnel and dried. Yield: 92%, 8.67 g of product.

3.1.2   Characterization

NMR
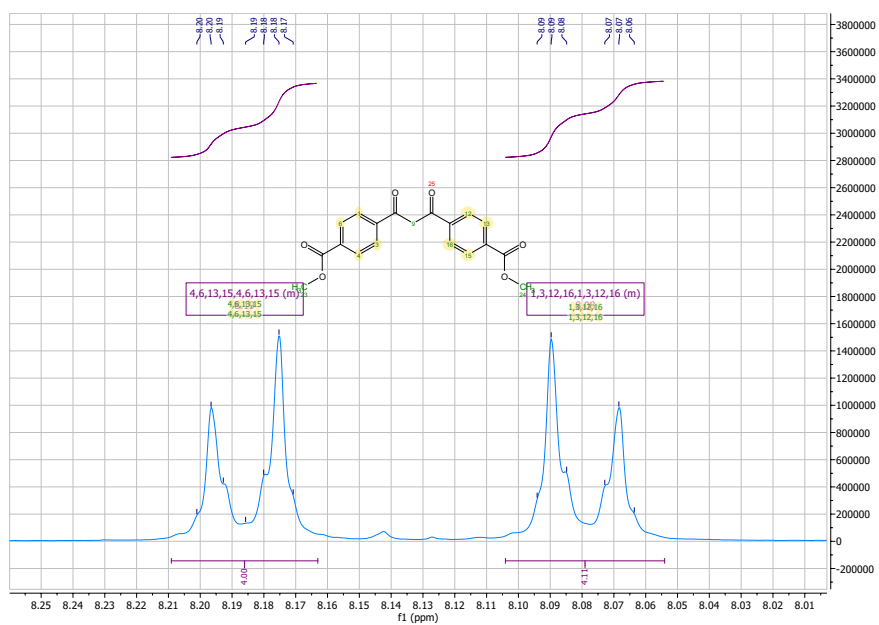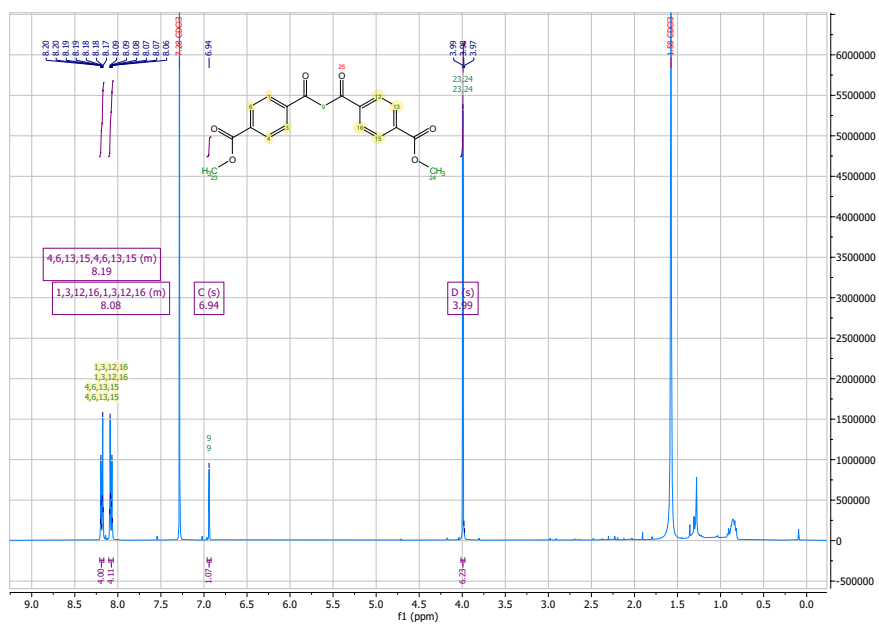
IR

3.2   4,4'-malonyldibenzoic acid

3.2.1   Synthesis

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
Dimethyl 4,4'-malonyldibenzoate	-	340.31	1.000	2.940	1
LiOH	1310-65-2	23.95	1.411	58.80	20

In a small beaker, 1.411 g of freshly ground LiOH is dissolved in 10 mL of water. In an flask, 1.000 g of 4,4'-malonyldibenzoate is suspended in 10 mL of THF, then the LiOH solution is slowly added. The mixture is allowed to react 7 hours at room temperature under vigorous stirring. It is then proceeded by extracting with CHCl$_3$ (3x25 mL). The aqueous phase is collected and acidified with 10 % HCl to acidic pH, observing the formation of straw-yellow precipitate. The solid is recovered by filtration over buchner and allowed to air dry.

3.2.2 Characterization

NMR

IR

3.3 Methyl 4,4'-(1H-pyrazole-3,5-diyl)dibenzoate

3.3.1 Synthesis from methyl 4,4'malonyldibenzoate

EtOH

A typical synthesis is described here: 500 mg of dimethyl 4,4'-malonyldibenzoate is suspended in 20 mL of EtOH in a 50 mL flask. The mixture is left in ultrasonic bath for 30'. 350 μL of hydrazine monohydrate are slowly added to the mixture that is then heated to reflux overnight.
The compound is collected as a pale yellow solid on a teflon funnel. Yield: 25%.

DMF

500 mg of dimethyl 4,4'-malonyldibenzoate is suspended in 15 mL of DMF in a 50 mL flask. The mixture is left in ultrasonic bath for 30'. 350 μL of hydrazine monohydrate are slowly added to the mixture that is then heated to 150°C overnight.
Evaporation of the solvent under vacuum gives a white ivory solid. The compound is then washed adding 25 mL of ethanol and heat to reflux for 2 hours. Pure solid is collected on a Buchner funnel. Yield: 48%.

DMSO

500 mg of dimethyl 4,4'-malonyldibenzoate is solubilized in 20 mL of DMSO in a 50 mL flask. The mixture is left in ultrasonic bath for 30'. 350 μL of hydrazine monohydrate are slowly added to the mixture that is then heated to 150°C overnight.

Characterization

NMR

IR

3.3.2 Synthesis from Methyl 4-[(1E)-3-[4- (methoxycarbonyl) phenyl]-3-oxoprop-1-en- 1-yl]benzoate

Acetic acid

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
Unsaturated compound	-	324.332	0.370	1.14	1
$N_2H_4 \cdot H_2O$	7803-57-8	50.05	0.166	3.42	3

EtOH with HCl

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
Unsaturated compound	-	324.332	0.300	0.92	1
$N_2H_4 \cdot H_2O$	7803-57-8	50.05	0.138	2.76	3

EtOH with pyridine

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
Unsaturated compound	-	324.332	0.300	0.92	1
$N_2H_4 \cdot H_2O$	7803-57-8	50.05	0.138	2.76	3

EtOH

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
Unsaturated compound	-	324.332	0.500	1.54	1
$N_2H_4 \cdot H_2O$	7803-57-8	50.05	0.231	4.62	3

3.3.3 Characterization

NMR

IR

3.4 4,4'-(1H-pyrazole-3,5-diyl)dibenzoic acid

3.4.1 Synthesis

LiOH

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
4,4'-(1H-pyrazole-3,5-diyl)dibenzoic acid	-	336.47	0.3023	0.8988	1
LiOH	1310-65-2	23.95	0.4505	17.796	20

In a small beaker, 0.4505 g of freshly ground LiOH is dissolved in 6 mL of water. In an flask, 0.3023 g of methyl 4,4'-(1H-pyrazole-3,5-diyl)dibenzoate is suspended in 6 mL of THF, then the LiOH solution is slowly added. The mixture is allowed to react overnight at room temperature under vigorous stirring. It is then proceeded by extracting with CHCl$_3$ (3x25 mL). The aqueous phase is collected and acidified with 10 % HCl to acidic pH, observing the formation of white precipitate. The solid is recovered by filtration over buchner and allowed to air dry.

3.4.2 Characterization

NMR

IR

3.5   4,4'-malonyldibenzonitrile

3.5.1   Synthesis

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
methyl 4-cyanobenzoate	1129-35-7	161.16	4.0370	25.05	0.93
4-acetylbenzonitrile	1443-80-7	145.16	3.3862	23.32	1.00
NaH	7646-69-7	23.998	2.7780	115.75	4.96

The reaction is carried out under N_2 atmosphere, the reactant are carefully dried beforehand.

In a flask, 2.778 g of NaH dispersion in mineral oil are washed with anhydrous THF (15 mL x 2). 4.0370 g of methyl 4-cyanobenzoate and 3.3862 g of 4-acetylbenzonitrile are dissolved in 45 mL of anhydrous THF, and then NaH suspension in THF is added under ice bath while vigouros mixing. The mixture is allowed to reach room temperature and then is heated to reflux overnight.

Evaporation of the solvent under vacuum gives a dark green compound. The solid is taken up in water and acidified with HCl 20% in an ice bath. Taking care to leave it to react for a few hours and checking the pH to make sure it is acid. More water is added if mixing is particularly difficult. The raw product is recovered as a yellow solid by filtration with a Buchner funnel. To eliminate the impurities of the remaining reagents the solid is washed with ethanol. The solid is lastly collected on a Buchner funnel and dried. Yield: 74%, 4.76 g of product.

3.5.2 Characterization

NMR
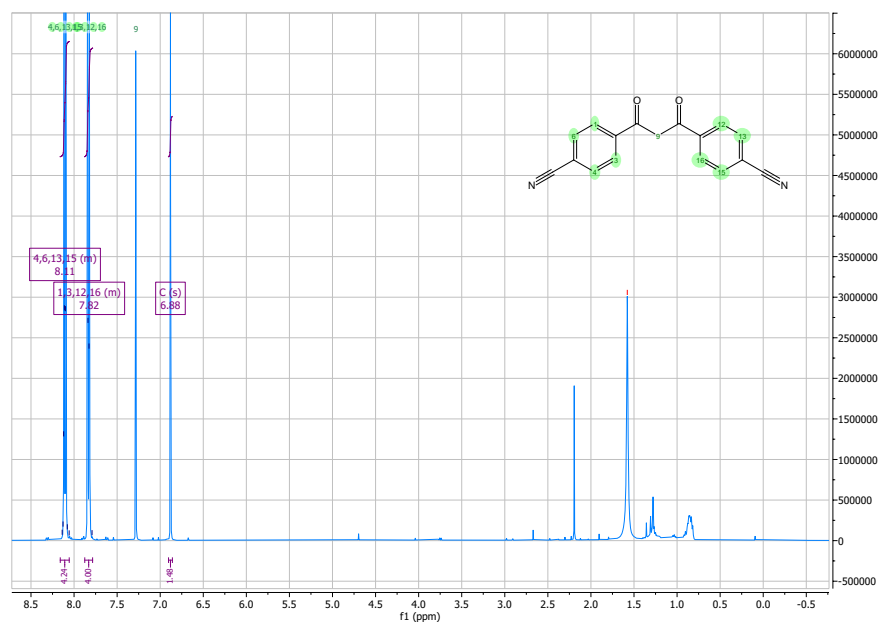
Figure 3.1: DikDiCN ^{1}H-NMR (400 MHz)
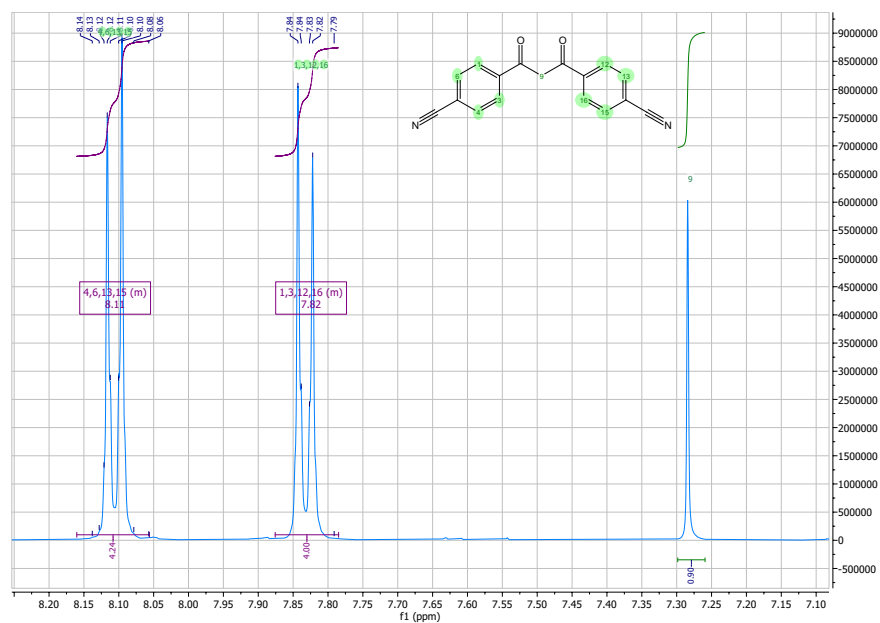
Figure 3.2: DikDiCN ^{1}H-NMR (400 MHz), zoom on diagonistic peaks

3.6 Methyl 4-[(1E)-3-[4-(methoxycarbonyl)phenyl]-3-oxoprop-1-en-1-yl]benzoate

3.6.1 Synthesis

Reagent	CAS	MW / g mol^{-1}	m / g	n / mmol	SR
methyl 4-formylbenzoate	1571-08-0	178.18	0.500	2.8	1
methyl 4-acetylbenzoate	3609-53-8	164.16	0.550	3.5	1.2
NaOH	1310-73-2	39.998	0.230	5.6	2

In a small beaker, 230 mg of freshly ground NaOH is dissolved in 10 mL of methanol.

In a flask, 500 mg methyl 4-formylbenzoate and 550 mg methyl 4-acetylbenzoate are dissolved in 5 mL MeOH. The mixture is placed under vigorous stirring in an ice bath. The freshly prepared sodium hydroxide solution is slowly added. The reaction is left to react overnight.

The yellowish solid is collected on Buchner funnel and allowed to air dry. Yield 94%, 0.85 g of product.
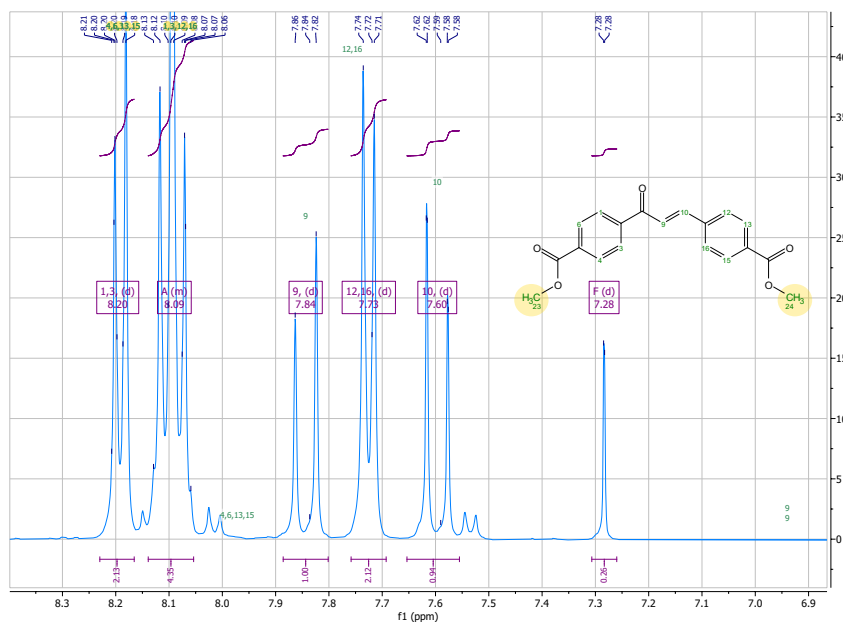
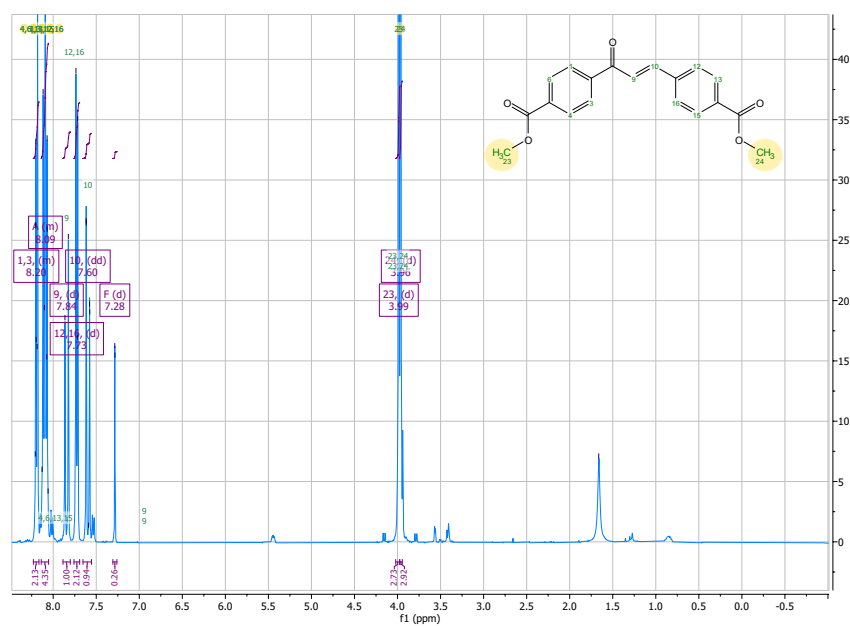### 3.6.2 Characterization

NMR

Figure 3.3: HNMR

Figure 3.4: HNMR zoom

4 Tool Development

4.1 Ultra specific command-line based software, why?

In recent times, there has been a decline in the affinity towards command-line interfaces, which is a regrettable development. While graphical user interfaces (GUIs) have gained popularity due to their intuitive nature and visual appeal, the diminishing use of command-line tools is unfortunate. Despite their perceived complexity, command-line interfaces offer unparalleled efficiency, flexibility, and control, making them indispensable for certain tasks and user groups. It is important to recognize the value of command-line tools and the unique advantages they bring to the table, fostering a balanced approach that embraces both GUIs and command-line interfaces.

Today, there is still a need for ultra-specific command-line-based tools despite the availability of graphical user interfaces (GUIs) for several reasons:

1. Efficiency: Command-line tools often offer a more efficient and streamlined way of performing tasks compared to GUIs. With command-line interfaces, users can execute complex operations using concise commands and leverage scripting capabilities to automate repetitive tasks. This efficiency is particularly valuable in scenarios involving large-scale data processing, system administration, or software development.

2. Flexibility and Customizability: Command-line tools provide users with a high degree of flexibility and customizability. They offer a wide range of options, flags, and parameters that can be combined to achieve specific outcomes. Users can tailor commands to their specific needs, combining different tools and building complex workflows. This level of control and adaptability is often preferred by power users and professionals who require fine-grained control over their tools.

3. Remote Access and Automation: Command-line tools are ideal for remote access and automation scenarios. Through remote shell access, users can manage and control systems from anywhere, which is particularly useful for server administration, cloud computing, or managing distributed environments. Additionally, command-line tools can be easily integrated into scripts and workflows, allowing for automated execution of tasks and easy integration with other tools or systems.

4. Resource Efficiency: GUIs typically consume more system resources compared to command-line tools. This is because GUIs require graphical rendering, user interface components, and other overhead, whereas command-line tools operate in a text-based environment, requiring fewer system resources. In resource-constrained environments or when dealing with large-scale operations, command-line tools can be more efficient and less demanding on the system.

5. Reproducibility and Version Control: Command-line tools facilitate reproducibility and version

control in software development and data analysis workflows. By documenting the specific commands and parameters used, it becomes easier to reproduce results or share workflows with others. Additionally, version control systems like Git are well-suited for tracking changes to text-based code and configuration files, making it easier to collaborate and manage projects effectively.

4.1.1 Open Source Software

Open source software plays a significant role in the fields of university education and chemistry. In academia, the use of open source software promotes collaboration, knowledge sharing, and innovation. Universities often encourage the adoption of open source tools as they provide students and researchers with accessible and cost-effective alternatives to proprietary software.

In the domain of chemistry, open source software offers a wide range of benefits. It empowers researchers to analyze, model, and simulate chemical systems, enhancing their understanding and enabling them to make informed decisions. Open source software encourages transparency and reproducibility in scientific experiments, as the source code is freely available for scrutiny and modification.

Furthermore, open source software encourages the development of a vibrant community around a particular tool. Chemists and researchers can contribute to the improvement of open source software by suggesting new features, reporting bugs, or even collaborating on code development. This collaborative environment fosters a sense of collective knowledge creation and sharing, benefiting the entire chemistry community.

Moreover, open source software aligns with the principles of academic freedom and intellectual property sharing. It enables researchers to customize software according to their specific needs, adapt it to evolving research requirements, and redistribute their modifications to the community. This ethos promotes the rapid advancement of scientific knowledge and encourages interdisciplinary collaborations.

Overall, open source software in the context of university and chemistry serves as a catalyst for innovation, education, and scientific progress. Its accessibility, flexibility, and collaborative nature make it an invaluable resource for students, researchers, and professionals, helping to accelerate breakthroughs and advancements in the field of chemistry.

Python and Matplotlib

Python and the matplotlib library have played pivotal roles in the fields of university education and chemistry due to their versatility, ease of use, and extensive visualization capabilities.

Python, being a powerful and accessible programming language, has gained immense popularity among students and researchers. Its intuitive syntax and extensive libraries make it an ideal choice for scientific computing and data analysis tasks. In the context of university education, Python serves as an effective tool for teaching programming fundamentals, data manipulation, and numerical computing.

In chemistry, Python offers a wide range of libraries and frameworks specifically designed for scientific analysis and modeling. These libraries, such as NumPy, SciPy, and Pandas, provide efficient numerical computations, statistical analysis, and data manipulation capabilities. Python's extensive ecosystem also includes specialized chemistry libraries like RDKit, PyMOL, and ASE, which enable

chemists to perform tasks such as molecular visualization, molecular dynamics simulations, and quantum chemistry calculations.

One of the key strengths of Python in chemistry lies in its seamless integration with the matplotlib library. Matplotlib is a popular plotting library that allows the creation of high-quality, publication-ready visualizations. It provides a wide range of plot types, customization options, and interactive features, making it a valuable tool for visualizing chemical data, molecular structures, spectra, reaction pathways, and more. Matplotlib's integration with Python's scientific libraries enables researchers and educators to create informative and visually appealing graphs, aiding in data analysis, hypothesis testing, and presentation of research findings.

Python and matplotlib also contribute to the reproducibility of scientific research. Researchers can document their analysis workflows, including data processing, modeling, and visualization steps, using Python scripts. These scripts can be shared, reused, and easily reproduced, facilitating collaboration and enhancing the transparency of scientific investigations.

4.2 Plotty

In light of the aforementioned reasons, I have made the decision to give a small contribution to this domain by developing a small command-line utility aimed at rapidly generating graphs from IR and UV spectroscopy data, offering predefined styles and commonly used processing options.

The purpose of this utility is to provide researchers and analysts with a streamlined and efficient tool for graph generation. By leveraging the command-line interface, users can swiftly generate plots by executing concise commands, thereby enhancing productivity and reducing manual efforts.
The utility will offer a range of predefined styles, enabling users to choose from various visual representations that align with common conventions in spectroscopy. These styles will encompass color schemes, line types, markers, and other visual attributes that facilitate the clear and accurate representation of spectral data.

This tool will strive to uphold the principles of open-source software, facilitating collaboration, reproducibility, and customization. It is my aspiration that this utility will assist researchers, educators, and practitioners in their endeavors, streamlining the visualization and analysis of spectroscopic data and fostering further discoveries in this domain.

4.2.1 Major Functionalities

The Python program is organized into several components, including a command-line interface, a `CSVPlotter` class, and specific plot styles. Here is an overview of its functionality:

Command-Line Interface

The program utilizes the `click` library to create a command-line interface, allowing users to execute it from the command line with various options and arguments.

CSVPlotter Class

The core of the program is the `CSVPlotter` class, responsible for handling the following tasks:

- Initializing with parameters such as CSV file paths, output format, plot style, and the option to create combined plots.

- Reading CSV files and extracting data (x and y values) for plotting.

- Generating plots based on the specified plot style, supporting styles like IR, HNMR, CNMR, and UV spectroscopy.

- Saving the generated plots as image files in the chosen output format.

Supported Plot Styles

The program supports multiple plot styles, each with its specific configurations:

- **IR (Infrared Spectroscopy) Style**: Plots labeled with "Wave Number / cm^{-1}" on the x-axis and "T / %" on the y-axis. It supports combined plots with multiple CSV files.

- **UV (Ultraviolet-Visible Spectroscopy) Style**: Plots labeled with "λ / nm" on the x-axis and "A / a.u." on the y-axis. It supports combined plots with multiple CSV files.

Output Files

The program saves the generated plots as image files in the specified output format (e.g., PNG, SVG, PDF, EPS). The output filenames are derived from the input CSV filenames.

Combined Plots

If the `--combined` flag is specified when executing the program, it can create combined plots that overlay data from multiple CSV files into a single plot.

Success Message

After generating the plots, the program prints a success message to indicate that the plots have been created successfully.

4.3 Complete Code

For completeness below is the program code in its entirety. There are several comments identified by the "#' symbol and highlighted in red that make it easier to read and were found to be essential in the writing.

```python
import click
import os
import csv
import random
from matplotlib import pyplot as plt
from rich.console import Console

console = Console()


class CSVPlotter:
    """Plot from csv file."""

    def __init__(self, csv_files, output_format, plot_style,
        combined):
        """Initialize all the variables."""
        self.csv_files = csv_files
        self.output_format = output_format
        self.plot_style = plot_style
        self.combined = combined
        self.colors = [
            "blue",
            "red",
            "green",
            "orange",
            "purple",
            "black",
        ]  # Add more colors if needed

    def process_csv_file(self, csv_file):
        """Create x and y list for each file."""
        x = []
        y = []

        with open(csv_file, "r") as csvfile:
            plots = csv.reader(csvfile, delimiter=";")
            for row in plots:
                try:
                    x.append(float(row[0]))
                    y.append(float(row[1]))
                except ValueError:
```

```
41                      continue
42
43          return x, y
44
45      def create_plot(self):
46          """Create plot from x and y list."""
47          if self.combined:
48              # Create a single figure and axes objects
49              fig, ax = plt.subplots()
50          else:
51              fig, ax = None, None
52
53          for csv_file in self.csv_files:
54              x, y = self.process_csv_file(csv_file)
55
56              color = "black" if not self.combined else random.choice(
57                  self.colors)
57              label = os.path.splitext(os.path.basename(csv_file))[0]
58
59              if self.plot_style == "IR":
60                  if self.combined:
61                      ax.plot(x, y, color=color, linewidth=0.4, label=
                          label)
62                      ax.set_xlabel("Wave␣Number␣/␣$cm^{-1}$")
63                      ax.set_ylabel("T␣/␣%")
64                      ax.set_xlim(4000, 500)  # Set the x-axis limits
65                      ax.set_ylim(0, 100)  # Set the y-axis limits
66                      ax2 = ax.twinx()
67                      # Set the same x-axis ticks as the primary axis
68                      ax2.set_xticks(
69                          ax.get_xticks()
70                      )
71                      ax2.set_xlabel("")
72                      ax2.set_ylabel("")
73
74                      # Remove tick labels from ax2
75                      ax2.set_yticklabels([])
76                      ax3 = ax.twiny()  # Create the tertiary x-axis
77                      ax3.set_xlabel("")
78                      ax3.set_ylabel("")
79                      # Set the same x-axis ticks as the primary axis
80                      ax3.set_xticks(
81                          ax.get_xticks()
82                      )
83
84                      # Remove tick labels from ax2
85                      ax3.set_xticklabels([])
86
```

```python
87                else:
88                    fig, ax = plt.subplots()
89                    ax.plot(x, y, color=color, linewidth=0.4, label=
                         label)
90                    ax.set_xlabel("Wave␣Number␣/␣$cm^{-1}$")
91                    ax.set_ylabel("T␣/␣%")
92                    ax.set_xlim(4000, 500)   # Set the x-axis limits
93                    ax.set_ylim(0, 100)   # Set the y-axis limits
94                    ax2 = ax.twinx()
95                    # Set the same x-axis ticks as the primary axis
96                    ax2.set_xticks(
97                        ax.get_xticks()
98                    )
99                    ax2.set_xlabel("")
100                   ax2.set_ylabel("")
101
102                   # Remove tick labels from ax2
103                   ax2.set_yticklabels([])
104
105                   ax3 = ax.twiny()   # Create the tertiary x-axis
106                   ax3.set_xlabel("")
107                   ax3.set_ylabel("")
108                   # Set the same x-axis ticks as the primary axis
109                   ax3.set_xticks(
110                       ax.get_xticks()
111                   )
112
113                   # Remove tick labels from ax2
114                   ax3.set_xticklabels([])
115
116                   # Add the legend to the axes object
117                   ax.legend()
118                   output_filename = (
119                       os.path.splitext(csv_file)[0] + "." + self.
                         output_format
120                   )
121                   plt.savefig(output_filename, format=self.
                        output_format)
122                   plt.clf()
123                   click.echo(f"Plot␣saved␣as␣{output_filename}")
124
125           elif self.plot_style == "HNMR":
126               if self.combined:
127                   ax.plot(x, y, color=color, linewidth=0.4, label=
                         label)
128                   ax.set_xlabel("Numero␣d'onda␣/␣$cm^{-1}$")
129                   ax.set_ylabel("T␣/␣%")
130                   ax.set_xlim(15, 0)   # Set the x-axis limits
```

```
131             ax2 = ax.twinx()
132             # Set the same x-axis ticks as the primary axis
133             ax2.set_xticks(
134                 ax.get_xticks()
135             )
136             ax2.set_xlabel("")
137             ax2.set_ylabel("")
138
139             # Remove tick labels from ax2a
140             ax2.set_yticklabels([])
141             ax3 = ax.twiny()  # Create the tertiary x-axis
142             ax3.set_xlabel("")
143             ax3.set_ylabel("")
144             # Set the same x-axis ticks as the primary axis
145             ax3.set_xticks(
146                 ax.get_xticks()
147             )
148
149             # Remove tick labels from ax2
150             ax3.set_xticklabels([])
151
152         else:
153             fig, ax = plt.subplots()
154             ax.plot(x, y, color=color, linewidth=0.4, label=
                    label)
155             ax.set_xlabel("Numero␣d'onda␣/␣$cm^{-1}$")
156             ax.set_ylabel("T␣/␣%")
157             ax.set_xlim(10, 0)  # Set the x-axis limits
158             ax.set_ylim(0, 1e6)  # Set the x-axis limits
159             ax2 = ax.twinx()
160
161             # Set the same x-axis ticks as the primary axis
162             ax2.set_xticks(
163                 ax.get_xticks()
164             )
165             ax2.set_xlabel("")
166             ax2.set_ylabel("")
167
168             # Remove tick labels from ax2
169             ax2.set_yticklabels([])
170
171             ax3 = ax.twiny()  # Create the tertiary x-axis
172             ax3.set_xlabel("")
173             ax3.set_ylabel("")
174
175             # Set the same x-axis ticks as the primary axis
176             ax3.set_xticks(
177                 ax.get_xticks()
```

```
178                    )
179
180                    # Remove tick labels from ax2
181                    ax3.set_xticklabels([])
182                    ax.legend()  # Add the legend to the axes object
183                    output_filename = (
184                        os.path.splitext(csv_file)[0] + "." + self.
                            output_format
185                    )
186                    plt.savefig(output_filename, format=self.
                        output_format)
187                    plt.clf()
188                    click.echo(f"Plot␣saved␣as␣{output_filename}")
189
190            elif self.plot_style == "CNMR":
191                if self.combined:
192                    ax.plot(x, y, color=color, linewidth=0.2, label=
                        label)
193                else:
194                    fig, ax = plt.subplots()
195                    ax.plot(x, y, color=color, linewidth=0.2, label=
                        label)
196                    ax.set_xlabel("$\lambda$␣/␣nm␣")
197                    ax.set_ylabel("A␣/␣a.u.")
198                    ax.set_xlim(15, 0)  # Set the x-axis limits
199                    ax.legend()  # Add the legend to the axes object
200                    output_filename = (
201                        os.path.splitext(csv_file)[0] + "." + self.
                            output_format
202                    )
203                    plt.savefig(output_filename, format=self.
                        output_format)
204                    plt.clf()
205                    click.echo(f"Plot␣saved␣as␣{output_filename}")
206
207            elif self.plot_style == "UV":
208                if self.combined:
209                    ax.plot(x, y, color=color, linewidth=0.2, label=
                        label)
210                else:
211                    fig, ax = plt.subplots()
212                    ax.plot(x, y, color=color, linewidth=0.2, label=
                        label)
213                    ax.set_xlabel("$\lambda$␣/␣nm␣")
214                    ax.set_ylabel("A␣/␣a.u.")
215                    ax.legend()  # Add the legend to the axes object
216                    output_filename = (
217                        os.path.splitext(csv_file)[0] + "." + self.
```

```
                                output_format
218                         )
219                         plt.savefig(output_filename, format=self.
                                output_format)
220                         plt.clf()
221                         click.echo(f"Plot saved as {output_filename}")
222
223             if self.combined:
224                 combined_filename = "_".join(
225                     [os.path.splitext(os.path.basename(f))[0] for f in
                            self.csv_files]
226                 )
227                 output_filename = f"{combined_filename}.{self.
                        output_format}"
228                 ax.legend()  # Add the legend to the axes object
229                 plt.savefig(output_filename, format=self.output_format)
230                 plt.clf()
231                 click.echo(f"Combined plot saved as {output_filename}")
232
233
234   @click.command()
235   @click.argument("files", nargs=-1, type=click.Path(exists=True))
236   @click.option(
237       "--output-format",
238       type=click.Choice(["png", "svg", "pdf", "eps"]),
239       default="png",
240       help="Output format for the plot",
241   )
242   @click.option(
243       "--style",
244       type=click.Choice(["IR", "UV", "HNMR", "CNMR"]),
245       default="default",
246       help="Plot style",
247   )
248   @click.option(
249       "--combined",
250       is_flag=True,
251       default=False,
252       help="Create a combined plot with multiple CSV files",
253   )
254   def export_plot(files, output_format, style, combined):
255       """Call an instance to export the plot."""
256       if not files:
257           click.echo("Please provide one or more CSV files.")
258           return
259
260       plotter = CSVPlotter(files, output_format, style, combined)
261       plotter.create_plot()
```

```
262
263        # Print a success message
264        console.print(":tada: [bold green] Plots created successfully![/
               bold green] :tada:")
265
266
267  if __name__ == "__main__":
268        export_plot()
```