Supplementary information

Contents

1. The theory of VTNA

For the hypothetical reaction in Eq. S1, catalysed by a catalyst "cat", the global rate equation can be written as Eq. S2 where a, b and c denote the reaction orders with respect to species A, B and cat, respectively, depending on the reaction mechanism. The product concentration at any time t_n is given by Eq. S3 by integration of Equation S2 over time.

$$A + B \rightarrow P \qquad (Eq.S1)$$

$$\frac{d[P]}{dt} = k[A]^a[B]^b[cat]^c \qquad (Eq.S2)$$

$$\int_0^{t_n} d[P](t) = \int_0^{t_n} k[A]^a[B]^b[cat]^c dt \qquad (Eq.S3)$$

Performing two experiments with identical conditions except altering one initial concentration, such as $[A]_0$ (Exp 1 and 2 Fig. S1A), leads to distinct product concentration profiles. This divergence in the product concentration profiles of experiments with varying $[A]_0$ is illustrated in Figure S1B and occurs because the integrand in equation S3 is not identical for the experiments over time, unless the order value *a* = 0.

A:

Initial concentrations:

	Exp 1	Exp 2	Exp 3
$[A]_0$	10 M	7.5 M	5 M
$[B]_0$	10 M	10 M	10 M
$[cat]_0$	0.5 M	0.5 M	0.5 M

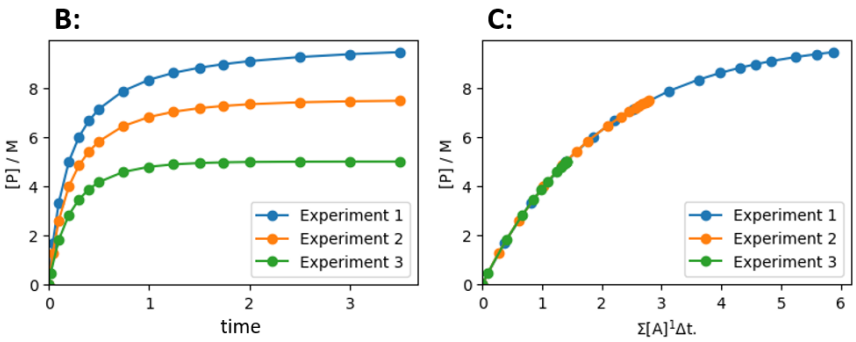
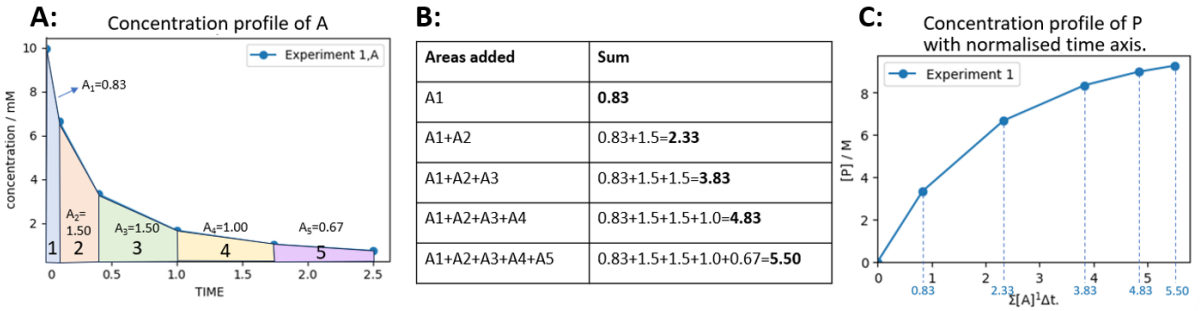


Figure S1: Simulated different excess data for a reaction A + B → P catalysed by cat, with the initial concentrations shown in subfigure A. The ODEs used to generate the data are d/dt([A])= d/dt([B])=- d/dt([P])=-k[A]*[B]*[cat] with a 4th order Runge-Kutta solver. B: Simulated concentration profiles for the formation of P for experiments 1, 2 and 3. C: VTNA overlay plot for the formation of P with a time axis normalised with respect to the concentration profile of A.*

However, changing the variable of integration from t to $[A]^a(t)t$, results in a new integrand $k[B]^b[cat]^c$ which is the same for each experiment as the initial concentrations of B and cat are unaltered between runs. This means that a new time axis $\int_0^{t_n} f(t)dt = \int_0^{t_n}[A]^a(t)dt$ rather than $\int_0^{t_n} dt = t_n$ is created along which the product concentration profiles will overlay, as long as the correct order in A is used. Since the analytical form of $[A](t)$ is unknown, the normalised time axis is obtained by numerical integration using measurements of $[A]$ over time, either inferred from the product concentration profile or preferably measured directly (Figure S2). The product concentration profiles from experiments with different $[A]_0$ are projected perfectly onto the same curve when the effect of different $[A]_0$ on the product concentration profiles is normalised for by this integration method (Fig. S1C).

$$\frac{d[P]}{dt} = k[cat]^c[A]^a[B]^b \therefore \frac{d[P]}{dt[A]^a} = k[cat]^c[B]^b dt \therefore \frac{d[P]}{df(t)} = k[cat]^c[B]^b$$
$$\therefore d[P] = k[cat]^c[B]^b df(t) \qquad (Eq.S4)$$



Areas added	Sum
A1	**0.83**
A1+A2	0.83+1.5=**2.33**
A1+A2+A3	0.83+1.5+1.5=**3.83**
A1+A2+A3+A4	0.83+1.5+1.5+1.0=**4.83**
A1+A2+A3+A4+A5	0.83+1.5+1.5+1.0+0.67=**5.50**

Figure S2: A): Concentration profile of A from the simulated kinetic data labelled "experiment 1" in Figure S1 with 6 selected datapoints. The trapezoid areas between each datapoint are illustrated and labelled 1-5. B): Table showing how areas are added to give rise to the transformed time axis $\sum[A]^1\Delta t$. *C): The concentration profile of the product P plotted with the transformed time axis* $\sum[A]^1\Delta t$. *This is the same concentration profile as in Figure S1 C but with less dense representation of the simulated kinetic data.*

The same logic can be extended to experiments in which the initial concentration of more than one reaction species is altered. For example, if both the initial concentrations of A and B are varied, the transformed time axis will be given by the time integral of $f(t) = [A]^a[B]^b$. Furthermore, if the initial concentrations of every reaction species are varied, the time axis can be replaced by the integral of $f(t) = [A]^a[B]^b[cat]^c$. In practice, integrating $f(t)$ requires the concentration profiles of the reaction species for which the time axis is normalised. Although the analytical form of these concentration profiles is unknown, the integral can be estimated from concentration measurements made during the reaction using the trapezoid rule. This is shown below for normalising the time axis with respect to A (Eq. S5) and with respect to A, B, and cat (Eq. S6).

$$df(t) = dt[A]^a \therefore f(t_n) = \int_0^{t_n} [A]^a(t)dt \approx \sum_{i=1}^{n} \left(\frac{[A]_i + [A]_{i+1}}{2}\right)^a (t_i - t_{i-1}) \qquad (Eq.S5)$$

$$df(t) = dt[A]^a[B]^b[cat]^c \therefore f(t_n) = \int_0^{t_n} [A]^a[B]^b[cat]^c dt$$
$$\approx \sum_{i=1}^{n} \left(\frac{[A]_i + [A]_{i+1}}{2}\right)^a \left(\frac{[B]_i + [B]_{i+1}}{2}\right)^b \left(\frac{[cat]_i + [cat]_{i+1}}{2}\right)^c (t_i - t_{i-1}) \qquad (Eq.S6)$$

2. Methods to quantify concentration profile overlay

2.1. Variance of fitted concentration profiles

During the development of Auto-VTNA, a different method to evaluate concentration profile overlay to that described in the main paper was explored.

The alternative method involved rescaling the transformed time axis between 0 to 1. A 4th order polynomial was then employed to fit each concentration profile. The choice of a 4th degree polynomial was made to balance flexibility in capturing the true concentration profile shape without introducing over-fitting effects (Figure S3). Subsequently, fitted values across an evenly spaced range of transformed time values were grouped by transformed time value (Figure S4B). Any fitted value lying beyond the transformed time domain of the corresponding concentration profile was excluded. The variance of the fitted curves across a range of transformed time points was then calculated and the average variance defined as the overlay score (Figure S4 and S5). If concentration profiles overlay, their fitted curves would overlay as well, giving a low average variance. In contrast, poor concentration profile overlay would give a high average variance as the fitted functions would occupy a range of different output concentrations for any transformed time value. This is illustrated by the average variance of 0.0022 for incorrect order values of 0 in both reactants **1** and **2** in the Aza Michael reaction in EtOH (Figure S4)[1], and the average variance of 0.000083 for the correct order values of 1 in both reactants (Figure S5).

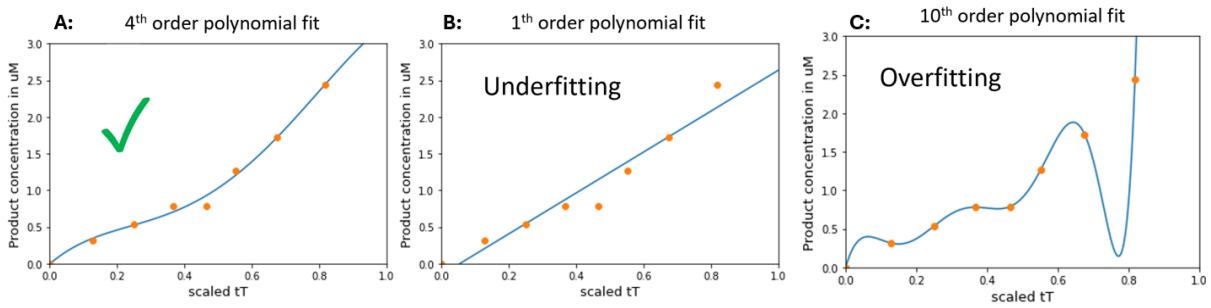
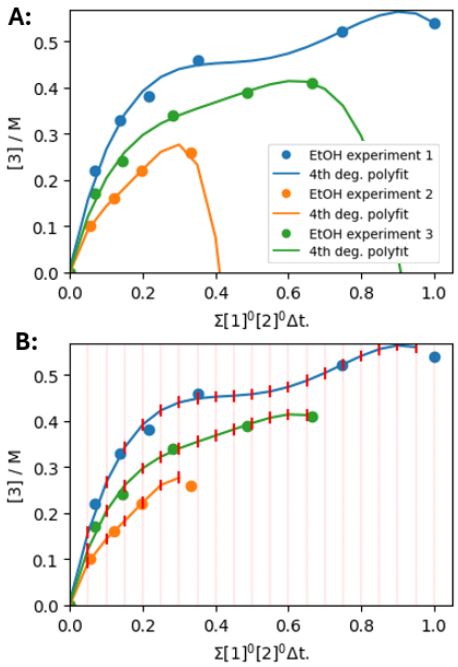


Figure S3: The initially developed overlay score involved fitting each concentration profile using a polynomial. A: A 4th degree polynomial fit was found to avoid both under- and overfitting. B): A linear fit led to underfitting, i.e. the true shape of the profile was not captured. C): Using overly high degree polynomials such as 10th led to overfitting.

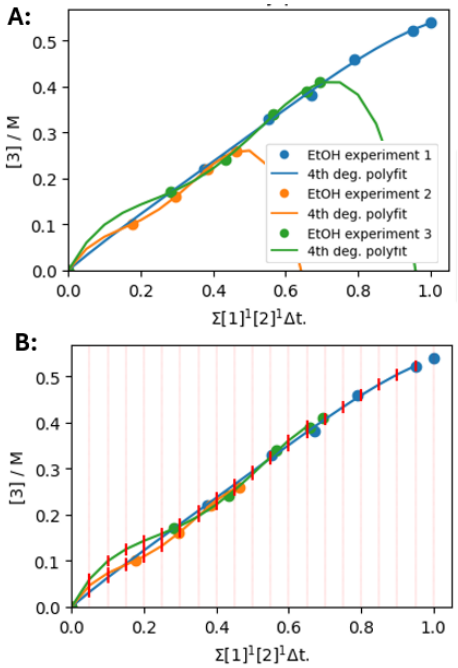


C:

Range of tT values:	Corresponding fit values Exp 1	Exp 2	Exp 3	Variance:
0.0	[0.009,	0.001,	0.004]	1e-05
0.05	[0.158,	0.092,	0.122]	0.00073
0.1	[0.266,	0.143,	0.204]	0.00252
0.15	[0.342,	0.182,	0.261]	0.00427
0.2	[0.392,	0.223,	0.297]	0.00478
0.25	[0.423,	0.261,	0.322]	0.00446
0.3	[0.441,	0.276,	0.341]	0.00461
0.35	[0.448,	0.354]		0.00221
0.4	[0.452,	0.368]		0.00176
0.45	[0.454,	0.381]		0.00133
0.5	[0.458,	0.395]		0.00099
0.55	[0.464,	0.407]		0.00081
0.6	[0.473,	0.414]		0.00087
0.65	[0.487,	0.412]		0.00141
0.7	[0.504]			
0.75	[0.522]			
0.8	[0.541]			
0.85	[0.556]			
0.9	[0.564]			
0.95	[0.561]			
1.0	[0.561]			

Average variance: 0.00220

Figure S4: Overview of the first method trialled for overlay assignment. A): VTNA overlay plot for Experiments 1, 2 and 3 from kinetic data on the Aza Michael reaction (Scheme 1).[1] Incorrect order values of 0 have been used to illustrate how this overlay score functions. B): Same VTNA overlay plot with truncated fit functions. Red vertical lines represent the tT-value grid used for evaluating the fit functions. The dark red lines show which values were collected from each fit. C): Fit values at each tT and the corresponding variance. No variance is calculated at tT>0.7 as only the time normalised RM3 covers this domain.



C:

Range of tT values:	Corresponding fit values Exp 1	Exp 2	Exp 3	Variance:
0.0	[0.0,	0.0,	0.0]	0.000000
0.05	[0.032,	0.047,	0.061]	0.000140
0.1	[0.063,	0.073,	0.099]	0.000230
0.15	[0.093,	0.091,	0.125]	0.000243
0.2	[0.122,	0.109,	0.143]	0.000196
0.25	[0.151,	0.132,	0.159]	0.000128
0.3	[0.182,	0.162,	0.175]	0.000069
0.35	[0.208,	0.197,	0.196]	0.000030
0.4	[0.236,	0.231,	0.222]	0.000034
0.45	[0.265,	0.256,	0.253]	0.000026
0.5	[0.294,	0.288]		0.000009
0.55	[0.322,	0.325]		0.000002
0.6	[0.351,	0.361]		0.000025
0.65	[0.379,	0.391]		0.000036
0.7	[0.406]			
0.75	[0.433]			
0.8	[0.459]			
0.85	[0.482]			
0.9	[0.504]			
0.95	[0.523]			
1.0	[0.542]			

Average variance: 0.000083

Figure S5: Overview of the first method trialled for overlay assignment with the correct reaction order values of 1 in both reactants for the Aza Michael reaction (Scheme 1).[1] Equivalent to Figure S4.

The average variance overlay score was sufficiently reliable to enable the calculation of the optimal order values for the kinetic data on the Aza Michael reaction (Scheme 1).[1] However, the method suffered from significant drawbacks:

1. The overlay score was biased towards reaction order values which compress output concentration profiles to different transformed time domains. This is because the shorter a concentration profile is relative to the longest, the fewer the time points for which it contributes to the variance score (Experiment 2 < Experiment 3 < Experiment 1 in Figure S4B).
2. The greater time domain to which only the longest time normalised concentration profile reaches (0.7 to 1.0 in Figure S4), the lower the number of large variance values that contributes to the average variance.
3. The number of datapoints in a concentration profile does not effect the overlay score (only the fitted curve directly effects the overlay score). This makes comparison between overlay scores for different density datasets challenging.
4. It is more computationally expensive than the overlay score utilized in Kinalite or the total fitting overlay score implemented in the final version of Auto-VTNA.[2]

Consequently, the method was abandoned in favour of the total fitting method described in the main publication.

2.2. The Kinalite overlay score

An alternative method for computationally assessing overlay, utilised by Kinalite, a Python package for automatically deriving the order values maximising concentration profile overlay, calculates an overlay score ("error").[2] This score is obtained by sorting the data points from every time normalised concentration profile according to their normalised time value and summing the concentration difference between adjacent datapoints (Fig. S6). If concentration profiles do not overlay, these concentration differences will be large as neighbouring datapoints will belong to curves with different y-values. However, if the profiles overlay, the concentration differences are limited to the gradual increase of the single curve onto which the profiles overlay.

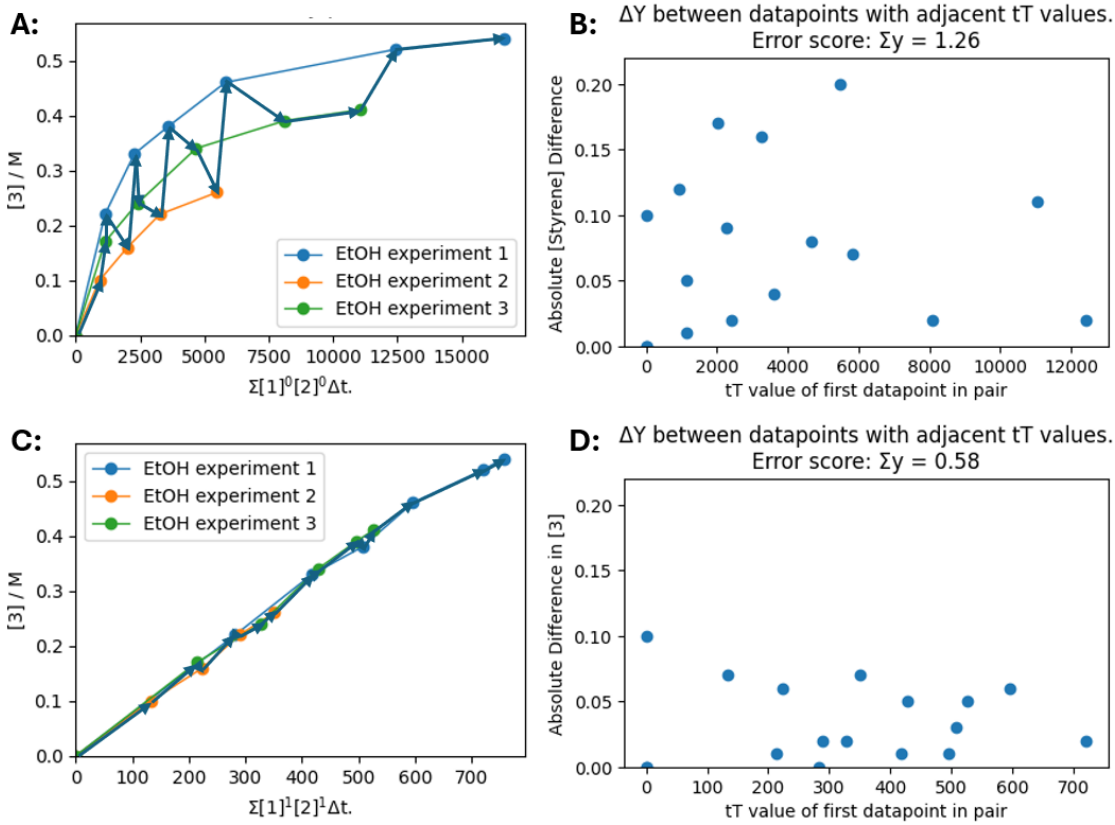


*Figure S6: Illustration of the principle behind Kinalite's overlay score. A: VTNA overlay plots for the aza Michael addition of **2** to **1** with incorrect reaction order values of 0 with dark blue arrows illustrating the y-difference between datapoints with adjacent x values. C: Corresponding VTNA overlay plot for the correct order values (1 in both reactants). B and D: Absolute values for the y-value difference between datapoints with adjacent x value for overlay plots A and C respectively.*

Although often identifying the best reaction orders, Kinalite's overlay score is biased towards reaction orders which separate concentration profiles into separate transformed time domains. This lowers the average difference between neighbouring datapoints as most of these points will belong to the same reaction progress profile. This gives rise to erratic overlay score versus order plots and can cause incorrect order values to be outputted for challenging kinetic data.[2]

2.3. The total polynomial fit overlay score

As outlined in the main paper, the most useful overlay score explored in the development of Auto-VTNA involves fitting all the relevant normalised reaction profiles to the same flexible function and quantifying the goodness of fit. We opted for a polynomial fitting procedure as the polynomial degree can be modulated easily. The degree of the polynomial should be sufficient to avoid both under and over-fitting (see Section S6). Using the Polyfit python package[3], monotonic fitting could also be implemented, as chemical reaction species concentrations are generally monotonic, to limit the effect of overfitting on the overlay score at incorrect reaction orders (see below).

A standard goodness-of-fit measure must be defined to enable the comparison of overlay scores across different datasets as well as for different order values within a dataset. The standard error of regression was considered as it reduces with the number of datapoints like the standard error in the mean, thereby avoiding bias towards sparse kinetic data. However, this becomes a disadvantage when concentration profiles of varying datapoint density are compared. For instance, dense kinetic data acquired using an in-situ IR probe will yield a lower minimum standard error compared to less dense kinetic data obtained by GC or HPLC, even when the profile overlay remains consistent. RMSE was considered the superior measure of goodness-of-fit for quantifying concentration profile overlay, as it captures the average difference between predicted and actual datapoint values irrespective of the number of datapoints.

3. Preparing the kinetic data for Auto-VTNA.

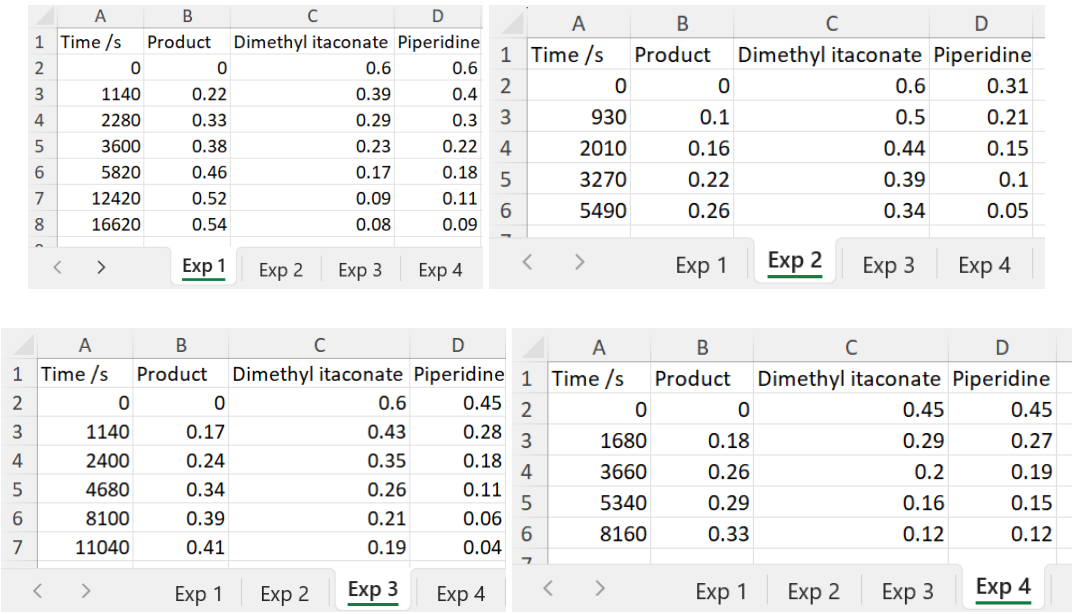
3.1. Collecting kinetic data

Auto-VTNA is a kinetic analysis technique for deriving reaction orders from concentration time data based on VTNA. Appropriate kinetic data can be collected by a range of different methods such as *ex situ* HPLC or GC, *in situ* IR, or by setting up reaction repeats and measuring the concentrations of products and/or reactants following workup at different times. For relevant examples, please see the literature examples referenced in Table 2. The datasets used to derive the results in Table 2 can also be accessed on the Auto-VTNA Github repository.[28]

To derive the reaction order in a reaction species, its concentration profiles in at least two different excess experiments must be tabulated (see section S1). If available, measured concentration values should be utilized. Otherwise, the concentration profile of a reactant could be inferred from its known initial concentration and the measured concentration profile of the product or another reactant, assuming perfect mass balance. Additives and catalysts that are not consumed during the reaction are most often assumed to remain constant during the reaction. For examples of how VTNA datasets have been prepared in previous studies, see Table S7.

3.2. Importing the kinetic data

To utilize Auto-VTNA, the kinetic data should be organized into a .xlsx file in Excel with concentration time data from experiments separated into different sheets with appropriate names. Each sheet should contain the time axis in column A and the concentration profiles of different reaction species in the next columns (Fig. S7).



	A	B	C	D
1	Time /s	Product	Dimethyl itaconate	Piperidine
2	0	0	0.6	0.6
3	1140	0.22	0.39	0.4
4	2280	0.33	0.29	0.3
5	3600	0.38	0.23	0.22
6	5820	0.46	0.17	0.18
7	12420	0.52	0.09	0.11
8	16620	0.54	0.08	0.09

Exp 1 | Exp 2 | Exp 3 | Exp 4

	A	B	C	D
1	Time /s	Product	Dimethyl itaconate	Piperidine
2	0	0	0.6	0.31
3	930	0.1	0.5	0.21
4	2010	0.16	0.44	0.15
5	3270	0.22	0.39	0.1
6	5490	0.26	0.34	0.05

Exp 1 | Exp 2 | Exp 3 | Exp 4

	A	B	C	D
1	Time /s	Product	Dimethyl itaconate	Piperidine
2	0	0	0.6	0.45
3	1140	0.17	0.43	0.28
4	2400	0.24	0.35	0.18
5	4680	0.34	0.26	0.11
6	8100	0.39	0.21	0.06
7	11040	0.41	0.19	0.04

Exp 1 | Exp 2 | Exp 3 | Exp 4

	A	B	C	D
1	Time /s	Product	Dimethyl itaconate	Piperidine
2	0	0	0.45	0.45
3	1680	0.18	0.29	0.27
4	3660	0.26	0.2	0.19
5	5340	0.29	0.16	0.15
6	8160	0.33	0.12	0.12

Exp 1 | Exp 2 | Exp 3 | Exp 4

Figure S7: Screenshot of a representative .xlsx file which can be used as input for Auto-VTNA.

To avoid errors, ensure that the concentration profile of a given reaction species appears in the same column on each sheet of the .xlsx file. If the concentration profile for a reaction species is unknown or has not been measured for a given experiment, leave it empty. However, if this reaction species is selected as a normalized reaction species for a calculation involving the relevant experiment sheet, the calculation will fail and produce an error. It is generally recommended that users calculate all reaction species for each experiment, as this enables total VTNA calculations. Alternatively, if only the product and reactant with varied initial concentrations are known, you can prepare separate .xlsx files for each reaction species and perform automatic sequential VTNA on these.

To upload the .xlsx file to the Auto-VTNA Calculator, the user must click browse and select the relevant .xlsx file (see Section S5.1).

If Auto-VTNA is being used without its GUI, the .xlsx file should be imported using pd.read_excel():

```
kinetic_data=pd.read_excel('filename.xlsx',sheet_name=None)
```

Setting sheet_name to “None” ensures that the kinetic_data variable is a dictionary with keys corresponding to the different experiments. The values of the experiment keys will be defined as pandas dataframe objects with the same Table structure as in the original .xlsx file.

It is important that there are no errors in the .xlsx file as this will cause Auto-VTNA to fail. For example, no non-numerical values should be in any sheet expect for the column titles in row 1. Other requirements on the input data are as follows:

- The column headers in row 1 should be identical for each sheet.
- The time column needs to come first, i.e. in column A.
- The time column should be monotonically increasing.
- There should not be any empty sheets or sheets with values in incorrect position.

To aid this process, a function called “check_kinetic_data” has been written. The function checks that the kinetic data is loaded correctly by several criteria and generates a report which specifies any errors in the .xlsx file so help the user resolve the issue. This function should be applied first time a .xlsx file is being analysed by Auto-VTNA (Fig. S8).

```
Kinetic data check report.
--------------------------
Check 1: Correct loading of kinetic data from Excel.
  ✓ The kinetic data is loaded as a dictionary.
  ✓ Each element associated a key in the kinetic data dictionary is a pandas dataframe object.
  ✓ The dataframe element(s) in the kinetic data dictionary all contain 4 columns.
Check 2: Column title consistency.
  ✓ All experiment datasets have the same column titles.
Check 3: Numerical datapoints.
  ✓ Every experiment dataset contains only numericalvalues.
Check 4: Positive datapoints.
  ✓ No datasets contain negative time or concentration values.
Check 5: Time column monotonicity.
  ✓ The time column (first column) of the dataset in every sheet is monotonically increasing.
Check 6: Time and concentration columns.
  ✓ All dataframe elements in the kinetic data dictionary has column titles which are strings.
  ✓ The column titles of every experiment dataset are all strings.
NB: Make sure that the time columns are the first columns in each dataframe in kinetic_data.
Currently, the first column titles are: Time /s.
Conclusion: The kinetic data has been loaded correctly if the time axis label is correct.
```

Figure S8: Report generated by check_kinetic_data() using a correctly loaded and formatted .xlsx file.

4. User guide for the Auto-VTNA Python Code

Users intending to utilize Auto-VTNA *via* its GUI (the Auto-VTNA Calculator) may skip to Section S5. In general, **the GUI is recommended** as it offers more features and is written for optimal user-friendliness.

4.1. Visualising the kinetic data

Once the kinetic data has been loaded successfully into Jupyter Notebook or a code editor like VSCode, it can be visualised in several ways using functions located in the __init__.py file of the package (accessed by typing auto_vtna.function_name()). For example, plot_data() generates graphs showing the concentration profiles in each experiment. The graphs can be generated side-by-side in one window by setting the plot_mode argument to 'together'. Alternatively, the graphs can be generated in an interactive manner where the user can use the right and left arrow keys to scroll through the experiments (plot_mode='scrollable'). Another function plot_data_MB() also exists which can be used to visualise the same data but with an added mass balance curve on a secondary axis specified by a list of reaction species names to include and another list of their corresponding stoichiometry (Fig. S9).

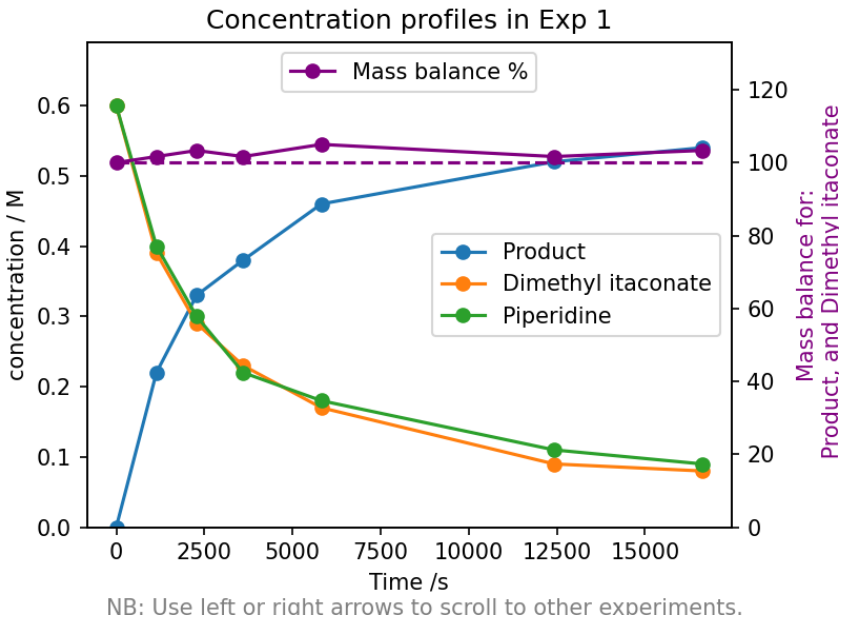


Figure S9: Graph visualising the product, dimethyl itaconate and piperidine concentration profiles (left axis) as well as the mass balance in dimethyl itaconate (right axis).

To visualize all the concentration profiles for a selected number of reaction species across all experiments, plot_data_together() can be used (Fig. S10). As well as the kinetic data variable, this function inputs a list of reaction species to be included (one species is recommended to avoid a cluttered plot). A function initial_concs()can be used to generate a Table containing the initial concentrations of every reaction species across all experiments. This can be useful to get an overview of the contents of each experiment dataset.

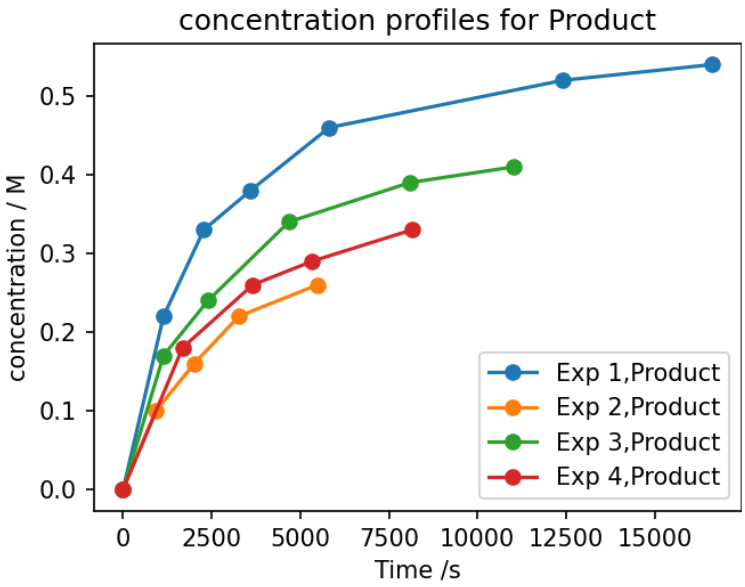


Figure S10: Concentration profiles of the product across all experiments generated using plot_data_together().

Each of these data visualisation functions discussed above contain a range of keyword arguments to specify units, figure size, number of decimal places, titles etc. and the user is advised to consult the __init__.py file of the package for more information.

4.2. The VTNA selection dictionary

To perform automatic or normal VTNA, a VTNA selection dictionary is needed. The selection dictionary contains information necessary to carry out variable time normalisation such as which experimental datasets to analyse ('RM' for reaction mixture) and whether some datapoints or datapoint ranges should be omitted from the analysis. The omissions can either be provided as a list of datapoints or as a range of values where 100 is defined as the highest time ('range' followed by two numbers such as 50 and 100 to remove the second half of the datapoints). The selection dictionary also specifies which reaction species to use for normalising the time scale ('normalised species') with their respective reaction orders. The output species of the calculation is also specified ('output species'). A selection dictionary template can be generated using the function make_VTNA_selection() for the given dataset (Fig. S11A) which can be completed by specifying the experiments, any datapoint or datapoint range omissions, the output species and the normalised species, with their respective order values (Fig. S11B).

A:

```
auto_vtna.make_VTNA_selection(data)
```

```
{'RM': {'Exp 1': {'omissions': None},
  'Exp 2': {'omissions': None},
  'Exp 3': {'omissions': None},
  'Exp 4': {'omissions': None}},
 'output_species': 'OS',
 'normalised_species': {'NS': 1}}
```

B:

```
selection
```

```
{'RM': {'Exp 1': {'omissions': [4]},
  'Exp 2': {'omissions': None},
  'Exp 3': {'omissions': None}},
 'output_species': 'Product',
 'normalised species': {'Dimethyl itaconate': 1,
  'Piperidine': 1}}
```

Figure S11: A: VTNA selection dictionary for the example dataset in Figure S9 generated automatically using the function make_VTNA_selection(). B: Properly filled out VTNA selection dictionary for the same dataset with experiments, output species and normalised species specified.

4.3. Normal VTNA

While the Auto-VTNA functions are readily available in the __init__.py file of the package, accessible through importing auto_vtna and using auto_vtna.function_name(), the conventional VTNA code is structured as a class within a distinct Python file named Normal_VTNA.py. By conventional VTNA, it is meant that the output of Normal_VTNA contains the normalised time axis ('tT') of each experiment calculated using the order values and normalised species given by the VTNA selection dictionary (Fig. S12).

```
from auto_vtna.Normal_VTNA import Normal_VTNA
VTNA_etoh=Normal_VTNA(data,selection)
VTNA_etoh.result
```

Exp 1:

	Time /s	Product	Dimethyl itaconate	Piperidine	tT
0	0	0.00	0.60	0.60	0.000
1	1140	0.22	0.39	0.40	282.150
2	2280	0.33	0.29	0.30	417.810
3	3600	0.38	0.23	0.22	507.042
4	12420	0.52	0.09	0.11	739.890
5	16620	0.54	0.08	0.09	775.590

Exp 2:

	Time /s	Product	Dimethyl itaconate	Piperidine	tT
0	0	0.00	0.60	0.31	0.0000
1	930	0.10	0.50	0.21	132.9900
2	2010	0.16	0.44	0.15	224.3580
3	3270	0.22	0.39	0.10	289.7205
4	5490	0.26	0.34	0.05	350.4930

Exp 3:

	Time /s	Product	Dimethyl itaconate	Piperidine	tT
0	0	0.00	0.60	0.45	0.0000
1	1140	0.17	0.43	0.28	214.2915
2	2400	0.24	0.35	0.18	327.3135
3	4680	0.34	0.26	0.11	428.1465
4	8100	0.39	0.21	0.06	496.4610
5	11040	0.41	0.19	0.04	525.8610

Figure S12: The output of a Normal VTNA calculation is found in the rightmost column 'tT' which can be used to generate a VTNA overlay plot (see below).

Once the instant Normal VTNA calculation has been run, the result can be visualised by calling the method .plot_VTNA() to generate a conventional VTNA overlay plot. 13 keyword arguments can be defined to customize the plot such as "size_scaler", "linewidth", "xtick_rotation" etc. (see documentation for more details). Alternatively, the plot_VTNA_with_overlay_score() method can be called to generate the overlay plot together with a line of best fit, either an ordinary or monotonic polynomial of a selected degree, or a straight line through origin. The selected overlay score, set to RMSE by default, will then be shown in the graph title (Fig. S13). If a degree of 1 has been defined, the title will also contain the slope of the fitted line with its standard error uncertainty. This slope can be useful for determining the observed rate constant when linearisation has occurred (Fig. S13C)

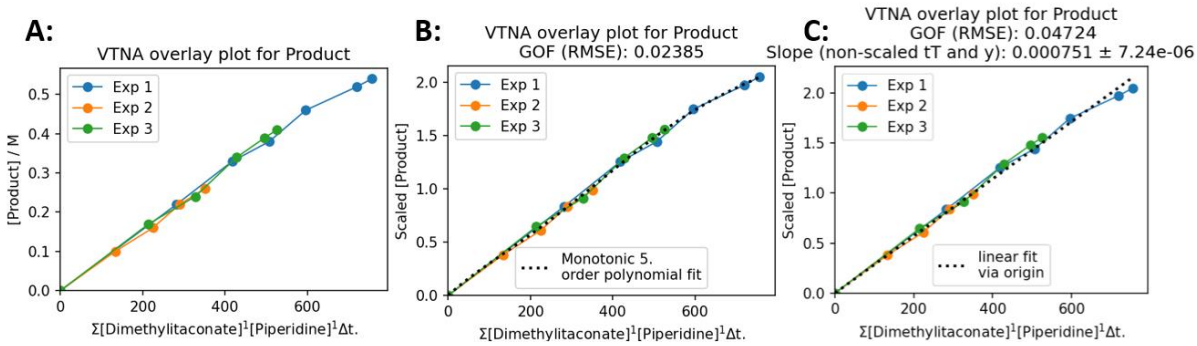


Figure S13: VTNA overlay plots generated by calling plot_VTNA() (A) or plot_VTNA_with_overlay_score() with deg=5 and constraint='monotonic' (B) or deg=1 and constraint='through origin' (C).

4.4. Automatic VTNA

As for Normal_VTNA, the automatic VTNA code is structured as a class within a distinct Python file in the Auto_VTNA package named Automatic_VTNA.py. An instance of Automatic VTNA is initiated by inputting the kinetic data and the VTNA selection dictionary to define the output reaction species and the normalised reaction species for which the optimal order values will be determined.

Nine keyword arguments can be specified to modify the settings of the calculation such as the number of iterations, the range of order values to investigate and the resolution of the order value mesh (see documentation for more details). The calculation will then run, and the best order values of each iteration printed in the terminal. Once the calculation is complete, the results can be viewed via the result, best_orders, interval, and best_slope attributes of the Automatic_VTNA object:

- The result attribute contains the overlay scores obtained across all reaction order combinations.
- The interval attribute contains the 15% uncertainty intervals for each normalised reaction species as discussed in the main publication.

If the calculation was set to elucidate the order values in 1 or 2 reaction species, the results can be visualized by calling the plot_orders_vs_overlay() method of the Automatic_VTNA class (see Fig. 3 and 7).

To customise the Figures, 16 different keyword arguments such as “zoom”, “y_unit” and “interval” can be adjusted. The graph or contour plots generated by plot_orders_vs_overlay() are interactive and will generate the overlay plot corresponding to the order values of the click point , either with (right-click) or without (left-click) the overlay score and fitted polynomial (see Fig. 4).

4.5. Automatic VTNA with fixed orders

For an automatic VTNA calculation, any number of normalised reaction species can be selected in the VTNA selection dictionary as long as its name corresponds to a column in the uploaded kinetic dataset (dictionary of Pandas dataframe objects). However, a list of "fixed order species" can also be provided which will be fixed at the order value specified in the VTNA selection dictionary. Instead of trialling a range of order values for the fixed reaction species, the time axis will remain normalised with these reaction species at the specified order value, and rather optimise the order values of the non-fixed normalised species. As mentioned in the main publication, this effectively reduces the dimensionality of the automatic VTNA calculation.

As an example, the correlation between overlay score and orders in reactants **14** and **15** for the entire kinetic data on the reaction system in Entry 3 in Table S2 can be visualised by fixing the catalyst order to 1 (Fig. S14A). Likewise, the correlation between overlay score and orders in reactants **18** and **19** for the entire kinetic data on the reaction system in Entry 4 in Table S2 can be visualised by fixing the catalyst and base orders to 1 and 0 respectively (Fig. S14B).

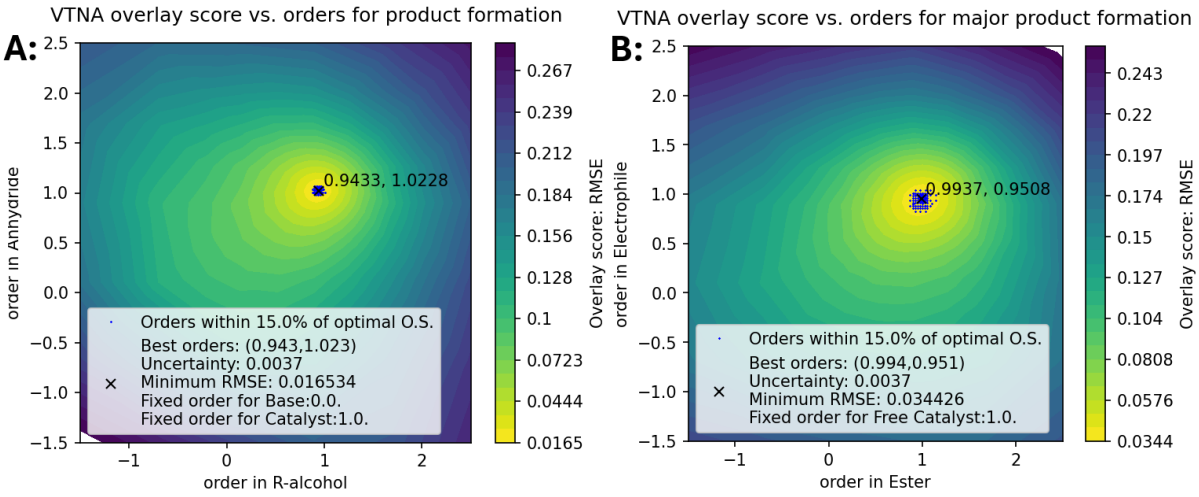


Figure S14: Overlay score vs. reactant orders for the kinetic data on the reaction systems in Entries 3 (A) and 4 (B) in Table S2, obtained via automatic VTNA calculations on the entire dataset in which the catalyst orders were set to the previously calculated value of 1 and the base order fixed at 0 to generate contour plot A. See additional information in the figure captions.

5. User guide for the Automatic VTNA Calculator GUI

A graphical user interface (GUI) named the "Automatic VTNA Calculator" has been developed to streamline the use of the features of the Auto-VTNA Python package without the need for coding knowledge.

The following sections outline the recommended steps for carrying out automatic VTNA using the Auto-VTNA Calculator. The workflow can be summarized into four essential steps:

1. Upload a correctly formatted .xlsx file.
2. Select experiments, normalised reaction species for which orders should be derived and an output reaction species. (Note, initially the "quick setting" is best used if more than 4 reaction species are investigated).
3. Click run under "Automatic VTNA" to calculate the order value which optimises the overlay score for each normalised species.
4. Visualise the results of the calculations via "VTNA Overlay Plots" or by clicking "Plot Results" if 1 or 2 normalised reaction species have been selected.

A YouTube tutorial has also been produced running through these steps, which can be found linked to the GitHub: https://github.com/ddalland/Auto-VTNA.

The following sections go into more details of each feature of the Automatic VTNA calculator.

NOTE: Be aware that the results from an automatic VTNA calculation are only meaningful if the following criteria have been met:

- The different sheets in the .xlsx file corresponds to experiments carried out with identical reaction conditions apart from varying initial concentrations of reactants, additives or catalysts.
- To obtain a meaningful order value in a selected normalised reaction species, experiments with different initial concentrations in the reaction species must be selected.
- The kinetic data must be of sufficiently high data point density for accurate numerical integration (at least 5 data points per concentration profile is recommended).
- An appropriate total fitting method must be selected (see Section S6).

5.1. Uploading the Kinetic Dataset

Once the Auto-VTNA Calculator executable file has been run, the dashboard of the Auto-VTNA Calculator is activated by uploading the kinetic data as an .xlsx file (browse button) or as CSV files (Use CSVs). This automatically populates the "Select Experiments", "Select Normalised Species" and "Select Output Species" boxes (Fig. S15).

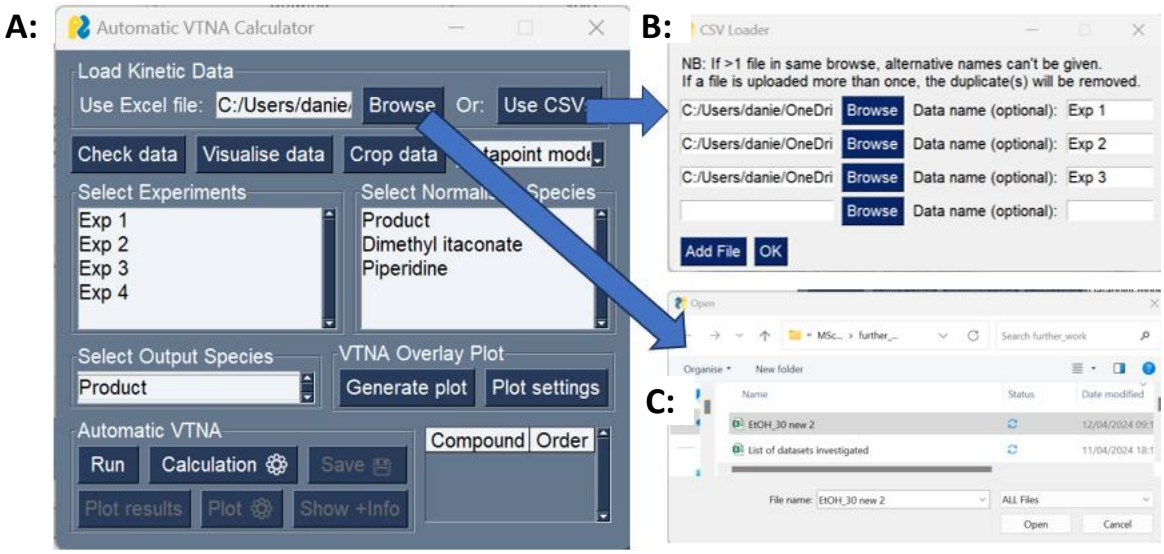
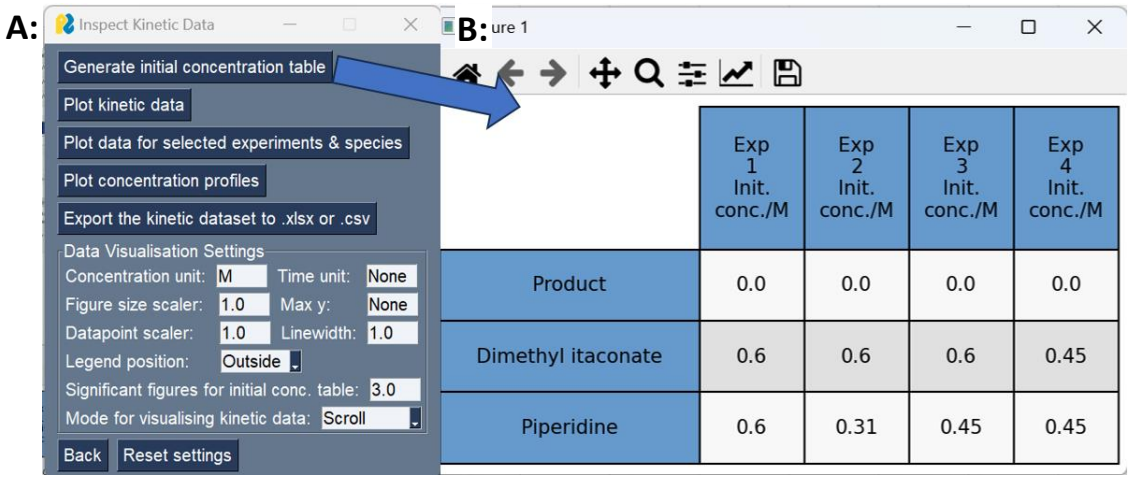


Figure S15: Uploading kinetic data onto the Automatic VTNA dashboard (A). This can be done either by selecting the relevant .xlsx file (C) or by clicking use CSVs and selecting the desired number of CSV files (C).

5.2. Inspecting the kinetic data

Once the kinetic data has been uploaded, it can be checked by the "Check Data" button or visualised in different ways by clicking "Visualise Data". For example, an initial concentration Table can be generated by clicking "Visualise Data" followed by "Generate Initial Concentration Table" (Figure S16). The kinetic data can also be visualised in different ways to generate plots as shown in Section 4.1.



	Exp 1 Init. conc./M	Exp 2 Init. conc./M	Exp 3 Init. conc./M	Exp 4 Init. conc./M
Product	0.0	0.0	0.0	0.0
Dimethyl itaconate	0.6	0.6	0.6	0.45
Piperidine	0.6	0.31	0.45	0.45

Figure S16: The "Inspect Kinetic Data" menu generated by clicking Visualise Data (A). From this sub-window, several different Auto-VTNA functions can be accessed, such as generation of an initial concentration Table (B).

Several settings can also be adjusted such as the position of legends, the size of Figures and Concentration units, providing more user-friendly access to the keyword arguments of the underlying Auto-VTNA functions.

5.3. Automatic VTNA Calculations

To perform automatic VTNA calculations or generate VTNA overlay plots, the user must select:

- the normalised reaction species;
- the experiment datasets to include; and
- the output reaction species (normally the product).

Once these inputs have been selected, ordinary VTNA overlay plots can be generated by clicking "Generate Plot" within the "VTNA Overlay plot" frame (Fig. S17). If desired, normal VTNA can be performed by then manually altering order values for the reaction species.

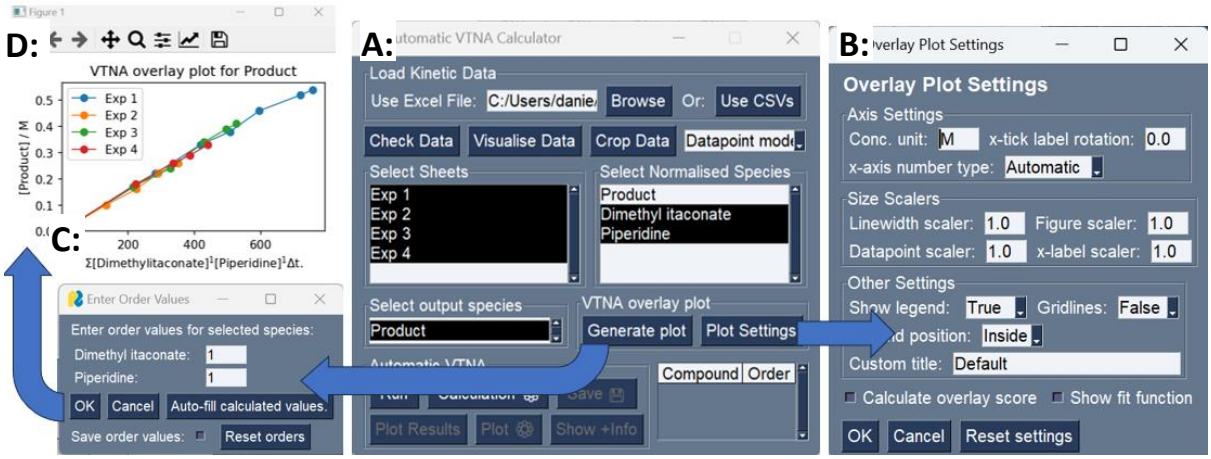


Figure S17: Illustration of how the Automatic VTNA Calculator (A) can be used to generate overlay plots (D) by specifying the relevant order values (C). The VTNA overlay plots can be modified via the "Plot Settings" menu (B).

The reaction order values of the "Selected Normalised Species" that maximise concentration profile overlay can be calculated automatically by clicking "Run" in the "Automatic VTNA" frame. Once the calculation has been completed, the results can be visualised by clicking plot results. This will generate an overlay score *vs.* order value graph if one normalised species has been defined, or a contour plot if two normalised species have been defined.

The Automatic VTNA settings can be accessed and modified by clicking on the calculation and plot setting buttons. The calculated orders will also appear in the bottom left Table. Moreover, the order value intervals and overlay score of the best order values can also be shown by clicking on the "Show + Info" button (Fig. S18). The percentage cutt-off defining the order value intervals can also be updated to generate a new table.

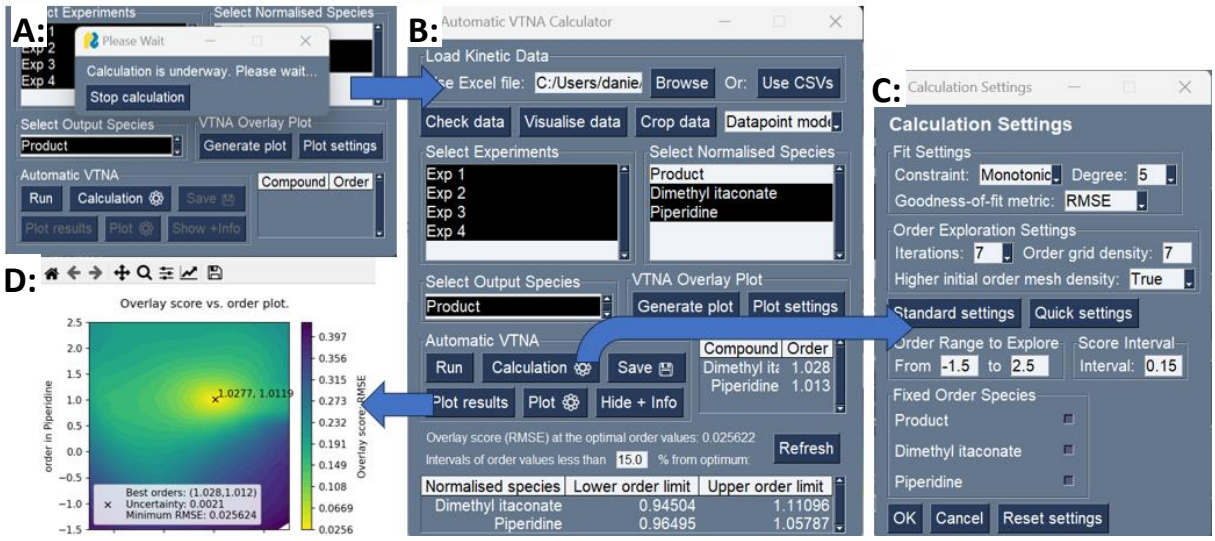


Figure S18: After running an automatic VTNA calculation with two normalised reaction species (A), the results can be visualised by clicking "Plot Results" to generate an overlay score versus order contour plot (D). The calculation and plot settings buttons can be used to alter the standard keyword arguments of the Automatic_VTNA class and its plotting methods (C).

5.4. Quick and Standard settings for the Automatic VTNA Calculation

The time taken for an Automatic VTNA calculation depends on several settings such as the number of algorithm iterations, the resolution of the order value grid of each resolution (see Section S7). **To minimise the calculation time,** a "Quick Settings" button has been included in the calculation settings menu which applies iteration and order grid density parameters that optimise the processing time. Alternatively, the "Standard Settings" box can be clicked to apply settings that are slower but cover a denser range of order value combinations in each iteration. The settings menu can also be used to specify the range of order values to investigate, and fixed order values for any of the normalised reaction species to lower the dimensionality of the calculation as discussed in Section S7.

5.5. Saving the Results from an Automatic VTNA Calculation

To save the results from an automatic VTNA calculation using the Automatic VTNA Calculator, the button "Save" can be clicked. The resulting .xlsx or .csv file will contain the applied automatic VTNA settings, the overlay score of every trialled order value combination as well as the uncertainty intervals. If the results are saved as a .xlsx file, the most recently generated overlay *vs.* order plot will also be included.

5.6. Cropping the Kinetic Data

If necessary, parts of the loaded kinetic data can be removed before performing automatic VTNA by clicking the "Crop" button. If all experiments contain less than 25 time points, the default cropping setting is set to "Datapoint mode" in which the datapoints of each experiment are numbered and can be ticked for removal (Fig. S19B). Alternatively, "Range mode" can be selected, giving rise to a menu in which ranges of time values (either absolute values or as a percentage of t_{end}) can be selected for

each experiment within which all datapoints are removed (Fig. S19C). Range mode also allows the datapoint density of experiments to be reduced by only keeping every X datapoint, useful for highly dense datasets which give rise to slower automatic VTNA calculations. The cropping applied by these methods is reversible and can be removed by clicking “Reset” followed by “OK”.

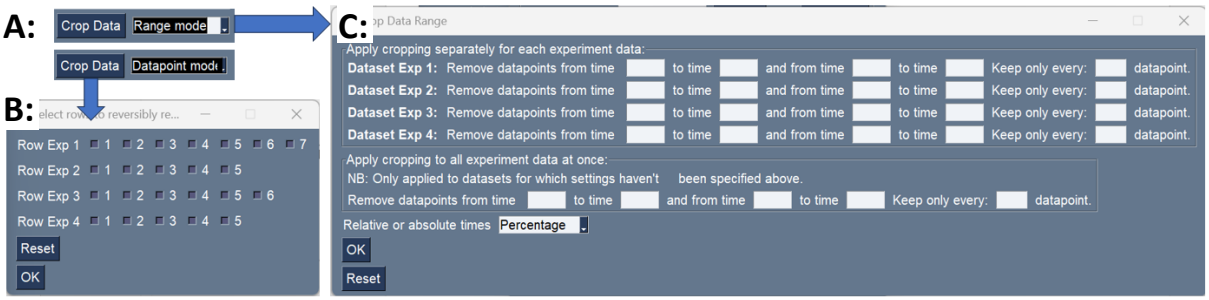


Figure S19: Data cropping menus for “Datapoint mode” (B) and “Range mode” (C) for modifying the uploaded kinetic data. The relevant menu can be generating by clicking “crop” once the correct data cropping mode has been selected.

6. Over- and under-fitting effects on overlay score

6.1. The importance of the monotonicity constraint to avoid over-fitting

The standard settings of Auto-VTNA are 5th degree polynomial fitting and monotonically constrained fitting which are applied to obtain the reaction order value(s) to give the best concentration profile overlay. Utilising ordinary polynomial fitting, without monotonic constraints, overfitting may occur when incorrect order values are applied, rendering the overlay score less reliable. The effect of overfitting on the overlay score also gives rise to rugged overlay score versus order plots with possible local minima (Fig. S20-22). Overfitting occurs when non-overlapping concentration profiles are simultaneously approached by a meandering polynomial with several extrema. For example, overfitting gives rise to the local minima at order values of 0.3 and 2.5 in Figure S20A when an ordinary rather than monotonic fit function is applied.

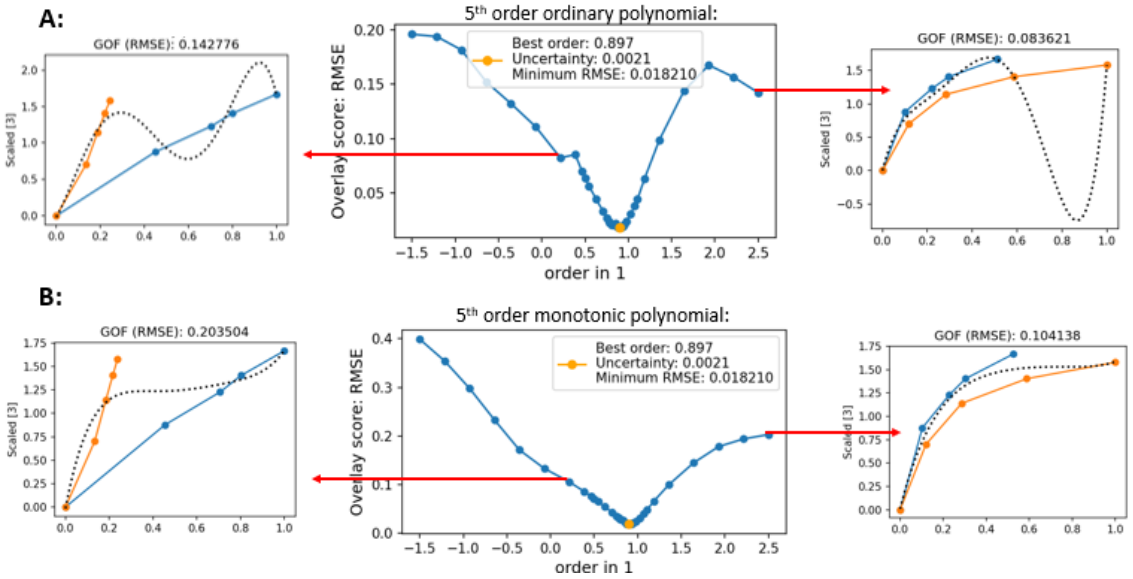


*Figure S20: A: Overlay versus order plot for Experiments 2 and 3 from kinetic data reported by Clark et al.[4] on the Aza-Michael addition of **1** and **2** in DMSO using both an ordinary (A) and a monotonic polynomial fitting method (B). The VTNA overlay plots and fit functions are shown for orders of 2.5 and 0.3 to illustrate the over-fitting effects which give rise to irregularities the overlay score vs. order plot generated using an ordinary polynomial.*

In general, the overfitting effects of ordinary polynomials without monotonic constraints are more pronounced when the data point density of concentration profiles is lower, and the degree of the polynomial is higher. For example, increasing the degree of the polynomial used in Figure S20 to 7 causes Auto-VTNA to fail with an ordinary polynomial fitting function. However, by monotonically constraining the polynomial, a useful overlay score vs. order graph can still be obtained for this high polynomial degree (Fig. S21).

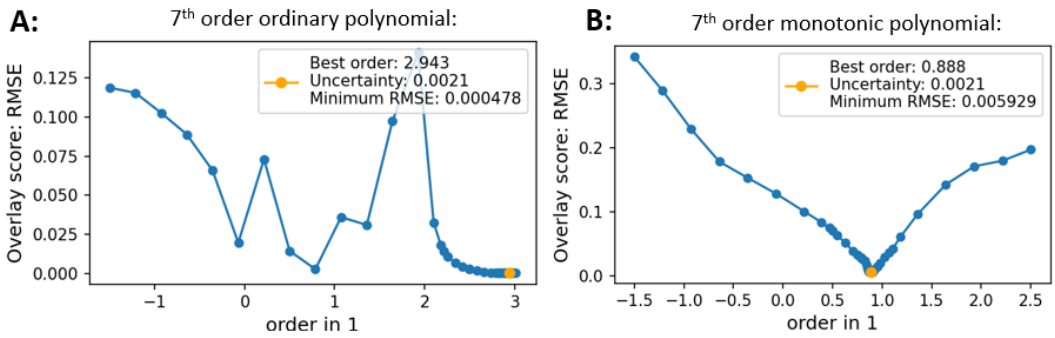


*Figure S21: Overlay versus order plot for Experiment 2 and 3 from kinetic data reported by Clark et al. on the Aza-Michael addition of **1** and **2** in DMSO using both an ordinary (A) and a monotonic 7th degree polynomial fitting method (B).*

This same overfitting effect gives rise to rugged contour plots when the order values for two normalised species are determined using an ordinary 7th degree polynomial fit without monotonic constraints. (Fig. S22).

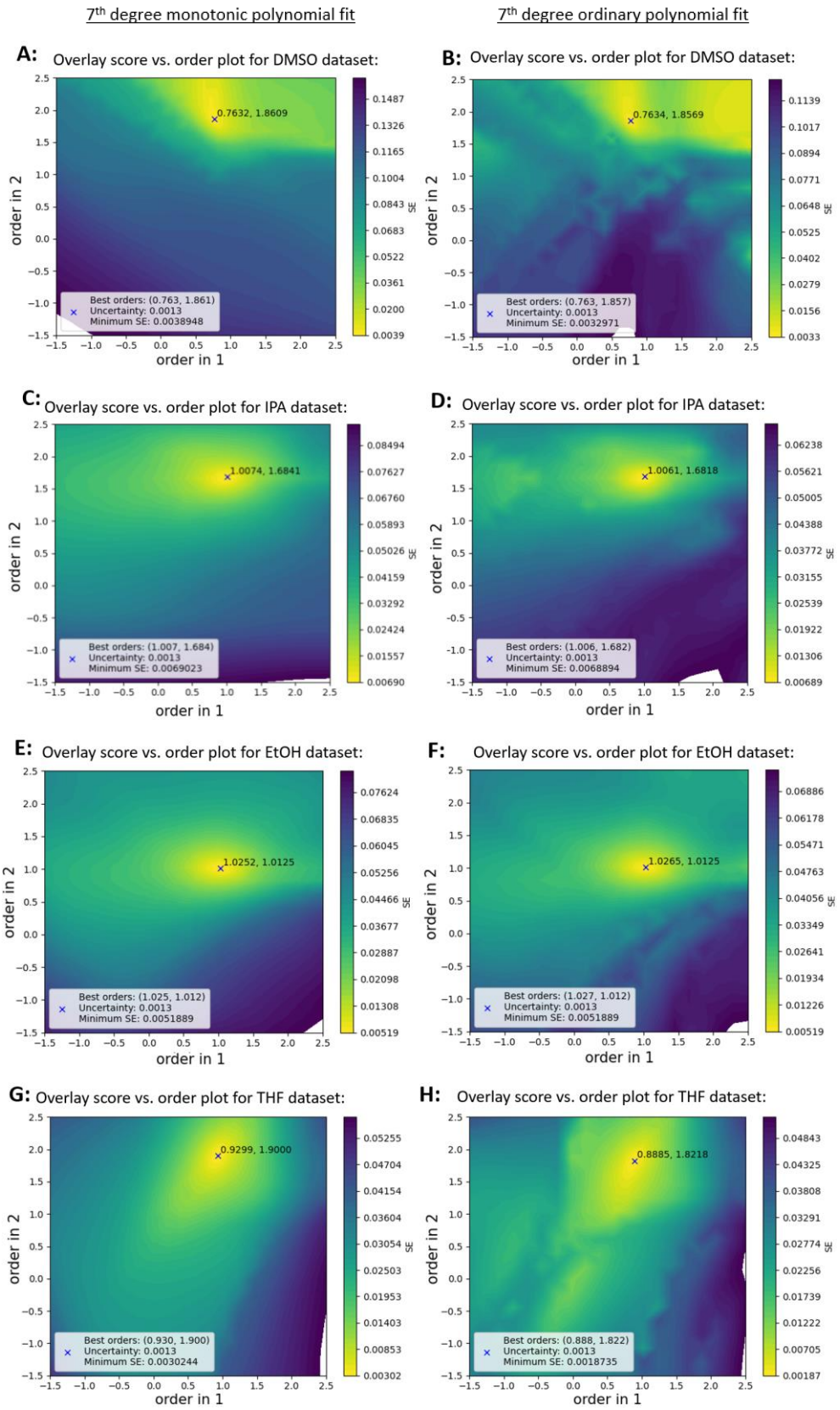


*Figure S22: Contour plots illustrating the degree of concentration profile overlay as the standard error of the total fit against the reaction order in dimethyl itaconate **1** and piperidine **2** for the aza-Michael reaction in DMSO (A and B), IPA (C and D), EtOH (E, F) and THF (G and H). The left and right plots represent automatic VTNA fitted with a monotonic and ordinary 7th degree polynomial, respectively.*

6.2. The use of linear fitting functions

By using a flexible high degree polynomial fitting function, Auto-VTNA identifies the order value(s) that optimise concentration profile overlay upon time axis normalisation. If the time axis has been normalised with respect to every reaction species in the rate equation for which concentration changes over time, overlay occurs concurrently with linearisation. Hence, a linear fitting function can also be utilized to derive reaction order values in such cases.

However, in practice, the order value(s) giving greatest linearisation are unlikely to be exactly the same as the order value(s) giving greatest concentration profile overlay. A good example of this issue is the application of Auto-VTNA to the kinetic data on the aza-Michael addition of piperidine to dimethyl itaconate in different solvents as discussed in the main paper.[4] Applying Auto-VTNA to different excess experiments in both reactants simultaneously with both a linear and 5th degree monotonic polynomial gives rather similar reaction order values for the IPA and EtOH datasets but significantly different order values for the THF dataset and for the order in piperidine from the DMSO dataset (Table S1). Hence, a 5th or 7th degree monotonic polynomial is recommended for the calculation of reaction orders using Auto-VTNA.

Table S1: Order values obtained by applying Auto-VTNA to the aza Michael addition of 1 to 2 in four different solvents.[4]

	Linear fit through origin			**5th degree monotonic polynomial fit**		
Dataset	**Order value dimethyl itaconate**	**Order value piperidine**	**Overlay score**	**Order value dimethyl itaconate**	**Order value piperidine**	**Overlay score**
DMSO	0.761	1.939	0.0404	0.762	1.872	0.0169
THF	1.190	2.186	0.0635	0.933	1.900	0.0153
EtOH	1.114	1.004	0.0373	1.029	1.015	0.0256
IPA	1.003	1.668	0.0509	0.997	1.671	0.0328

By applying a linear fit, through the Automatic VTNA Calculator GUI by changing the fit setting degree to 1 or utilising the python code through the method.plot_VTNA_with_overlay_score(deg=1) to a Normal_VTNA object, a VTNA overlay plot is generated with the slope of the fitted straight line in the table caption. This can be used to derive the observed rate constant of the reaction, although it should be noted that this value is sensitive to the order values selected for time axis normalisation (Fig. S23).

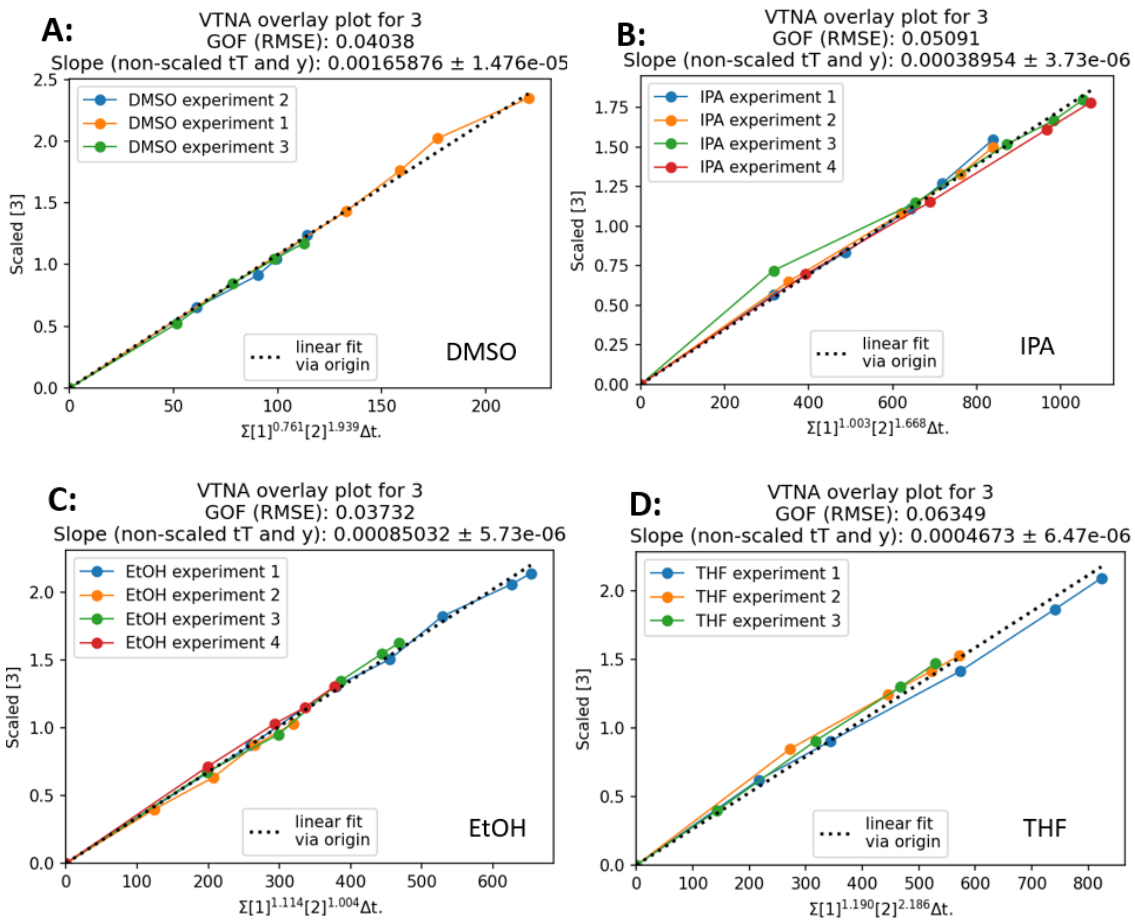


*Figure S23: VTNA overlay plots with reaction orders in dimethyl itaconate **1** and piperidine **2** calculated via automatic VTNA with a linear fitting function for the aza-Michael reaction in DMSO (A), IPA (B), EtOH (C) and THF (C).*[4]

6.3. Avoiding under-fitting

The accuracy and reliability of any automatic VTNA algorithm is bound by the robustness of the method by which concentration profile overlay is assessed computationally. As described previously (see Section S2), the overlay score method underpinning Auto-VTNA is total polynomial fitting, preferably monotonically constrained to limit over-fitting effects for datasets with low data point density. For total polynomial fitting to reliably yield accurate overlay scores, the degree of the polynomial must be sufficiently high to capture the true shape of the time normalised concentration profiles so that the score is minimised when concentration profiles overlay optimally. Otherwise, underfitting occurs, rending overlay scores and the assigned reaction order value(s) erroneous.

To assess the minimum polynomial degree required to avoid underfitting, the optimal order value and corresponding overlay score derived using Auto-VTNA can be plotted against the polynomial degree utilised. For example, the reaction order values in $Pd(OAc)_2$ for concentration time data collected by Newton *et al*. on the Heck reaction between iodobenzene and methyl acrylate, converge when the

polynomial degree reaches 5 (Fig. S24 and S26A).[5] Thus, as illustrated in Figure 5, a 5th degree polynomial fitting function is sufficiently flexible to capture the true shape of the overlaying time normalised concentration profiles (Fig. S25 and S26B).

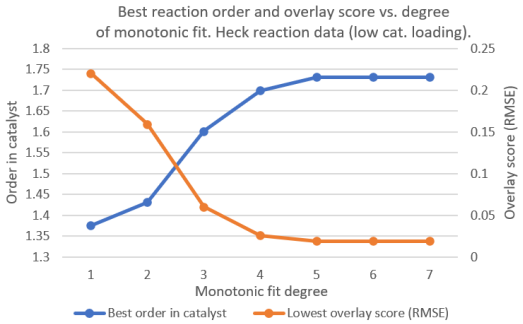


Figure S24: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for concentration profiles for the Heck reaction between iodobenzene and methyl acrylate at the lowest catalyst loadings of $Pd(OAc)_2$.[5]

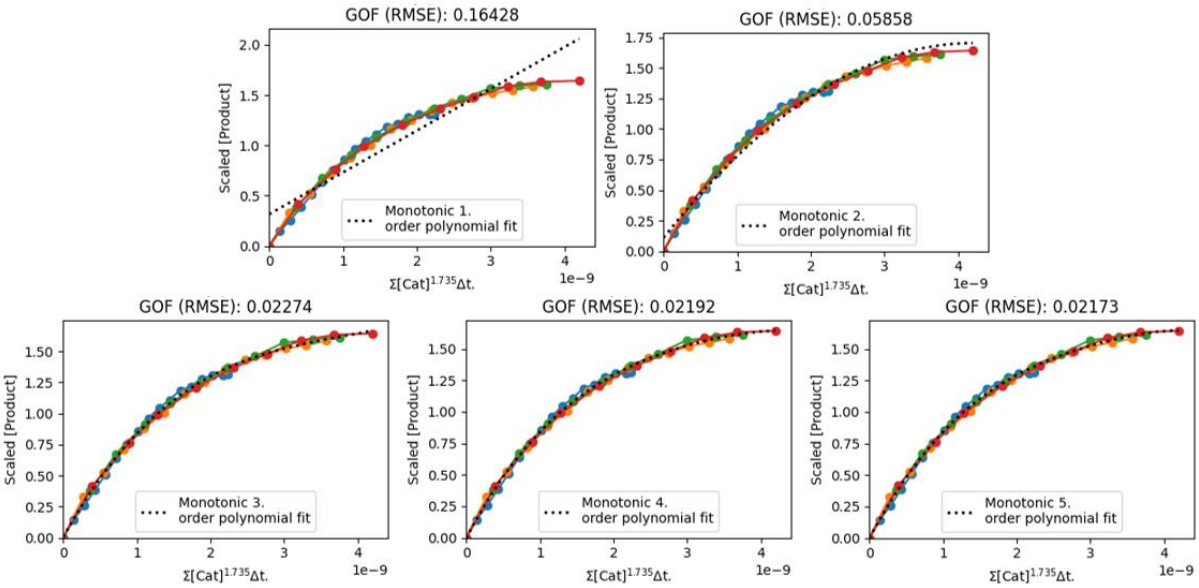


Figure S25: VTNA overlay plots for the best order value in catalyst (1.735) with the total fitting function (polynomial degrees of 1, 2, 3, 4 and 5) shown as a black stapled line.

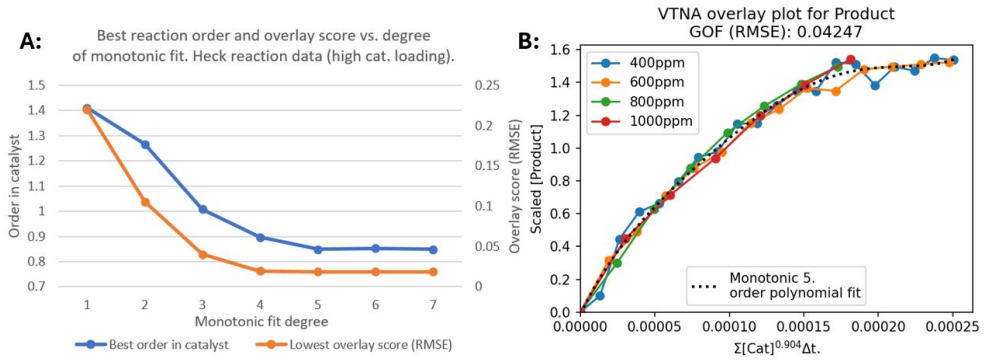


Figure S26: A: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for concentration profiles for the Heck reaction between iodobenzene and methyl acrylate at the highest catalyst loadings of $Pd(OAc)_2$.[5] *B: TNA overlay plot where the time axis has been normalised with respect to the initial concentration of $Pd(OAc)_2$ for the experiments with catalyst loadings at 400 ppm or higher.*[5] *The order value used for time normalisation was obtained via automatic VTNA with a monotonic polynomial degree of 5.*

The same convergence effect can be observed for the reaction order values of the transient imine directing group (TDG) and copper source CuF_2 obtained from VTNA on concentration-time data for a CH sulfonylation reaction developed by Bull *et al.* (Entry 11, Table S2).[6] For the CuF_2 order value calculation, the best overlay score and corresponding order value converge when the polynomial degree is 5 (Fig. S27).[6] However, for the TDG order value calculation, convergence is reached at a polynomial degree of 6 (Fig. S28), indicating that degrees higher than the standard value of 5 may be needed for some datasets to ensure that underfitting does not affect the order obtained.

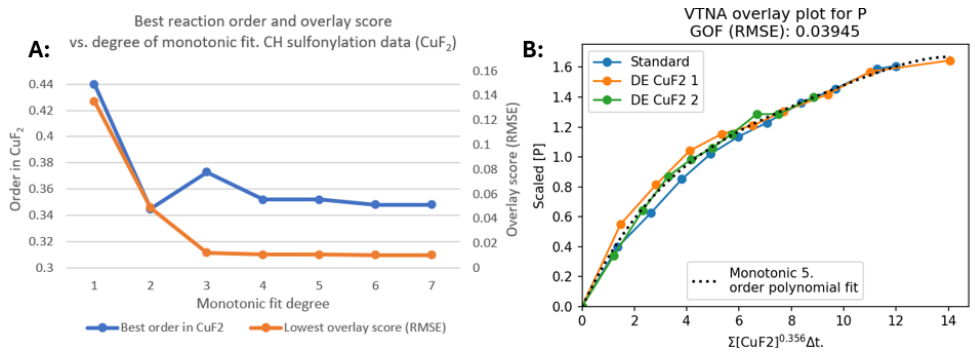


Figure S27: A: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for different excess data in CuF_2. B: TNA overlay plot where the time axis has been normalised with respect to the initial concentration of the CuF_2 copper source for the standard and relevant different excess experiments.[6]

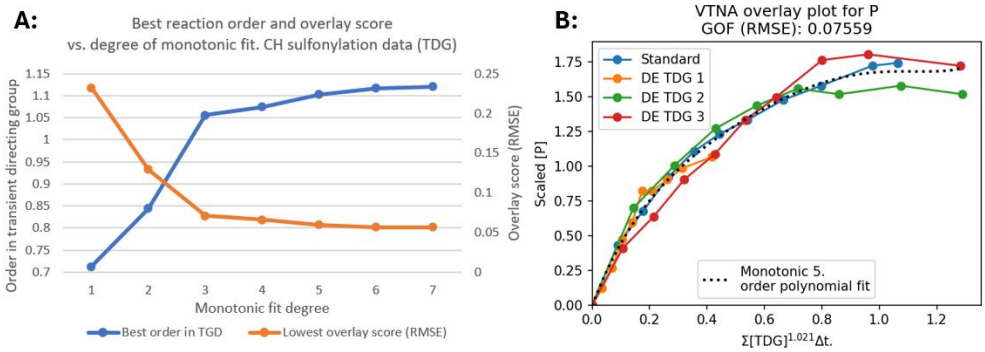
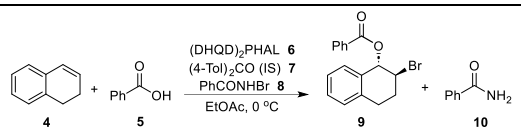
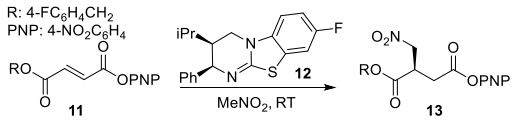
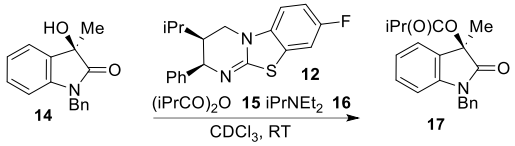
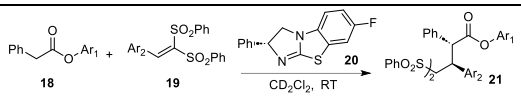
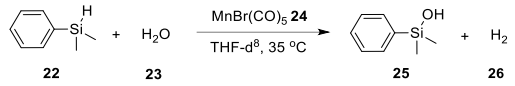
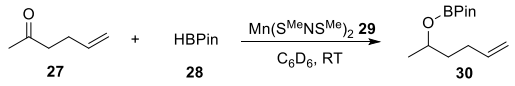
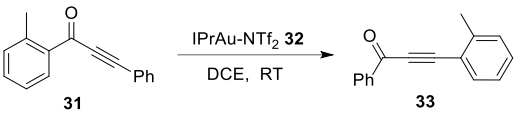
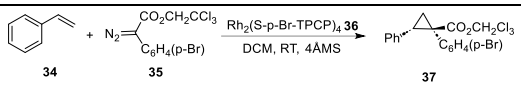
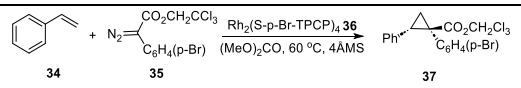
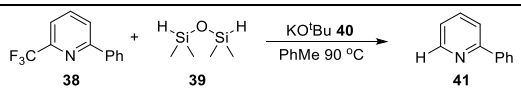
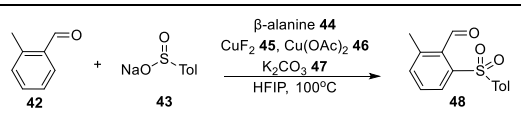
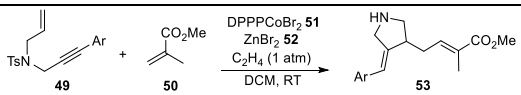
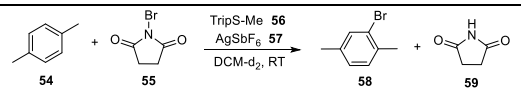
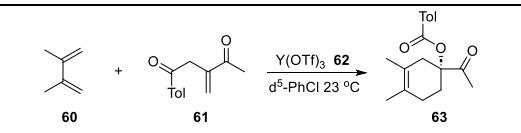


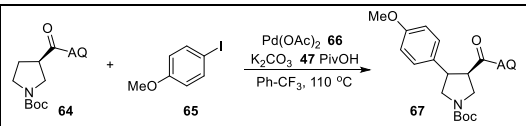
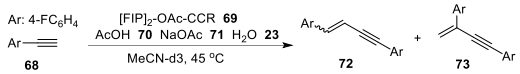
Figure S28: A: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for different excess data in transient directing group (Entry 11 Table S2). B: VTNA overlay plots where the time axis has been normalised with respect to the concentration profiles of the transient imine directing group (TDG) for the standard and relevant different excess experiments.[6]

6.4. Higher degree polynomial fittings to combat under-fitting

In some cases, a 5th degree polynomial could be insufficient to achieve complete convergence of the calculated order value obtained by automatic sequential VTNA, as compared and detailed in Table S2 to 6th and 7th degree fittings. If the order values were found to change by more than 0.025 upon increasing the degree of the fitting polynomial, this was seen as an indication that a 5th degree polynomial was insufficient to derive an accurate order value due to under-fitting. As highlighted in yellow in Table S2, this indeed occurs for some of the order values obtained by automatic sequential VTNA for the datasets in entry 1, 7, 8, 10 and 12.[7–11]

Table S2: VTNA datasets for complex reaction systems from literature with order values calculated by automatic sequential VTNA using a 5th, 6th and 7th degree polynomial (Entries correspond to those in Table 2). The reaction species for which the order value hasn't converged using a 5th order polynomial are highlighted in yellow.

Entry	Reaction system	Orders obtained by sequential VTNA		
		5th polynomial	6th polynomial	7th polynomial
1[11]		$[\mathbf{4}]^{0.996}[\mathbf{5}]^{1.133}$ $[\mathbf{6}]^{0.685}.[\mathbf{7}]^{-0.030}$ $[\mathbf{8}]^{1.044}$	$[\mathbf{4}]^{1.026}[\mathbf{5}]^{1.132}$ $[\mathbf{6}]^{0.756}.[\mathbf{7}]^{-0.001}$ $[\mathbf{8}]^{1.040}$	$[\mathbf{4}]^{1.022}[\mathbf{5}]^{1.127}$ $[\mathbf{6}]^{0.817}.[\mathbf{7}]^{-0.008}$ $[\mathbf{8}]^{1.025}$
2[12]		$[\mathbf{11}]^{0.854}[\mathbf{12}]^{0.957}$ $[\mathbf{13}]^{-0.483}$	$[\mathbf{11}]^{0.849}[\mathbf{12}]^{0.948}$ $[\mathbf{13}]^{-0.483}$	$[\mathbf{11}]^{0.845}[\mathbf{12}]^{0.950}$ $[\mathbf{13}]^{-0.490}$
3[13]		$[\mathbf{14}]^{0.849}[\mathbf{12}]^{1.044}$ $[\mathbf{15}]^{1.071}[\mathbf{16}]^{0.075}$	$[\mathbf{14}]^{0.849}[\mathbf{12}]^{1.036}$ $[\mathbf{15}]^{1.071}[\mathbf{16}]^{0.075}$	$[\mathbf{14}]^{0.849}[\mathbf{12}]^{1.037}$ $[\mathbf{15}]^{1.071}[\mathbf{16}]^{0.073}$
4[14]		$[\mathbf{18}]^{1.060}[\mathbf{19}]^{0.990}$ $[\mathbf{20}]^{1.021}$	$[\mathbf{18}]^{1.060}[\mathbf{19}]^{0.983}$ $[\mathbf{20}]^{1.026}$	$[\mathbf{18}]^{1.060}[\mathbf{19}]^{0.983}$ $[\mathbf{20}]^{1.026}$
5[15]		$[\mathbf{22}]^{0.967}[\mathbf{23}]^{-0.582}$ $[\mathbf{24}]^{0.940}$	$[\mathbf{22}]^{0.963}[\mathbf{23}]^{-0.579}$ $[\mathbf{24}]^{0.940}$	$[\mathbf{22}]^{0.963}[\mathbf{23}]^{-0.579}$ $[\mathbf{24}]^{0.940}$
6[16]		$[\mathbf{27}]^{0.913}[\mathbf{28}]^{0.974}$ $[\mathbf{29}]^{0.902}$	$[\mathbf{27}]^{0.913}[\mathbf{28}]^{0.974}$ $[\mathbf{29}]^{0.895}$	$[\mathbf{27}]^{0.913}[\mathbf{28}]^{0.974}$ $[\mathbf{29}]^{0.895}$
7[7]		$[\mathbf{31}]^{1.593}[\mathbf{32}]^{0.890}$ $[\mathbf{33}]^{0.120}$	$[\mathbf{31}]^{1.594}[\mathbf{32}]^{0.890}$ $[\mathbf{33}]^{0.139}$	$[\mathbf{31}]^{1.594}[\mathbf{32}]^{0.893}$ $[\mathbf{33}]^{0.136}$
8[8]		$[\mathbf{34}]^{-1.066}.[\mathbf{35}]^{1.053}$ $[\mathbf{36}]^{1.139}$	$[\mathbf{34}]^{-1.094}.[\mathbf{35}]^{1.053}$ $[\mathbf{36}]^{1.139}$	$[\mathbf{34}]^{-1.105}.[\mathbf{35}]^{1.053}$ $[\mathbf{36}]^{1.139}$
9[8]		$[\mathbf{34}]^{-0.429}[\mathbf{35}]^{1.598}$ $[\mathbf{36}]^{1.078}$	$[\mathbf{34}]^{-0.440}[\mathbf{35}]^{1.598}$ $[\mathbf{36}]^{1.078}$	$[\mathbf{34}]^{-0.429}[\mathbf{35}]^{1.587}$ $[\mathbf{36}]^{1.082}$
10[9]		$[\mathbf{38}]^{1.044}[\mathbf{39}]^{0.880}$ $[\mathbf{40}]^{0.723}$	$[\mathbf{38}]^{1.044}[\mathbf{39}]^{0.967}$ $[\mathbf{40}]^{0.819}$	$[\mathbf{38}]^{1.048}[\mathbf{39}]^{0.974}$ $[\mathbf{40}]^{0.832}$
11[6]		$[\mathbf{42}]^{1.053}[\mathbf{43}]^{0.771}$ $[\mathbf{44}]^{1.021}[\mathbf{45}]^{0.354}$ $[\mathbf{46}]^{-0.037}[\mathbf{47}]^{0.940}$	$[\mathbf{42}]^{1.033}[\mathbf{43}]^{0.777}$ $[\mathbf{44}]^{1.044}[\mathbf{45}]^{0.361}$ $[\mathbf{46}]^{-0.037}[\mathbf{47}]^{0.940}$	$[\mathbf{42}]^{1.010}[\mathbf{43}]^{0.772}$ $[\mathbf{44}]^{1.037}[\mathbf{45}]^{0.354}$ $[\mathbf{46}]^{0.033}[\mathbf{47}]^{0.936}$
12[6]		$[\mathbf{49}]^{0.139}.[\mathbf{50}]^{0.788}$ $[\mathbf{51}]^{1.239}[\mathbf{52}]^{0.913}$	$[\mathbf{49}]^{0.129}.[\mathbf{50}]^{0.784}$ $[\mathbf{51}]^{1.243}[\mathbf{52}]^{0.951}$	$[\mathbf{49}]^{0.114}.[\mathbf{50}]^{0.777}$ $[\mathbf{51}]^{1.237}[\mathbf{52}]^{0.944}$
13[17]		$[\mathbf{54}]^{0.854}[\mathbf{55}]^{0.125}$ $[\mathbf{56}]^{-0.071}[\mathbf{57}]^{0.947}$	$[\mathbf{54}]^{0.854}[\mathbf{55}]^{0.125}$ $[\mathbf{56}]^{-0.078}[\mathbf{57}]^{0.947}$	$[\mathbf{54}]^{0.854}[\mathbf{55}]^{0.121}$ $[\mathbf{56}]^{-0.078}[\mathbf{57}]^{0.940}$
14[18]		$[\mathbf{60}]^{0.741}[\mathbf{61}]^{0.963}$ $[\mathbf{62}]^{0.956}$	$[\mathbf{60}]^{0.741}[\mathbf{61}]^{0.952}$ $[\mathbf{62}]^{0.956}$	$[\mathbf{60}]^{0.741}[\mathbf{61}]^{0.956}$ $[\mathbf{62}]^{0.967}$

15[19]		$[\mathbf{64}]^{0.217}[\mathbf{65}]^{1.323}$ $[\mathbf{66}]^{0.965}$	$[\mathbf{64}]^{0.227}[\mathbf{65}]^{1.332}$ $[\mathbf{66}]^{1.003}$	$[\mathbf{64}]^{0.143}[\mathbf{65}]^{0.1362}$ $[\mathbf{66}]^{0.973}$
16[20]		$[\mathbf{68}]^{0.969}[\mathbf{69}]^{0.600}$ $[\mathbf{70}]^{0.114}[\mathbf{71}]^{0.176}$ $[\mathbf{23}]^{0.101}$	$[\mathbf{68}]^{0.964}[\mathbf{69}]^{0.598}$ $[\mathbf{70}]^{0.114}[\mathbf{71}]^{0.176}$ $[\mathbf{23}]^{0.101}$	$[\mathbf{68}]^{0.962}[\mathbf{69}]^{0.596}$ $[\mathbf{70}]^{0.114}[\mathbf{71}]^{0.176}$ $[\mathbf{23}]^{0.101}$

These findings were followed by the calculation the optimal order value and the corresponding goodness-of-fit value with monotonic polynomial degrees of 1 to 8, 9 or 10 for reaction species **6**, **33**, **34**, **39**, **40** and **49** which had not converged with a 5th degree monotonic polynomial. This analysis was carried out to establish the lowest polynomial fit degree required to avoid underfitting in the sequential VTNA calculations (Fig S29-S34).

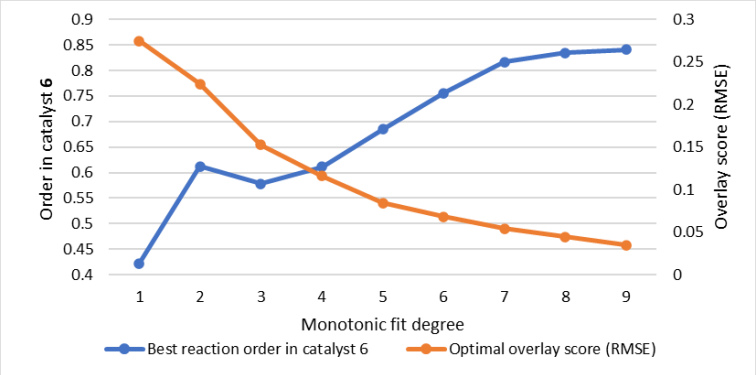


*Figure S29: Convergence of optimal reaction order values and overlay score calculated via automatic sequential VTNA with increasing monotonic fit degree for different excess data in catalyst **6** (entry 1 Table S2).*[11]

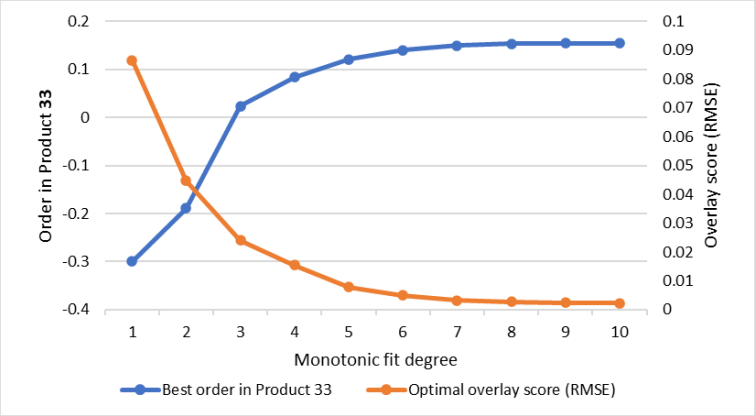


*Figure S30: Convergence of optimal reaction order values and overlay score calculated via automatic sequential VTNA with increasing monotonic fit degree for different excess data in product **33** (entry 7 Table S2).*[7]

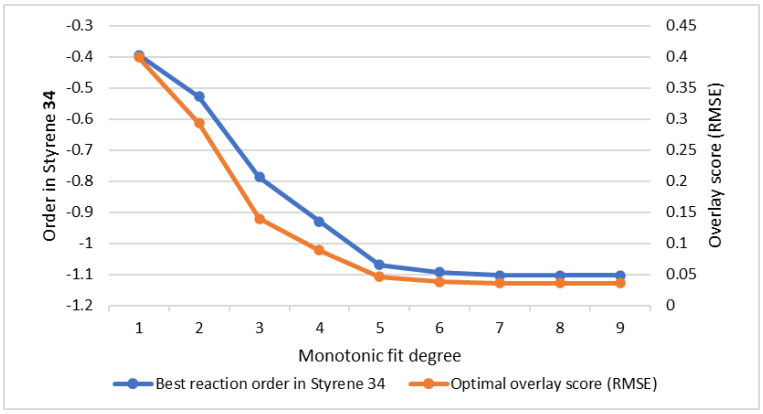


*Figure S31: Convergence of optimal reaction order values and overlay score calculated via automatic sequential VTNA with increasing monotonic fit degree for different excess data in styrene **34** (entry 8 Table S2).*[8]

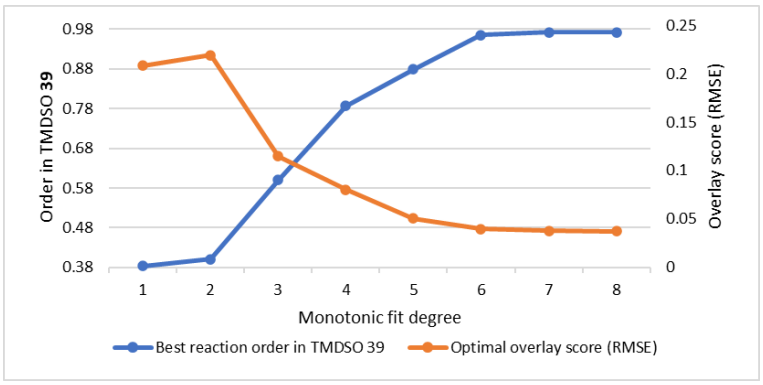


*Figure S32: Convergence of optimal reaction order values and overlay score calculated via automatic sequential VTNA with increasing monotonic fit degree for different excess data in TDMSO **39** (entry 10 Table S2).*[9]

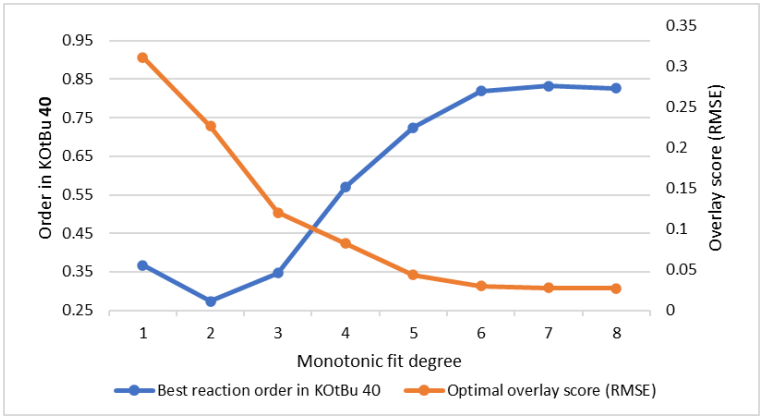


*Figure S33: Convergence of optimal reaction order values and overlay score calculated via automatic sequential VTNA with increasing monotonic fit degree for different excess data in KOtBu **40** (entry 10 Table S2).* [9]

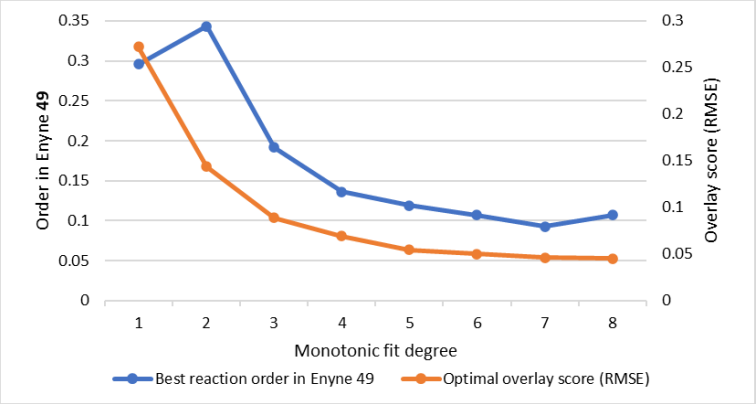


*Figure S34: Convergence of optimal reaction order values and overlay score calculated via automatic sequential VTNA with increasing monotonic fit degree for different excess data in Enyne **49** (entry 12 Table S2).*[10]

As shown in Figures S30-S34, optimal reaction orders in **34**, **39** and **40** and their corresponding overlay scores were found to converge when the degree of the monotonic polynomial fitting function reached 6. The automatic calculation of the optimal order in **33** and **49** required a 7th degree monotonic polynomial fitting function to reach convergence. The calculation of the optimal order in **6** was most challenging, requiring an 8th degree monotonic polynomial to reach convergence, although the 7th degree monotonic polynomial also performed well for determining the reaction order.

Close visual inspection of overlay plots using order values in **33** and **40** calculated with a 5th and 7th degree monotonic polynomial reveals that the latter gives better concentration profile overlay (Figures S35 and S36). This supports the idea that the convergence of calculated order values with polynomial fitting degree reflects the identification of the true order value as the polynomial becomes sufficiently flexible to capture the shape of the concentration profiles at the order values that give best overlay.

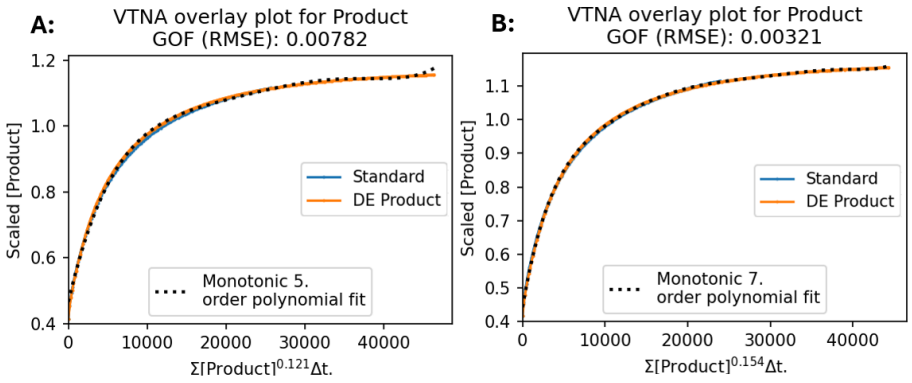


*Figure S35: VTNA overlay plots for the reaction system shown in Entry 7 in Table S2 for the orders in **33** obtained using a 5th (A) and a 7th (B) degree monotonic polynomial fitting function. The fitted 5th and 7th degree monotonic polynomials are shown as black stapled lines.*[7]

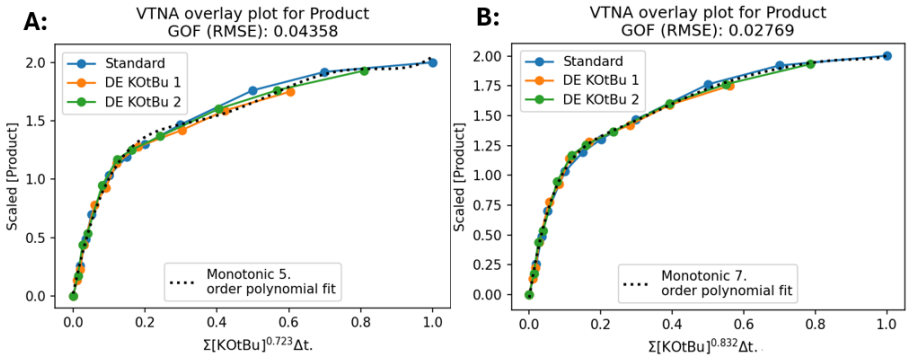


*Figure S36: VTNA overlay plots for the reaction system shown in Entry in Table S2 for the orders in **40** obtained using a 5th (A) and a 7th (B) degree monotonic polynomial fitting function. The fitted 5th and 7th degree monotonic polynomials are shown as black stapled lines.*[7]

As shown in Figures S35 and S36, the 5th degree polynomials fail to replicate the true shape of the overlapping concentration profiles due to oscillatory artifacts known as Runge's phenomenon.[21] Fortunately, this issue appears less pronounced with a 7th degree monotonically constrained polynomial fitting, leading to calculated order values better reflecting the optimal overlay of time normalised concentration profiles as judged by visual inspection.

Further investigations were made to better understand why a 5th degree monotonic polynomial was sufficient in most cases but not for calculating optimal orders for **6**, **33**, **34**, **39**, **40** and **49**. It was found that the greater the curvature and step-like nature of the concentration profiles, the higher the polynomial degree required to avoid under-fitting. For example, the degree required to reach reaction order convergence with polynomial degree for the determination of the order in bromination reagent **55**, alcohol **14** and catalyst **20** were found to be 2, 3 and 5 respectively, following a trend of increasing concentration profile curvature (Fig. S37-S39).

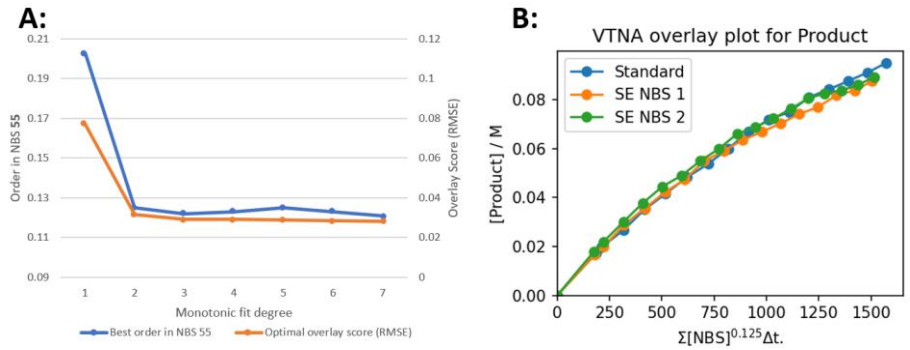


*Figure S37: A: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for the automatic sequential VTNA calculation of the order in NBS **55** (entry 13 Table S2).*[17] *B: the corresponding VTNA overlay plot with the order value calculated using a 5th degree monotonically constrained polynomial fitting method.*

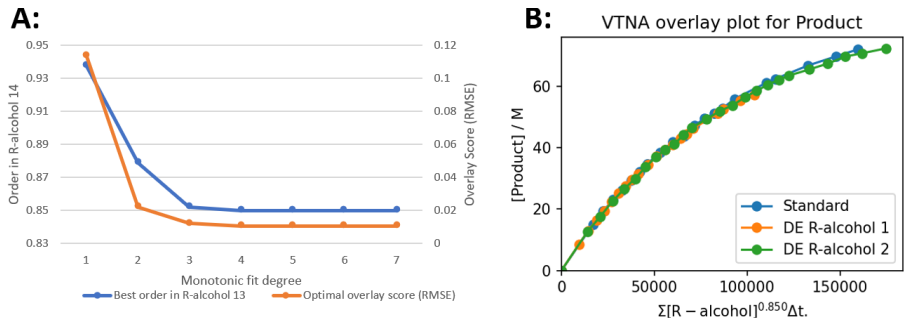


*Figure S38: A: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for the automatic sequential VTNA calculation of the order in R-alcohol **14** (Entry 3 Table S2).*[13] *B: the corresponding VTNA overlay plot generated with the order value calculated using a 5th degree monotonically constrained polynomial fitting method.*

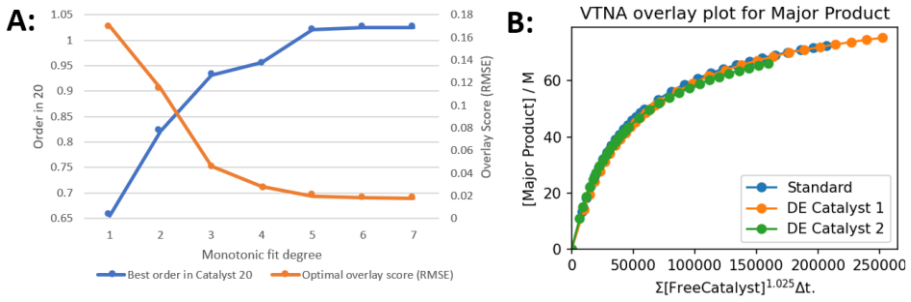


*Figure S39: A: Convergence of optimal reaction order values and overlay score with increasing monotonic fit degree for the automatic sequential VTNA calculation of the order in catalyst **20** (entry 4 Table S2).*[17] *B: the corresponding VTNA overlay plot with the order value calculated using a 5th degree monotonically constrained polynomial fitting method.*

Thus, as the curvature of the concentration profiles increases, a higher degree monotonic polynomial is required to prevent underfitting. The diminished performance of polynomial regression on functions with threshold effects such as steep regions that abruptly flatten is well known.[22,23] Although the issue can be solved by increasing the order of the monotonic polynomial to 6, 7 or 8, it suggests that there might exist a different fitting method that can better adopt the shape of flat as well as steep concentration profiles, such as fitting a sum of exponential functions to concentration profiles. However, this was not explored in this project, as polynomial fitting was deemed sufficient for almost all cases.

High concentration profile curvature is observed for reactions with high overall orders as is the case for datasets in entries 1, 10 and 12 in Table S2. Whereas time axis normalisation of a reactant, or other reaction species whose concentration falls during the reaction, decreases the curvature of the reaction profiles when a positive order is applied, using a negative order has the opposite effect. This difference in normalised concentration profile overlay explains why 3rd and 6th degree monotonic polynomials were required to obtain converged order values in **35** and **34** respectively (Fig. S40).

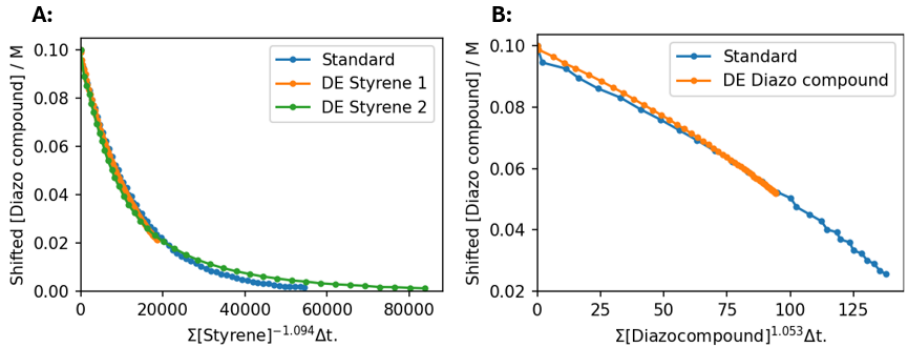


*Figure S40: VTNA overlay plots at the optimal order value for Styrene **34** (A) and Diazo Compound **35** (B) from the kinetic dataset in Entry 8 Table S2. Whereas the highly curved profiles in A required a 6th degree monotonic polynomial to avoid underfitting to achieve order value convergence, the profiles in B required merely a 3rd order polynomial to achieve order value convergence as it is less steep and therefore more easily amenable to low degree polynomial fitting.*

6.5. The effect of transformed time domains to which only one profile reaches

As shown in Figure S1, normalisation of the time axis with respect to the concentration profiles of reactants or other reaction species leads to dispersion of the x-axis domains of the reaction profiles. This means that one reaction profile generally extends beyond the x-axis domains of the other profiles, as seen for the "DE Styrene 2" and "Standard" reaction profiles in Figure S40A. Although this effect can generally does not affect the Auto-VTNA calculation, it can give an artificially low overlay score as the fitting function can generally fit the single profile well, lowering the RMSE of the fit. For example, the overlay score attained with optimal reaction orders derived by automatic total VTNA on the Au(I)-catalysed transposition of ynone **31** (Entry 7, Table S2) was evaluated at a very low 0.00495 but increased to 0.00546 when the "overhanging" "DE ynone" profile was cropped (Fig. S41).[7]

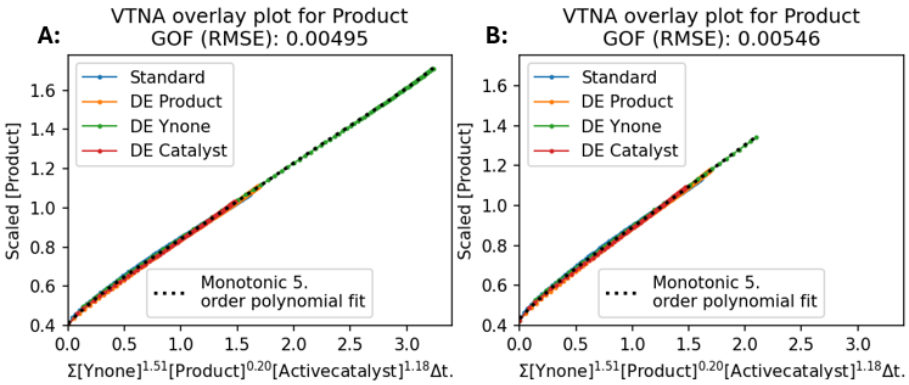


Figure S41: The total VTNA overlay plot for the kinetic dataset in Entry 7 in Table S2 with (B) and without (A) cropping of the "DE ynone" reaction profile to render the overlay score more representative of the true degree of concentration profile overlay.

Of the datasets analysed, the only instance where excessively "overhanging" concentration profiles led to issues, other than a slight lowering of the absolute overlay score value, was for the calculation or reaction orders for the kinetic dataset in Entry 1 of Table S2. For these datasets, the overhanging concentration profiles were flat and therefore fitted poorly by even a 7th degree polynomial. By

truncating the longest reaction profile to the time domain of the second longest profile, the polynomial degree required for order value convergence was reduced from 7 or 8 to 5 for the automatic sequential VTNA calculations for **4**, **5** and **8**. This demonstrates that cropping datasets to only the kinetically useful region can be a valuable approach. However, in some cases such as the automatic total VTNA calculation of the reaction orders in **1** and **2** for the Aza Michael reaction in Scheme 1 in DMSO, overhanging concentration profiles contribute to the elucidation of the correct order value (see Fig. SA38A).

7. Algorithm efficiency analysis

Sequential VTNA calculations in which the order of only one reaction species is calculated are one-dimensional optimisation problems and therefore do not require many time axis normalisation and total fitting events, giving short processing times. However, increasing the number of reaction species for which the order value is determined per iteration, increases the dimensionality of the optimisation problem, leading to exponentially longer processing times.

Big O notation can be used to assess algorithmic efficiency. [24] The Big O notation of the algorithm is shown in Eq. S7 where I is the number of iterations (each representing its own list of order value combinations), R is the resolution of the order value mesh for each iteration and n is the number of normalised species. For example, if 10 order values are trialled per species in 5 iterations (giving 6 rounds of different order meshes), the Big O notation for the algorithm is $O(6 \times 10^n)$.[24] This "bad" Big O notation means that the processing time will increase 10-fold each time the number of normalised species is increased by one. To improve the processing time of Auto-VTNA, several measures were taken.

$$O((I+1) \times R^n) \qquad (Eq.S7)$$

7.1. Measure 1: iterative convergence toward optimal order value(s)

To enable precise determination of the optimal order value for a given reaction species, the Auto-VTNA algorithm has been designed to converge towards the optimal order value(s) over successive iterations (see functions VTNA_orders(), explore_orders() and VTNA_tT() in VTNA_functions.py). First, the overlay score of a selected number of order values or order value combinations is evaluated, before a new mesh of order values is defined around the order(s) that gave the best overlay score in the first iteration (Fig. 2). This can be repeated for a selected number of iterations *I* to improve the precision of the calculated order value while minimising the resolution parameter R (Fig. S42). This is important, because increasing *R* increases the number of order value combinations trialled in a calculation

exponentially with increasing numbers of normalised species n, while increasing I increases the number of order value combinations linearly and independently of n (Eq. S7).

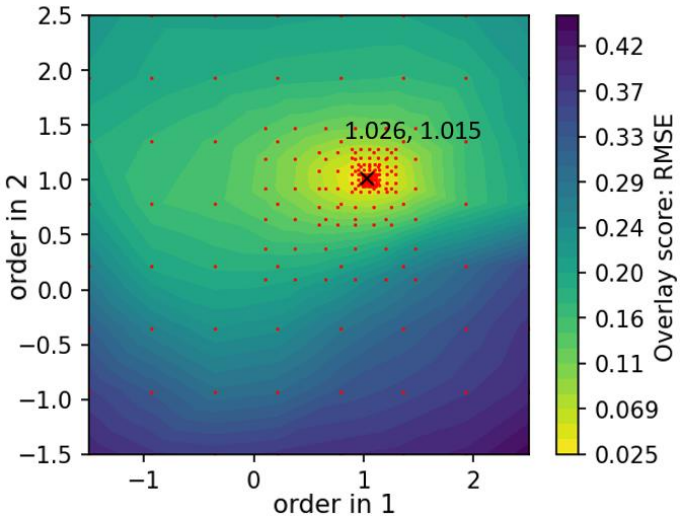


*Figure S42: Contour plot illustrating the degree of concentration profile overlay as the RMSE of the total fit against the reaction order in dimethyl itaconate **1** and piperidine **2** (Scheme 1). The red points illustrate the order combinations for which the overlay score has been calculated.*

To illustrate the importance of minimising R and rather use several iterations to achieve sufficient order value precision (dense mesh of order values around the optimum point) while minimising the number of order value combinations trialled to identify the order values optimising the overlay score, the results of two calculations have been shown in Fig. S43A and B. Both calculations achieved the order value spacing at the optimum point of 0.022, but in B this was done with only one dense order value mesh ($R = 120$ and $I = 0$) giving a processing time of 560 s, whereas in calculation A, the resolution R and the iterations I were both set to 5, giving a processing time of 3.8 s as fewer order combinations were necessary (Fig. S43). This excess in processing time would be even greater if the order values of 3 or more normalised species n had been determined in the calculation as the number of order values combinations scales exponentially with R^n.

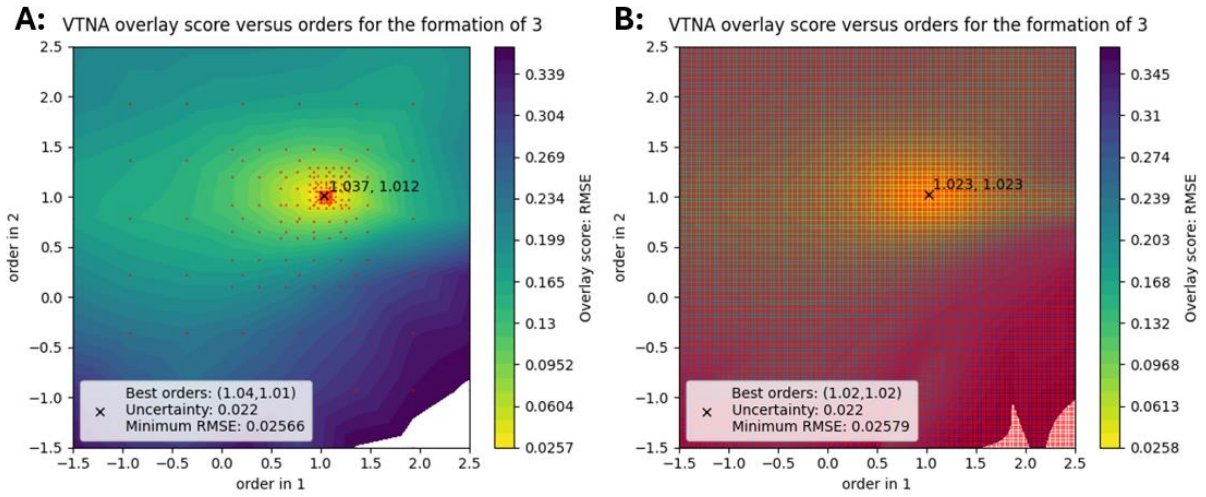


Figure S43: Contour plots with red points illustrating the order value combinations for which an overlay score has been calculated. A: resolution=5 and interations=5. Average processing time: 3.8 s. B: resolution=120 and iterations=0. Average processing time: 601 s.

Although the processing time can be minimised by reducing R, the requirement for the Auto-VTNA calculation to yield precise and accurate order values places a lower bound on R. Firstly, as mentioned above, the requirement for precise order values requires a dense mesh of order values around the optimum point. This can be achieved while minimising R by increasing the number of iterations the algorithm performs to hone in on the best order combinations. However, to ensure that the algorithm doesn't "miss" the optimal combination of order values by honing in on the wrong part of the n-dimensional order value surface, there is a limit to how narrow one order mesh can be relative to the former.

In the VTNA_orders() code in Auto_VTNA, the range of the subsequent order mesh is defined as $[BO - 1.4\Delta O_{-1}, BO + 1.4\Delta O_{-1}]$ where BO is the order value with the best overlay score found in the preceding iteration and ΔO_{-1} is the difference between neighbouring order values in the corresponding order value mesh. This ensures that the correct order value is not overlooked if the overlay score surface is rugged. Thus, the order mesh range of one iteration relative to the former depends on the resolution of each mesh. If $R = 10$, the next order value mesh range will be $2.8 \div 10 = 0.28$ of the mesh range of the previous iteration. However, if the resolution is minimised to 4, the subsequent order mesh would be $2.8 \div 4 = 0.70$ of the mesh range of the previous iteration, thus only 30% closer to the optimal order value combination. Therefore, to speed up the convergence towards the optimal order value combination, a new order value mesh is defined as $[BO - 1.1\Delta O_{-1}, BO + 1.1\Delta O_{-1}]$ if $R = 4$. Similarly, if $R = 5$, a new order value mesh is rather defined as $[BO - 1.2\Delta O_{-1}, BO + 1.2\Delta O_{-1}]$. This means that when $R = 4$, each order value mesh range is $2.2 \div 4 = 0.55$ of the former, and when $R = 5$, each order value mesh range is $2.4 \div 5 = 0.48$ of the former rather than 0.70 and 0.56 respectively.

In practice, the resolution of the order value mesh per iteration is set to alternate between R and $R + 1$. This was found to limit the number of very similar datapoints trialled during an automatic VTNA calculation. For example, if R is set to an even number, order values in the middle of the order value range would be overlooked. The algorithm is also by default set to increase the resolution of the initial order value mesh by a factor of 1.5, but this can be turned off by setting the argument "initial_mesh_denser" to False. All these parameters can be adjusted in the Auto-VTNA code by users with coding experience if desired.

The optimally "quick" settings for an automatic VTNA calculation have been defined as $resolution = 4$, $iterations = 10$ and $constraint = None$. Additionally, the initial order mesh can be set to be no denser than the preceding ones. These parameters can be accessed quickly in the calculation settings of the Automatic VTNA Calculator GUI by clicking the "Quick settings" button and give the modest processing times in Table S3. Pleasingly, the "quick settings" without monotonic fitting and low density order value meshes still afforded the order values optimising concentration profile overlay at every attempt.

Table S3: Average processing time for the automatic total VTNA calculations used to derived the order values in the "total VTNA" column in Table 2. The reported values are the average processing times of 3 identical calculation. The calculations were run from the Automatic VTNA Calculator using the "quick" settings (see above).

Entry	Average processing time for automatic total VTNA calculation / s
1 (all)	1412.1
1	242.3
2	14.3
3	207.9
4	27.9
5	21.6
6	4.3
7	65.0
8	22.8
9	30.5
10	4.1
11	991.6
12	101.5
13	39.4
14	6.1
15	3.5
16	237.9

7.2. Measure 2: improving the efficiency of overlay score calculations

Another approach taken to improve the time efficiency of the automatic VTNA algorithm is to maximise the computational efficiency of the overlay score calculation per order value combination.

The overlay score calculation can be divided into two parts: calculation of the transformed time values of each datapoint and the subsequent total fitting of all datapoints. To optimise the calculation of the

normalised time axis, vectorisation was applied using numpy functions such as np.power(), np.cumsum(), np.prod() and np.apply_along_axis(). This optimization resulted in a 200-fold improvement in the algorithm's time efficiency compared to nested loops, leveraging the inherent speed and efficiency of NumPy's implementation in C and Fortran. The automatic VTNA algorithm also limits redundant calculations. For example, the average of each neighbouring concentration values in each profile is evaluated initially in VTNA_orders() and subsequently raised to the selected order value in VTNA_tT() to limit the operations performed in VTNA_tT() as these are performed once per order value combination. However, a further incremental improvement in processing times may be achieved by rewriting the explore_orders() and VTNA_tT() functions to never raise a sub-array to the same order value more than once.

The processing efficiency of fitting all normalised concentration profiles to the same polynomial depends on the number of datapoints and whether the fit is monotonically constrained (as implemented in the Python package "polyfit"[3]). As shown in Table S4, monotonically constrained polynomial fitting is about 15 times slower than its ordinary counterpart using numpy ("np.polyfit()"). Hence, it is recommended to use an **ordinary polynomial for automatic total VTNA calculations with 4 normalised species or higher** to minimise processing time. The large number of datapoints fitted in automatic total VTNA calculations also reduce the risk of overfitting, which has only been seen for automatic VTNA calculations with few concentration profiles with low datapoint density.

Table S4: Average processing time with standard deviation for 7 repeats of automatic VTNA with an increasing number of experimental data and normalised reaction species used. Calculations were carried out using data from Bull et al. [25] The processing time of these calculations could have been further lowered by applying the "quick settings" detailed above.

Number of normalised species:	Processing time numpy polyfit		Processing time monotonic polyfit:	
	Average / s	STD / s	Average / s	STD / s
1	0.036	0.0018	0.598	0.0559
2	0.262	0.016	4.3	0.391
3	1.84	0.139	31.4	0.676
4	27.9	2.26	232	10.05
5	307	7.86	3794	1233
6	2608	28.7	N/A	N/A

For monotonic fitting, the processing time was found to increase gradually with increasing degrees up to 7. However, beyond this point, there is a stark shift, with the processing time spiking nearly tenfold slower for an 8th degree polynomial. Further escalation to 9th-degree monotonic polynomials leads to a substantial rise in processing time, peaking at 869 seconds—almost a hundredfold increase

compared to lower degrees (Table S5). Therefore, it's advisable to steer clear of monotonic polynomials exceeding the 7th degree, as this level proves more than adequate for most calculations.

Table S5: Processing time against the degree of the monotonic polynomial used to identify the best reaction order value in catalyst for the simulated dataset in

Degree	Average processing time / s	Standard deviation / s
1	9.75	0.785
2	9.63	0.227
3	10.3	0.594
4	10.1	0.968
5	10.4	0.717
6	14.2	0.626
7	15.1	1.29
8	104	3.33
9	869	375

8. The significance of calculated order values.

8.1. Rounding the calculated order values based on the order value precision.

The order values calculated using Auto-VTNA are presented either as “floating-point numbers” with up to 15 decimal figures or rounded according to the resolution of the order value grid density around the optimum point in the overlay score *vs.* order value surface. For example, when resolution and iterations parameters are set to 3 and 5 respectively, the order value mesh around the optimum order values is 0.16, which causes the Automatic VTNA Calculator to round the calculated order values to one decimal place as distinguishing between further significant figures would require a denser order value grid around the optimum (Fig. S44A). Increasing the number of Auto-VTNA iterations to 8 gives an “uncertainty” of 0.01 (Fig. S44B), which means that the order value mesh around the optimum has a spacing of 0.01. This greater precision of the order value mesh allows the calculated order value to be shown with 3 significant figures. Next, if the iterations parameter is further increased to 12, an order value spacing around the optimum of 0.0012 is achieved, leading to 4 significant figures in the calculated reaction order values (Fig. S44C).

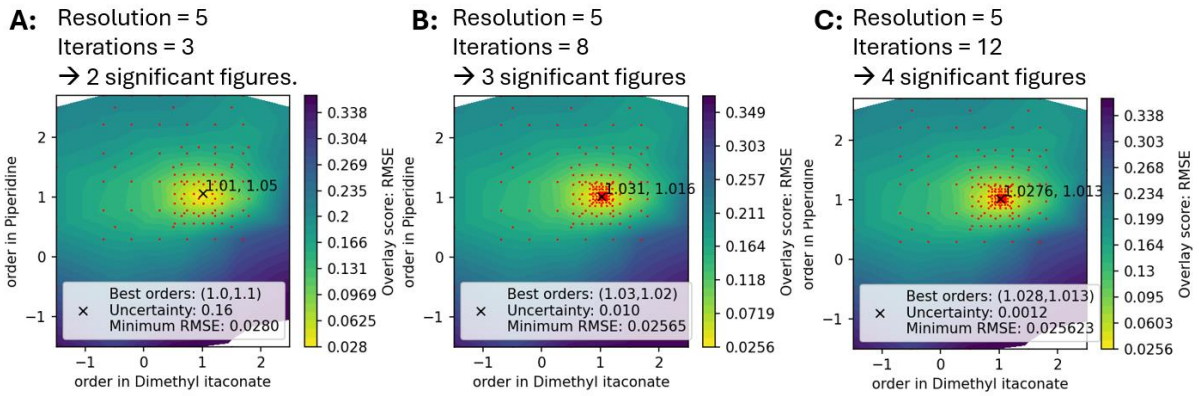


Figure S44: Overlay score versus order value contour plots illustrating the automatic VTNA calculation to determine the optimal reaction orders of the aza Michael reaction shown in Scheme 1.[4] *A): Iterations=3 leads to low order value density around the optimum, causing the reported order values to be rounded to 1 decimal place. B): Iterations=5 leads to an order value density around the optimum of 0.010, causing the reported order values to be rounded to 2 decimal places. C): Iterations=12 leads an order value density around the optimum of 0.0012, causing the reported order values to be rounded to 3 decimal places.*

8.2. The use of overlay score intervals to assess order value uncertainty.

The order value(s) outputted from an automatic VTNA calculation are those that gave the best overlay score out of all the order values or order value combinations trialled during the calculation. Thus, given that the overlay score is reliable, the calculated order value should primarily be considered as the value that maximises concentration profile overlay, without any human bias towards rounded values. However, there will always exist similar order values or order value combinations with overlay scores practically indistinguishable from the optimum.

For each reaction species, the range of order values a given fraction away from the optimum point (0.15 or 15% by default), is referred to as the overlay score intervals. The width of these intervals reflects the sharpness of the optimum point in each of its dimensions, which depends on the conclusiveness of the kinetic data when subjected to VTNA analysis. For example, *in silico* kinetic data will produce concentration profile overlay at the order values corresponding to the mechanism that was used to generate the data and exhibit very narrow overlay score intervals. In fact, the simulated kinetic data presented in Burés's original VTNA publication (A+B→P),[26] gives 15% overlay score intervals of (1.002,1.009) in A and (1.008,1.019) B when orders in both are calculated by automatic VTNA (Fig. S45A). In contrast, the 15% overlay score intervals for the calculation of optimal reaction orders in dimethyl itaconate **1** and piperidine **2** from kinetic data on the aza-Michael reaction in EtOH are considerably wider at (0.945,1.111) and (0.965,1.058) respectively. The more well-defined nature of the optimum point of the simulated kinetic data compared to the real kinetic data can also be appreciated by comparing their 100% and 500% overlay score intervals of (Fig. S45 and Table S6).

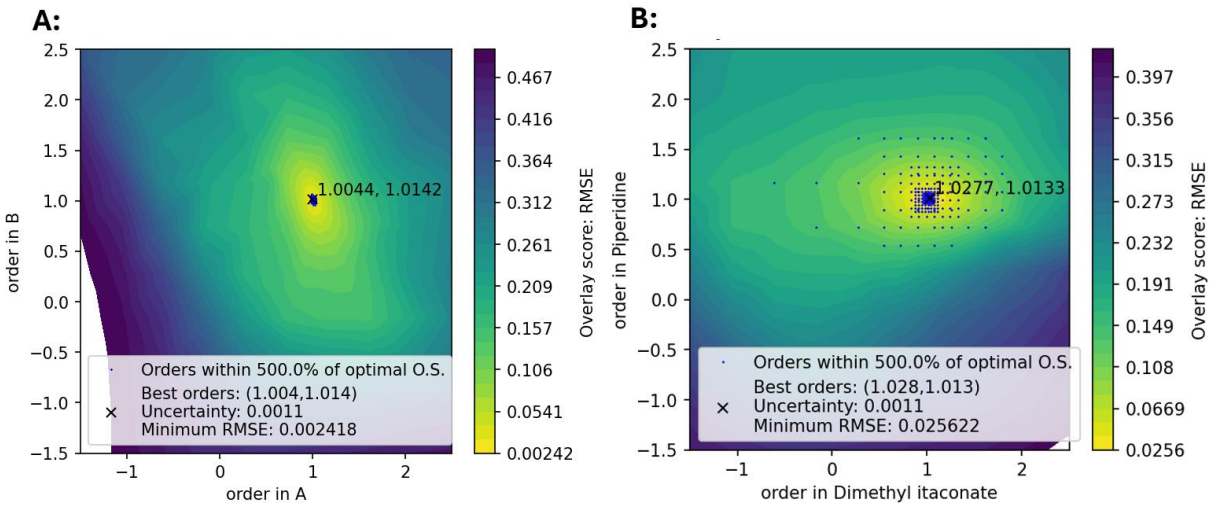


Figure S45: Contour plot illustrating the degree of concentration profile overlay across different reaction order values in A and B for simulated kinetic data (A+B → P)[26] *and in **1** and **2** for data on their aza-Michael addition in EtOH (B).*[4]

Table S6: Intervals of order values in reactants A and B for simulated kinetic data[26] *and real kinetic data on the aza-Michael addition of **2** to **1** in EtOH*[4] *which are represented in order value combinations that give overlay scores a given percentage above the overlay score (RMSE) of the optimal order values.*

Dataset	Simulated data.		Real kinetic data.	
Normalised species	**A**	**B**	**Dimethyl itaconate (1)**	**Piperidine (2)**
15% Lower order limit	1.002	1.008	0.94504	0.96495
15% Upper order limit	1.009	1.019	1.11096	1.05787
100% Lower order limit	0.996	1.002	0.82889	0.87867
100% Upper order limit	1.015	1.027	1.25556	1.16074
500% Lower order limit	0.978	0.959	-0.61111	0.54444
500% Upper order limit	1.033	1.072	1.78889	1.61111

Thus, the reliability and accuracy of calculated order values depends on how well defined the overlay score optimum point is, which can be assessed by visualising the correlation between overlay scores and order values (for 1 or 2 normalised reaction species), and/or by identifying the range of order values that give overlay scores within a given fraction of the optimum. The latter approach is a valuable way to study the uncertainty of calculated order values for 3 or more different normalised reaction species since the results from this calculation cannot be easily visualised. Any score interval value can be selected and used to generate the interval of order values that fall within the selected range from the optimum by calling the method identify_error_intervals() on the automatic VTNA calculation object. Using the GUI, the default score interval value of 0.15, or 15%, can be altered easily by the user to update the interval table that can be seen if the "Show + info" button has been clicked (Fig. S46).

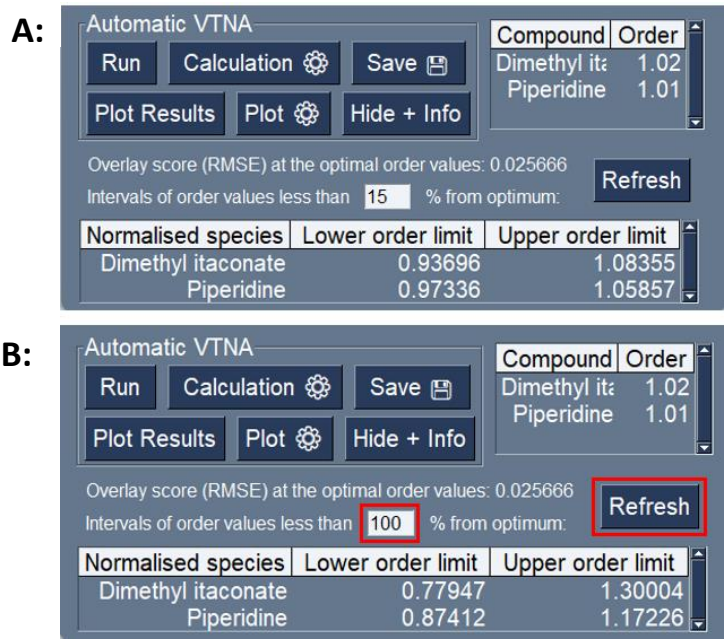


Figure S46: Overlay score interval tables generated by the Automatic VTNA Calculator which can be used to assess the uncertainty in the calculated order values. The overlay score interval is defined as a percentage of the optimal overlay score above (SE or RMSE) or below (R2) the optimal overlay score itself. A: overlay score intervals 15% from the optimum overlay score. B: overlay score intervals 100% from the optimum overlay score.

It should be noted that the overlay score intervals are based on the order value combinations that were trialled during the automatic VTNA calculation and therefore give a rough estimate of the "true" interval which could be obtained if every possible order value combination had been trialled. Thus, the intervals should be read as the lowest and highest order value for that reaction species present in the combinations of order values which gave overlay score within the selected fraction (0.15 or 15% by default) from the best overlay score identified.

For real kinetic data in the literature, a continuum of data quality and signal-to-noise ratio exists, depending on the power and use of analytical instrument, the degree of temperature control and the presence of hard-to-control factors such as catalyst deactivation and impurity concentrations. The intervals of order values which give overlay scores within 15%, or another set fraction of the optimal overlay score, gives a good indication of how conclusively the calculated order value represents the optimal concentration profile overlay. For example, the kinetic data represented by entries 1-8 and 14 in Table S2 exhibit high signal-to-noise ratio, as evidenced by their low optimal overlay scores and calculated order values close to the expected round values and give narrow 15% overlay score intervals less than 0.2 wide (Fig. S47). In particular, the kinetic dataset on the gold-catalysed transposition of ynones in Entry 7, [7] yields remarkably low 15% overlay score intervals less than 0.04 wide, comparable to the simulated data discussed above.

For kinetic data with more noisy reaction progress profiles, the 15% overlay scores can be as wide as a whole order value (Entries 9, 12 and 16), suggesting larger uncertainty in the order values calculated by automatic total VTNA. For example, the order value in **64** in Entry 15 of Table S2 should be taken with a grain of salt since order values between 0 and 0.5 have almost the same overlay score (Fig. SA34A). In such cases, use of Auto-VTNA would suggest that more reaction data should be collected to conclusively state order values.

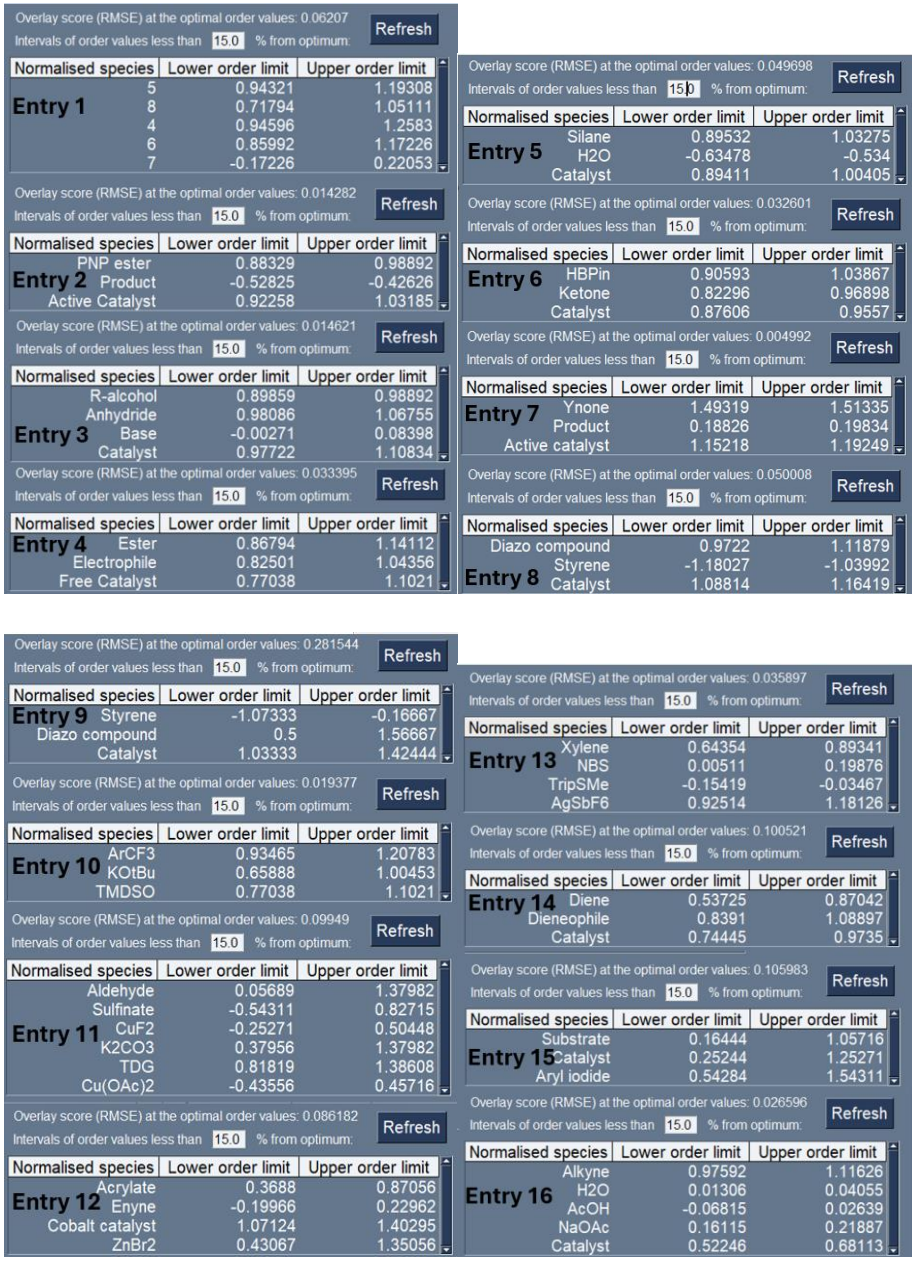


Figure S47: Overlay score interval tables generated by the Automatic VTNA Calculator which can be used to assess the uncertainty in the order values for the kinetic datasets in Entries 1-16 in Figure 2 and S2 calculated via automatic total VTNA.

8.3. Strategies for the collection and preparation of kinetic datasets.

Another source of errors in the calculated order values can be predicting the concentration profiles of reactants from that of the product, effectively assuming perfect mass balance. This does not account for reactants decomposing or converting to unknown by-products and might lead to erroneous time axis normalization and order value calculation. Similarly, assuming that the concentration of additives or catalysts do not change over time due to inhibition or catalyst decomposition can be a source of error. An overview of these assumptions in the kinetic data utilised in this research is shown in Table S7.

Table S7: Overview of how the literature kinetic datasets for the reaction systems in Entries 1-16 in Table S2 were sourced and processed, and how the authors collected the kinetic data experimentally.

Entry	Sourcing of kinetic data	Assumptions made by authors while preparing kinetic data
1	Kinetic data used to calculate the reaction orders in **4**, **5, 6, 7, 8** and **10** was provided by way of an Excel spreadsheet by the authors.	The concentration profiles of the product, measured using HPLC, was used to derive that of the reactants **4**, **5** and **8** as well as by-product **10**. The catalyst and internal standard concentrations were assumed to be constant during the reactions.[11]
2	The kinetic dataset used to calculate the reaction orders in **11**, **12** and **13** was retrieved from page S62-S65 in the ESI of the original publication.[12]	The concentration profiles of reactant **11** and product **13** were measured by quantitative $^{19}F\{H\}$ NMR over time. Both the starting material and the product were ^{19}F-labeled. The concentration of active non-protonated catalyst **12** was measured from its integral and its shift in chemical shift during the reaction.[12]
3	The kinetic dataset used to calculate the reaction orders in **14**, **12**, **15** and **16** was retrieved from page S117-S122 in the ESI of the original publication.[13]	The concentrations of reactant **14** and product **17** were measured over time using quantitative in-situ HNMR. The concentration profiles of the catalyst **12** and the base **16** were assumed to be constant, but the degree of protonation accounted for by measuring their shift in chemical shift over the course of the reaction. The concentration of anhydride **15** could not be measured directly and was calculated from its initial concentration and the concentration profile of **17**.[13]
4	The kinetic dataset used to calculate the reaction orders in **18**, **19** and **20** was retrieved from page S78-S85 in the ESI of the original publication.[14]	The concentration profiles of reactants **18** and **19** and both diastereoisomeric forms of product **21** were measured by *in situ* quantitative $^{19}F\{H\}$ NMR. The concentration of "active non-protonated catalyst" **20** was measured from its integral and its shift in chemical shift during the reaction. The major product was used as the output species in the automatic VTNA calculations.[14]
5	The kinetic dataset used to calculate the reaction orders in **22**, **23** and **24** was gathered using Webplot Digitizer on the concentration profiles of silanol **25** over time from the ESI of the original publication.[15,27]	The concentration profiles of **25** were measured by *in situ* quantitative HNMR. The concentrations profiles of silane **22** and water **23** were then inferred using their reported initial concentrations and the concentration profiles of **25**. The concentration profiles of **24** were assumed to be constant during the course of the reaction. The VTNA overlay plots generated using this dataset were compared to those in the ESI of the original publication and found to be indistinguishable.[15]
6	Kinetic data used to calculate the reaction orders in **27**, **28**	The kinetic data is based on concentration profiles of HBPin **28** which were measured during the standard and different excess

	and **29** was provided by way of an Excel spreadsheet by the authors.	experiments by in-situ ^{11}B NMR. The concentration profiles of ketone **27** and product **30** were calculated from their initial concentrations and the concentration profiles of **28**, assuming perfect mass balance. The concentration profiles of catalyst **29** were assumed to be constant (at the calculated initial concentration).[16]
7	The kinetic data used to calculate the reaction orders in **31**, **32** and **33** was obtained pages S31-S49 of the ESI of the original publication.[7]	The kinetic data was recorded by monitoring the concentrations of **31**, **32** and **33** via continuous HNMR.[7]
8	The kinetic dataset used to calculate the reaction orders in **34**, **35** and **36** was gathered using Webplot Digitizer on the concentration profiles of **35** over time from Figure 11 of the original publication.[8,27]	The concentration profiles of **34** were inferred from its initial concentrations and the corresponding concentration profiles of **35**. The concentration of catalyst **36** was assumed to remain constant at the initial value throughout each experiment.[8]
9	The kinetic dataset used to calculate the reaction orders in **34**, **35** and **36** was gathered using Webplot Digitizer on the concentration profiles of **35** over time from Figures 4 and S11 of the original publication.[8,27]	Same as for Entry 8.
10	The kinetic dataset used to calculate the optimal orders in **38, 39** and **40** was gathered using Webplot Digitizer on the concentration profiles of **41** over time from Figure S4-S6 of the ESI of the original publication.[9,27]	The kinetic dataset was collected by setting up 10 parallel experiments which were stopped at different time points. The crude quantitative HNMR yield of **41** was then measured. The resulting concentration profiles of **41** were used to infer those of **38**. The concentration profiles of **39** and **40** were assumed to remain constant throughout the reaction, which is a crude approximation since at least **39** is consumed to form the product. This data preparation procedure was not stated explicitly by the authors, but could be inferred because this was the only procedure that gave identical overlay plots to those in Figures S5-S6 in the ESI of the publication. Despite its limitations, this data preparation procedure was followed to enable the direct comparison of the results obtained by automatic VTNA in this work.[9]
11	The kinetic data used to calculate the optimal orders in **42**, **43**, **44**, **45**, **46** and **47** was retrieved as an Excel spreadsheet from the Imperial College data repository.[6]	The kinetic dataset was collected by setting up experiments with different initial concentrations of **42**, **43**, **44**, **45**, **46** and **47** and measuring the product concentrations over time by removing aliquots of the reaction mixture, performing an extraction and filtration before measuring the crude product yield by qualitative HNMR. The product concentration profiles were then used to infer the concentration profiles of **42** and **43** based on their initial concentrations. The concentrations of **44**, **45**, **46** and **47** were assumed to remain at the calculated initial concentration throughout the reaction.[6]
12	Kinetic data used to calculate the reaction orders in **49**, **50**, **51** and **52** was provided by way of an Excel spreadsheet by the authors.	The dataset was collected by monitoring the consumption of enyne **49** over the course of different excess experiments by *in situ* IR. The concentration profiles of **50** were inferred from its initial concentrations and the corresponding concentration profiles of **49**. The concentrations of **51** and **52** were assumed to remain constant at their calculated initial concentrations.[10]

13	The kinetic data used to calculate the optimal orders in **54**, **55**, **56** and **57** was retrieved using Webplot Digitizer on the graphs in Figure S4-S6 of the ESI of the original publication.[17,27]	The kinetic dataset was collected by *in situ* HNMR at 21-23 °C. The concentration profiles of **54**, **55**, **56**, **58** and **59** were measured directly while the concentration of **57** was assumed to remain at its initial value throughout the reaction.[17]
14	The kinetic dataset used to calculate the optimal orders in **60, 61** and **62** from the data tables in pages S51-S53 of the ESI of the original publication.[18]	The kinetic data was collected by setting up reactions in a J-Young tube at 23 °C. The concentration profiles of **60, 61** and **63** were measured over time by *in situ* quantitative HNMR. The concentration of **62** was assumed to remain at the initial concentration value throughout each experiment.[18]
15	The kinetic dataset used to calculate the optimal orders in **64, 65** and **66** from the data tables in pages S58-S62 of the ESI of the original publication.[19]	The Kinetic data was collected by setting up reactions at 110 °C under argon and stirring them for a designated time before diluting with cold EtOAc, filtering through Celite and measuring the crude yield by quantitative HNMR (see further details in page S54 of the ESI)[19]. The concentration profile of **67** was measured directly for each experiment. For the different excess experiments in **64** and **65**, the directly measured concentration profiles in the respective reactant was also measured and provided in the ESI, but the other was not provided and was therefore inferred from the product profile. For different excess experiments in catalyst, and the same excess experiments, the concentration profiles of **64** and **65**, which are required for the total VTNA calculation, were inferred from their initial concentrations and the measured profile of **65** (in this work), assuming full mass balance. Hence, the results from the total VTNA calculation is for this dataset deemed less reliable than the sequential VTNA calculation.[19]
16	The kinetic dataset used to calculate the optimal orders in **68, 69, 70**, **71** and **23** from tables 2-9 of the ESI of the original publication.[20]	The kinetic dataset was collected by *in situ* ^{19}FNMR at 45 °C. The concentration profiles of **68** and **69** were measured directly while the concentrations of **70**, **71** and **23** were assumed to remain at their initial values throughout the reaction.[20]

Finally, it is important to recognise that even though a reaction order value has been found to maximise concentration profile overlay, this does not mean that it is the "true" reaction order of the reaction system. Poor datapoint density leading to inaccurate numerical integration, systematic errors, or analytical errors such as incorrect HPLC calibration constants can all effect which order value maximise VTNA overlay. For example, the order values across reaction species in Entries 1, 3 and 4 can safely be reported as 1 as the improvement in overlay score of the calculated order value is very small compared to the rounded value (the rounded reported values are within the 15% overlay score interval of the calculation). However, for kinetic datasets where the calculated order values that maximise concentration profile overlay are significantly shifted from the nearest round value (outside the 15% overlay score interval) it is important to report the true numbers that maximise concentration profile overlay (calculated using Auto-VTNA), to communicate the uncertainty if rounded values are taken forward in discussions on mechanism. In other cases, the authors report more precise order values like

0, 0.9, 1.1 and 1 in Entry 12[10] or 0.85, -0.6 and 0.9 in Entry 5,[15] favouring closeness to the order value that maximises concentration profile overlay rather than rounding the order values to the nearest whole number. It should also be noted that order value calculations will only yield meaningful results if reaction progress profiles have been inputted from experiments with the same solvent and at the same temperature, but different initial concentrations of reactants, catalysts or additives.

9. Automatic VTNA analysis of Knorr Pyrazole kinetic data

As part of an effort to test the Auto-VTNA Calculator graphical user interface, the kinetic data collected by Schrecker *et al* on the Knorr pyrazole synthesis was analysed.[29] Since reactant concentration profiles could not be collected for this reaction, reactant concentrations were initially inferred from the initial concentrations and product profiles, assuming perfect mass balance.

Utilising Auto-VTNA, this gave rise to reaction orders of 1.29 and 1.11 and 15% order intervals of 0.16 and 0.17 in phenyl hydrazine and acetylacetone respectively (Fig. 48). However, upon inspection of the proposed microkinetic model from Schrecker *et al.*, it was noted that mass balance is not achieved from reactants and product alone due to the presence of persistent reaction intermediates. Hence, reactant concentrations over time cannot simply be inferred from the corresponding product concentrations, rendering conventional VTNA inapplicable since the true concentration profiles could not be measured.

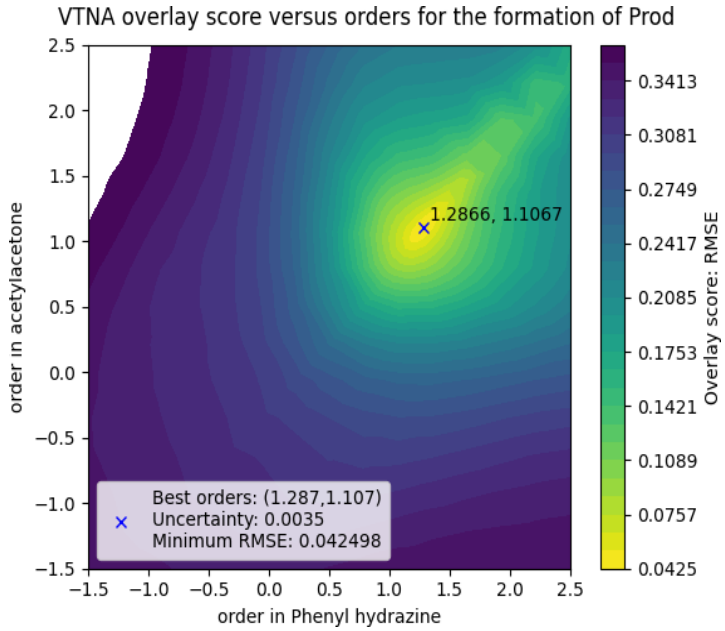


Figure S48: Overlay score versus order values in acetylacetone and phenyl hydrazine for the concentration profiles with inferred reactant concentration profiles (assuming perfect mass balance). Kinetic data from all recorded time ramp experiments were included.[29]

Instead, the reported microkinetic model involving several intermediates and autocatalytic steps was applied to generate simulated concentration profiles for every reaction component. Auto-VTNA could then be used to analyse the simulated data, revealing reaction order values in phenyl hydrazine and acetylacetone of 0.84 and 0.77 respectively with narrow 15% overlay score intervals of approximately 0.055. Interestingly, reaction order values of 0.95 and with complete linearisation was seen when defining the sum of the intermediates and products concentration profiles as the output species, suggesting that the reaction orders below 1 could partly be due to the presence of persistent intermediates which slowly convert to generate the product (Fig. S49).

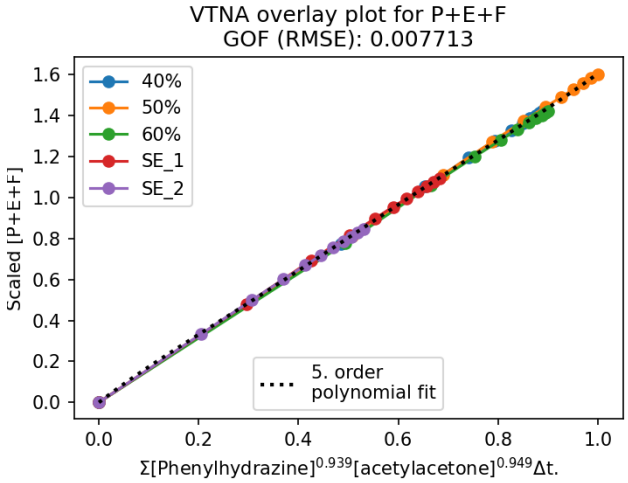


Figure S49: VTNA overlay plot achieved by defining the output concentration species as the sum of the simulated product (P) and intermediate concentration profiles (E and F). SE_1 and SE_2 refers to same excess experiments. 40%, 50% and 60% refer to the pump flowrate of the phenylhydrazine solution as a percentage of the total flowrate.

The simulated product concentration profiles were then used to infer reactant concentration profiles. This allowed investigation of the effect of missing initial datapoints from data generated by transient flow time ramp experiments on reaction order values obtained by automatic VTNA by simulation and inclusion of these early time points.[29,30] Interestingly, order values in phenyl hydrazine and acetylacetone of 1.20 and 0.99 were obtained respectively, demonstrating that loss of initial datapoints in transient flow concentration can give rise to inaccurate results. Although this does not affect the outcome in this case, as the inferred reactant concentration profiles were incorrect, it does show that more work is required to expand Auto-VTNA to kinetic data with missing initial time points as is often obtained by transient flow methods.

10. Code availability

The source code for the project is available on GitHub at https://github.com/ddalland/Auto-VTNA. This page also contains a .exe file for the GUI which can be downloaded and used directly (currently only for Windows). Additionally, the project is published on PyPI and can be found at https://pypi.org/project/auto-vtna/1.0.2/.

To run the graphical user interface (GUI) for the project, users need to download the "Auto_VTNA_GUI.exe" file from the Git Large File Storage (LFS), linked from the above Auto-VTNA Github page. Alternatively, the file Auto_VTNA_GUI.py can be located in the Auto-VTNA directory and run using a Python development environment such as Visual Studio Code or an equivalent program.

11. References

1 D. M. Day, T. J. Farmer, J. Sherwood and J. H. Clark, An experimental investigation into the kinetics and mechanism of the aza-Michael additions of dimethyl itaconate, *Tetrahedron*, 2022, **121**, 132921.

2 F. Bork, S. Clark, P. Burland, D. Sale and J. Hein, Kinalite A User-Friendly Online Tool for Automated Variable Time Normalization Analysis (VTNA), *Pre-print*, , DOI:10.26434/chemrxiv-2024-m15pr.

3 D. Schmitz, Polyfit, https://polyfit.readthedocs.io/en/latest/, (accessed 13 October 2023).

4 D. M. Day, T. J. Farmer, J. Granelli, J. H. Lofthouse, J. Lynch, C. R. McElroy, J. Sherwood, S. Shimizu and J. H. Clark, Reaction Optimization for Greener Chemistry with a Comprehensive Spreadsheet Tool, *Molecules*, 2022, **27**, 1–12.

5 O. J. Newton, K. Hellgardt, J. Richardson and K. K. M. Hii, 'Homeopathic' palladium catalysis? The observation of distinct kinetic regimes in a ligandless Heck reaction at (ultra-)low catalyst loadings, *J Catal*, 2023, **424**, 29–38.

6 J. I. Higham and J. A. Bull, Amine-Catalyzed Copper-Mediated C-H Sulfonylation of Benzaldehydes via a Transient Imine Directing Group, *Angewandte Chemie International Edition*, , DOI:10.26434/chemrxiv-2021-bhs2k-v2.

7 S. Aikonen, M. Muuronen, T. Wirtanen, S. Heikkinen, J. Musgreave, J. Burés and J. Helaja, Gold(I)-Catalyzed 1,3-O-Transposition of Ynones: Mechanism and Catalytic Acceleration with Electron-Rich Aldehydes, *ACS Catal*, 2018, **8**, 960–967.

8 B. Wei, J. C. Sharland, P. Lin, S. M. Wilkerson-Hill, F. A. Fullilove, S. McKinnon, D. G. Blackmond and H. M. L. Davies, In Situ Kinetic Studies of Rh(II)-Catalyzed Asymmetric Cyclopropanation with Low Catalyst Loadings, *ACS Catal*, 2020, **10**, 1161–1170.

9 P. St. Onge, S. I. Khan, A. Cook and S. G. Newman, Reductive Cleavage of C(sp2)-CF3 Bonds in Trifluoromethylpyridines, *Org Lett*, 2023, **25**, 1030–1034.

10 J. H. Herbort, R. F. Lalisse, C. M. Hadad and T. V. Rajanbabu, Cationic Co(I) Catalysts for Regiodivergent Hydroalkenylation of 1,6-Enynes: An Uncommon cis-β-C-H Activation Leads to Z-Selective Coupling of Acrylates, *ACS Catal*, 2021, **11**, 9605–9617.

11 D. C. Braddock, B. M. J. Lancaster, C. J. Tighe and A. J. P. White, Surmounting Byproduct Inhibition in an Intermolecular Catalytic Asymmetric Alkene Bromoesterification Reaction as Revealed by Kinetic Profiling, *Journal of Organic Chemistry*, 2023, **88**, 8904–8914.

12 A. Matviitsuk, M. D. Greenhalgh, D. B. Antúnez, A. M. Z. Slawin and A. D. Smith, Aryloxide-Facilitated Catalyst Turnover in Enantioselective α,β-Unsaturated Acyl Ammonium Catalysis, *Angewandte Chemie*, 2017, **129**, 12450–12455.

13 S. M. Smith, M. D. Greenhalgh, T. Feoktistova, D. M. Walden, J. E. Taylor, D. B. Cordes, A. M. Z. Slawin, P. H. Y. Cheong and A. D. Smith, Scope, Limitations and Mechanistic Analysis of the HyperBTM-Catalyzed Acylative Kinetic Resolution of Tertiary Heterocyclic Alcohols**, *European J Org Chem*, , DOI:10.1002/ejoc.202101111.

14 C. McLaughlin, A. M. Z. Slawin and A. D. Smith, Base-free Enantioselective C(1)-Ammonium Enolate Catalysis Exploiting Aryloxides: A Synthetic and Mechanistic Study, *Angewandte Chemie*, 2019, **131**, 15255–15263.

15 E. Antico, M. Leutzsch, N. Wessel, T. Weyhermüller, C. Werlé and W. Leitner, Selective oxidation of silanes into silanols with water using [MnBr(CO)5] as a precatalyst, *Chem Sci*, 2022, **14**, 54–60.

16 M. R. Elsby, M. Son, C. Oh, J. Martin, M. H. Baik and R. T. Baker, Mechanistic Study of Metal-Ligand Cooperativity in Mn(II)-Catalyzed Hydroborations: Hemilabile SNS Ligand Enables Metal Hydride-Free Reaction Pathway, *ACS Catal*, 2021, **11**, 9043–9051.

17 Y. Nishii, M. Ikeda, Y. Hayashi, S. Kawauchi and M. Miura, Triptycenyl Sulfide: A Practical and Active Catalyst for Electrophilic Aromatic Halogenation Using N-Halosuccinimides, *J Am Chem Soc*, 2020, **142**, 1621–1629.

18 S. D. Mendoza, M. Rombola, Y. Tao, S. J. Zuend, R. Götz, M. J. McLaughlin and S. E. Reisman, Expanding the Chiral Monoterpene Pool: Enantioselective Diels-Alder Reactions of α-Acyloxy Enones, *Org Lett*, 2022, **24**, 3802–3806.

19 D. Antermite, A. J. P. White, L. Casarrubios and J. A. Bull, On the Mechanism and Selectivity of Palladium-Catalyzed C(sp3)-H Arylation of Pyrrolidines and Piperidines at Unactivated C4 Positions: Discovery of an Improved Dimethylaminoquinoline Amide Directing Group, *ACS Catal*, 2023, **13**, 9597–9615.

20 C. Pfeffer, N. Wannenmacher, W. Frey and R. Peters, Stereo- and regioselective dimerization of alkynes to enynes by bimetallic syn-carbopalladation, *ACS Catal*, 2021, **11**, 5496–5505.

21 J. F. Epperson, On the Runge Example, *The American Mathematical Monthly*, 1987, **94**, 329–341.

22 F. E. Jr. Harrell, in *Regression Modeling Strategies*, Springer, 2nd edn., 2001, pp. 21–21.

23 X. Zhang, Y. Xin, N. Xie and Y. Zhao, Shading Surface Estimation using Piecewise Polynomials for Binarizing Unevenly Illuminated Document Images, *International Journal of Intelligent Engineering & Systems*, 2014, **7**, 1–9.

24 I. Chivers and J. Sleightholme, in *Introduction to Programming with Fortran*, Springer, London, 4th edn., 2018.

25 J. I. Higham and J. A. Bull, Amine-Catalyzed Copper-Mediated C−H Sulfonylation of Benzaldehydes via a Transient Imine Directing Group**, *Angewandte Chemie*, , DOI:10.1002/ange.202202933.

26 J. Burés, Variable Time Normalization Analysis: General Graphical Elucidation of Reaction Orders from Concentration Profiles, *Angewandte Chemie - International Edition*, 2016, **55**, 16084–16087.

27 A. Rohatgi, *WebPlotDigitizer User Manual Version 4.6*, 2022.

28 D. Dalland, Auto-VTNA, https://github.com/ddalland/Auto-VTNA, (accessed 12 June 2024).

29 L. Schrecker, J. Dickhaut, C. Holtze, P. Staehle, M. Vranceanu, K. Hellgardt and K. K. Hii, Discovery of unexpectedly complex reaction pathways for the Knorr pyrazole synthesis via transient flow, *React Chem Eng*, 2022, **8**, 41–46.

30 S. Mozharov, A. Nordon, D. Littlejohn, C. Wiles, P. Watts, P. Dallin and J. M. Girkin, Improved method for kinetic studies in microreactors using flow manipulation and noninvasive raman spectrometry, *J Am Chem Soc*, 2011, **133**, 3601–3608.

31 J. Burés, Variable Time Normalization Analysis: General Graphical Elucidation of Reaction Orders from Concentration Profiles, *Angewandte Chemie - International Edition*, 2016, **55**, 16084–16087.

Appendices

A1. Overlay versus reaction order plots for simulated kinetic data

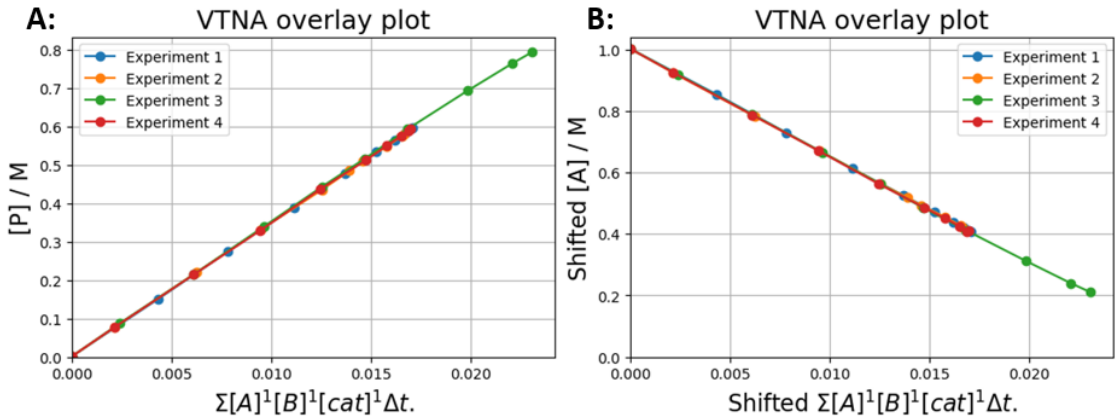
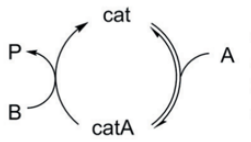


Figure SA1: VTNA overlay plots for time axis normalisation using the concentration profiles of reactants A, B and catalyst cat. A: Product concentrations used to show overlay. B: Shifted concentration profiles of A used to show overlay.



Experiment 1: $[A]_0 = 1.0\ M, [B]_0 = 0.6\ M, [cat]_T = 0.01$

Experiment 2: $[A]_0 = 0.6\ M, [B]_0 = 0.6\ M, [cat]_T = 0.01$

Experiment 3: $[A]_0 = 1.0\ M, [B]_0 = 0.8\ M, [cat]_T = 0.01$

Experiment 4: $[A]_0 = 1.0\ M, [B]_0 = 0.6\ M, [cat]_T = 0.03$

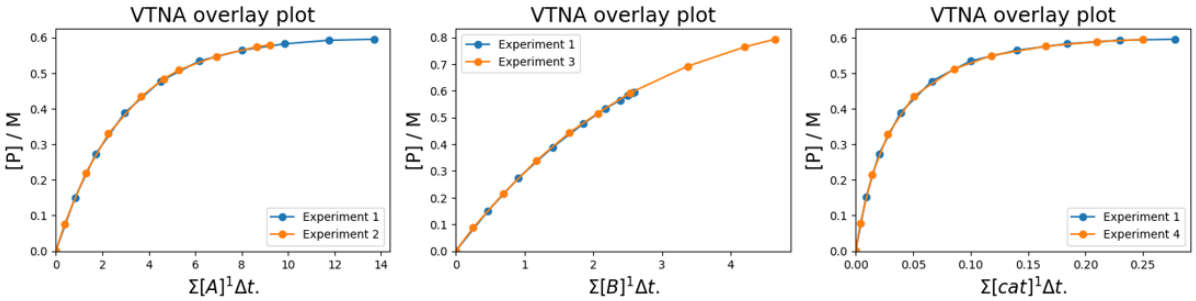


Figure SA2: Overlay plots for different excess experiments for reactants A and B as well as catalyst C. Variable time normalisation was performed using Normal_VTNA followed by the .plot_VTNA() method.

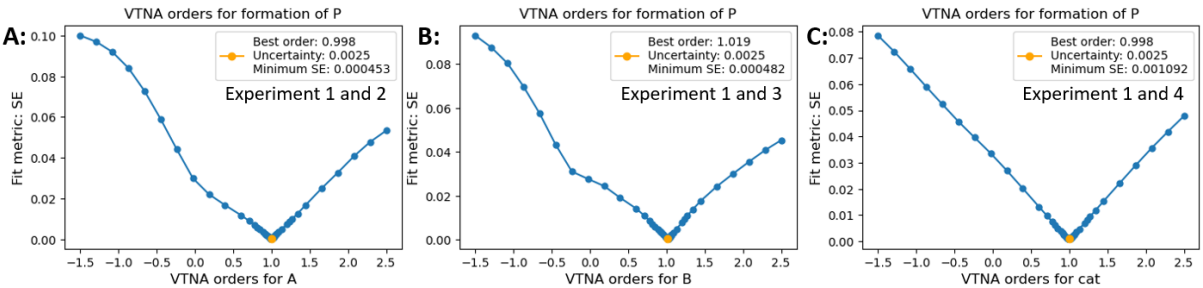


Figure SA3: Plots showing the overlay scores (standard error of 7th degreemonotonic polynomial fits) against the reaction order in A, B and cat (C) for the simulated data presented in Figure 2 in Jordi Bures' 2016 paper.[26]

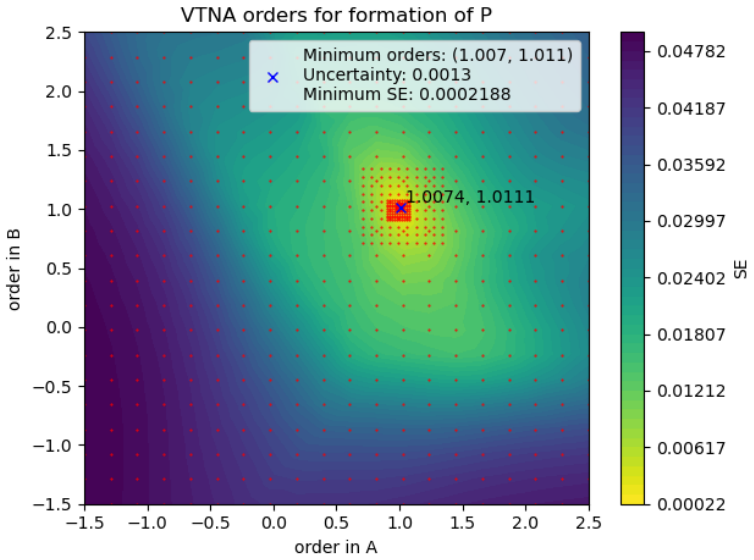
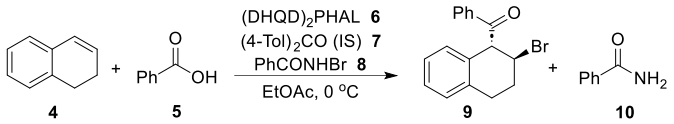


Figure SA4: Contour plot illustrating the degree of concentration profile overlay across different reaction order values in A and B for simulated data presented in Figure 2 of Jordi Bures' 2016 publication on VTNA.[31] The red dots represent reaction order values for which concentration profile overlay has been quantified and illustrates the iterative nature of the Auto-VTNA algorithm.

A2. VTNA overlay plots for orders obtained by Auto-VTNA for Table 2

A2.1. Entry 1



4: Alkene **5**: $PhCO_2H$ **6:** Catalyst
7: Internal Std **8**: PhCONBr

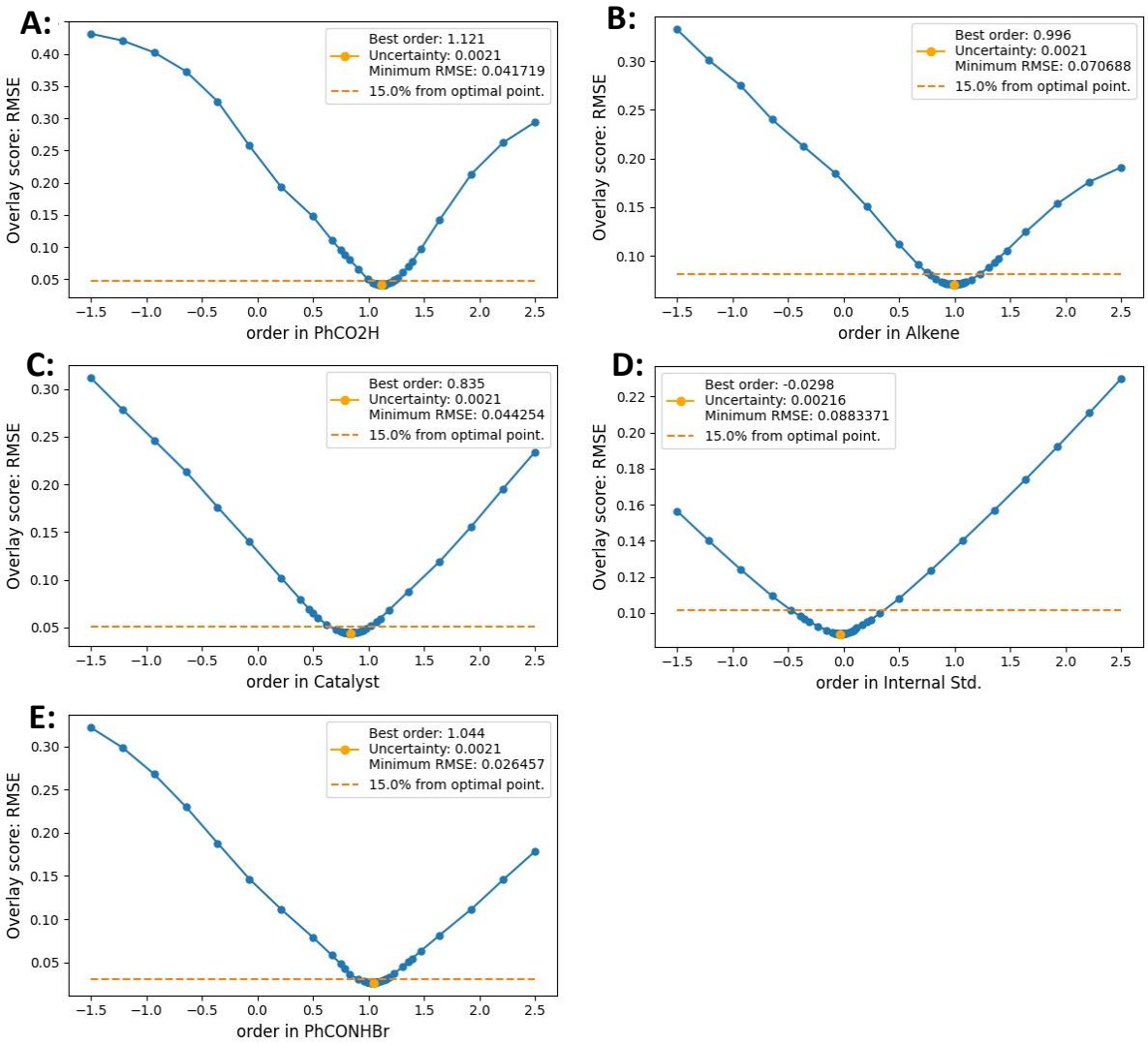


*Figure SA5: Overlay score vs. order plots for automatic sequential VTNA to obtain the order of the reaction in alkene **4** (A), acid **5** (B), catalyst **6** (C), internal standard **7** (D) and brominating reagent **8** (E). Product **9** was defined as the output reaction species in the calculations.*[11]

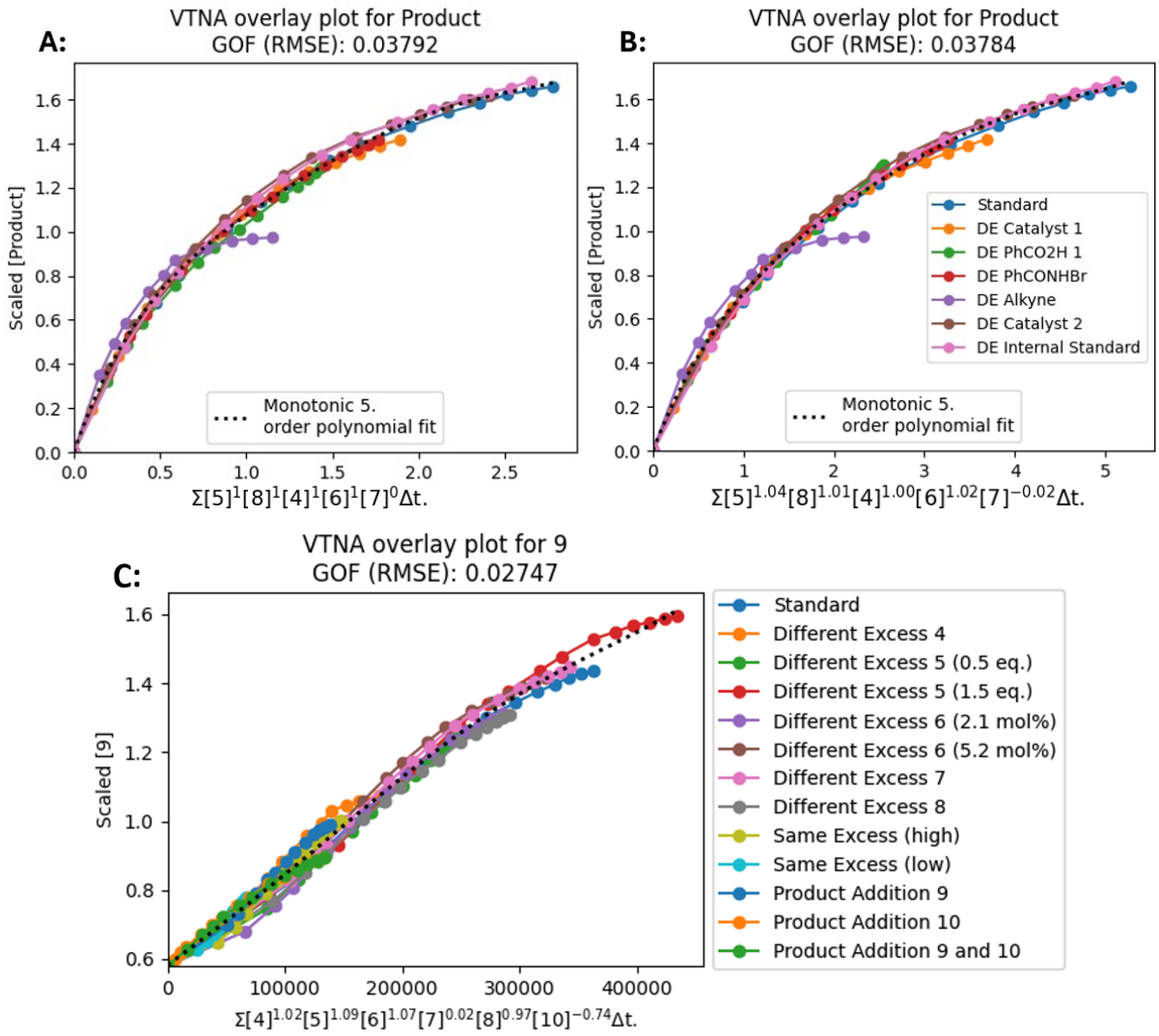
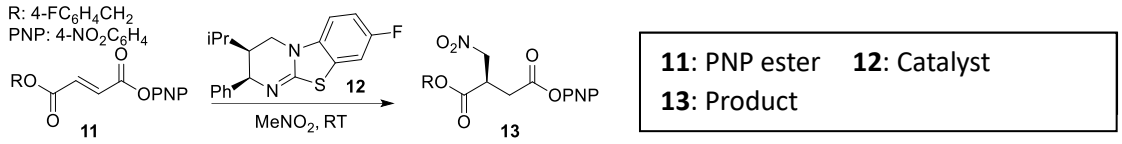


*Figure SA6: Total VTNA overlay plot with the reported (A) and calculated (B) order values in **4**, **5**, **6**, **7** and **8**. The order values of plot B were calculated via automatic total VTNA. C): total VTNA overlay plot for the calculated reaction order values in **4**, **5**, **6**, **7**, **8** and **9**.*[11]

A2.2. Entry 2



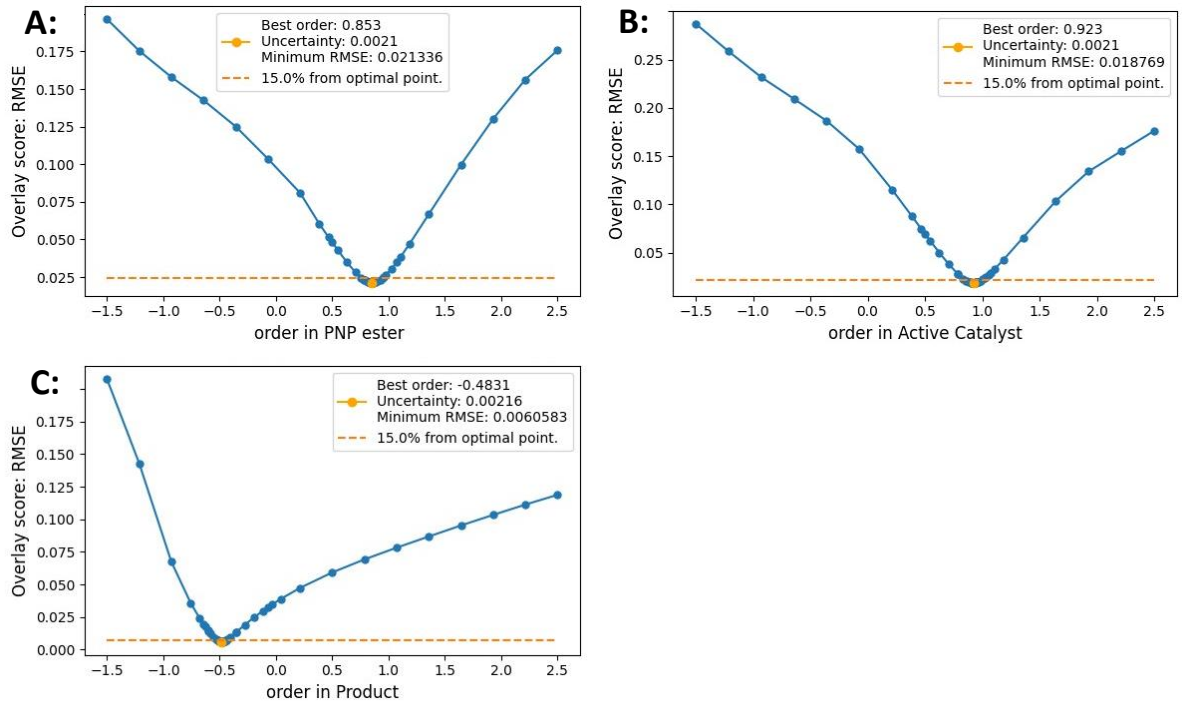


*Figure SA7: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the reaction in PNP ester **11** (A), the active catalyst **12** (B), and the product **13** (C). The product **13** was defined as the output reaction species in the calculations.* [12]

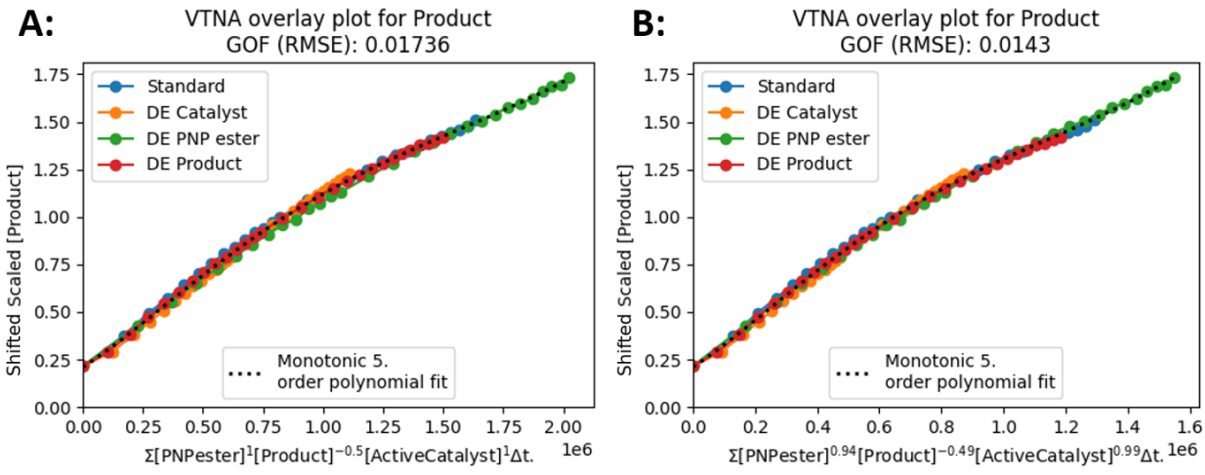
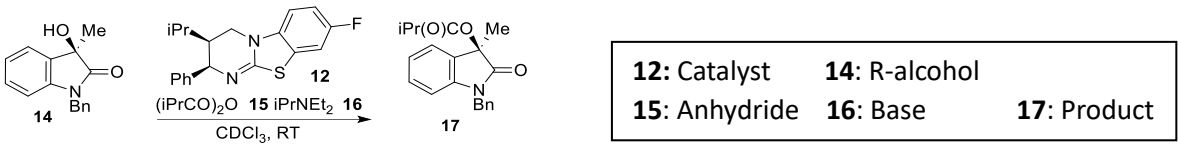


*Figure SA8: Total VTNA overlay plot for the formation of **13** with the reported (A) and calculated (B) order values in in PNP ester **11**, the active catalyst **12**, and the product **13**. The order values of plot B were calculated via automatic total VTNA.* [12]

A2.3. Entry 3



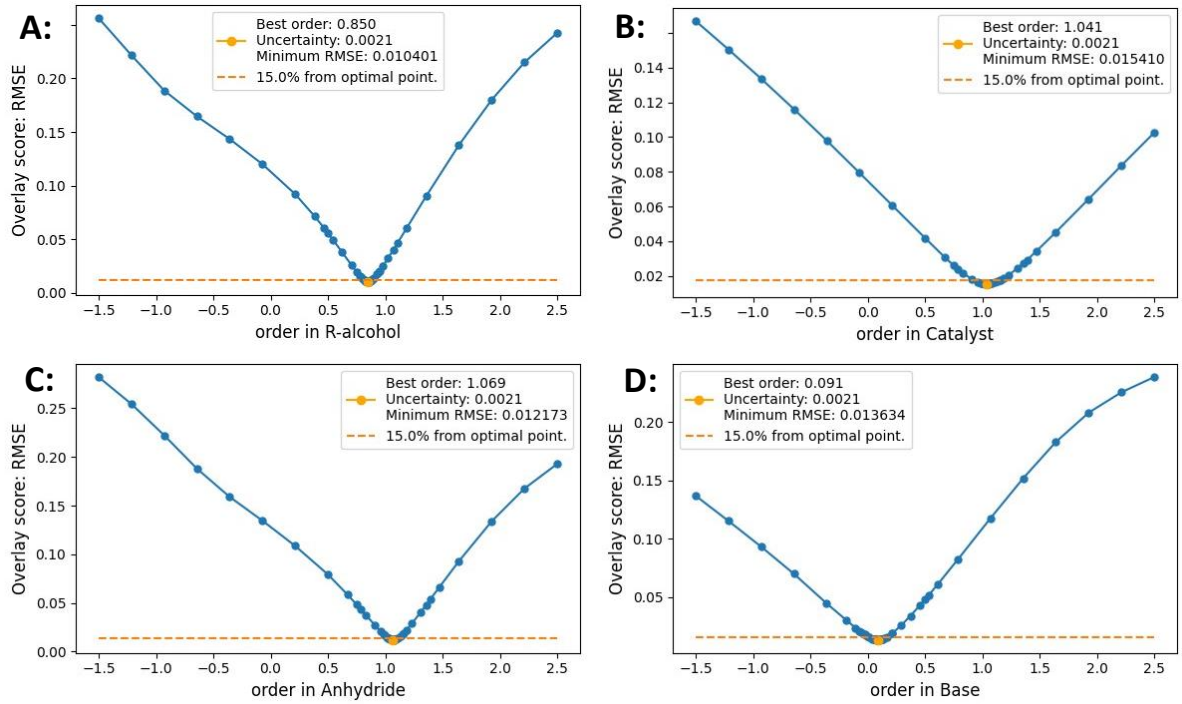


*Figure SA9: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the reaction in R-alcohol **14**, catalyst **12**, anhydride **15** and base **14**. Product **17** was defined as the output reaction species in the calculations.*[13]

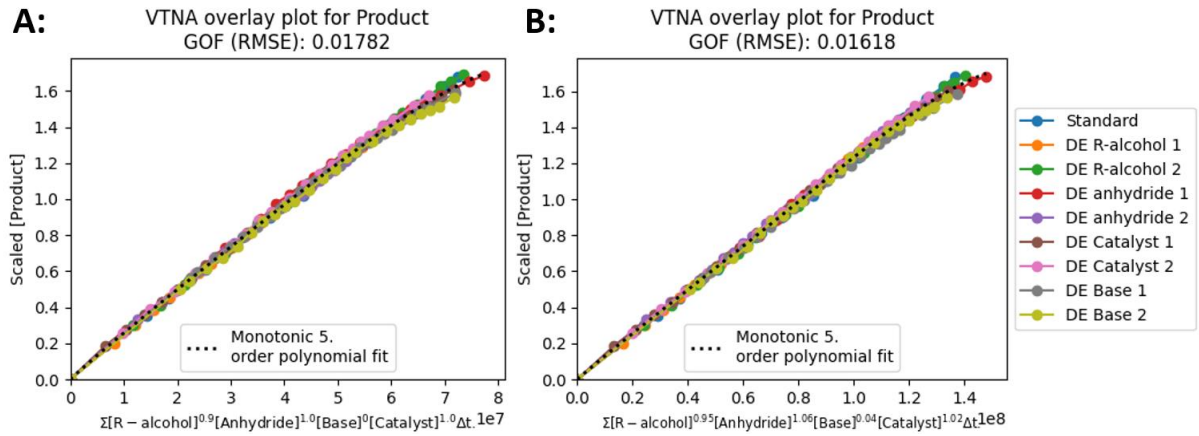
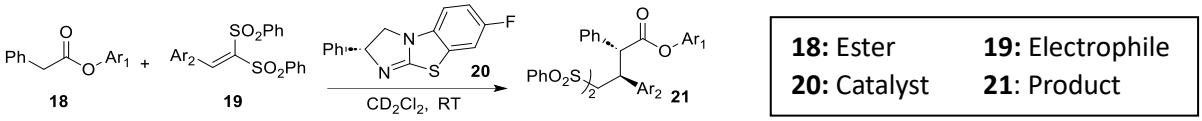


*Figure SA10: Total VTNA overlay plot for the formation of **17** with the reported (A) and calculated (B) order values in in R-alcohol **14**, catalyst **12**, anhydride **15** and base **14**. The order values of plot B were calculated via automatic total VTNA.*[13]

A2.4. Entry 4



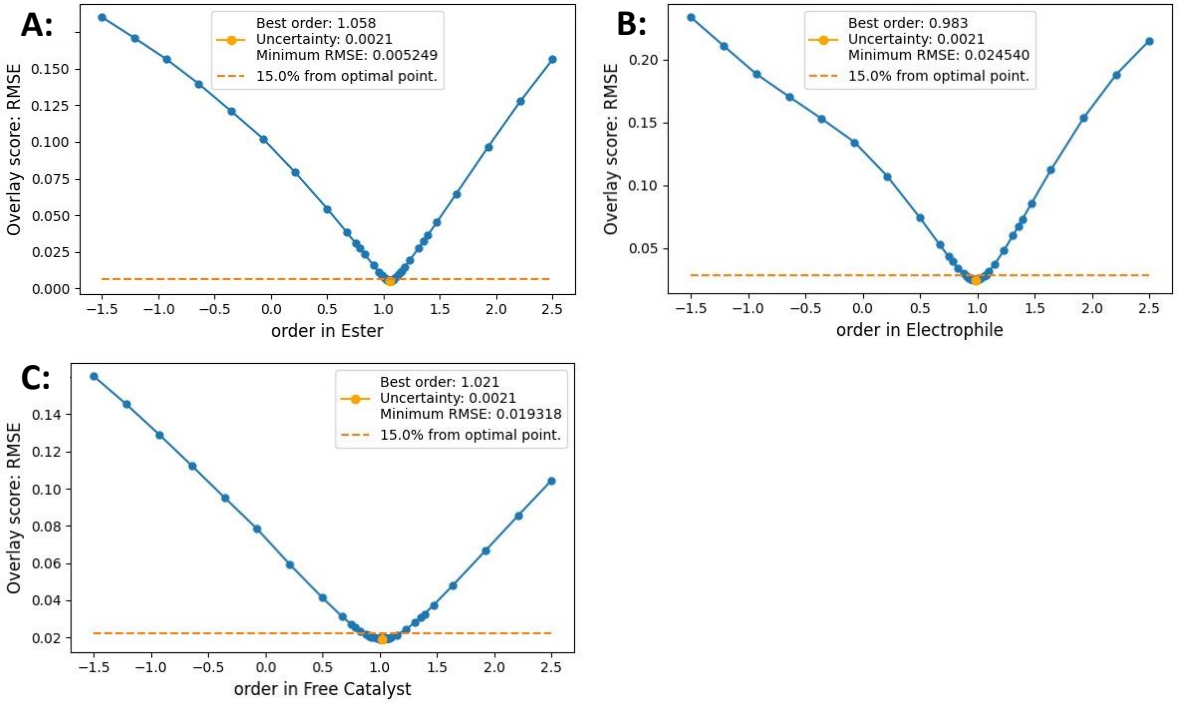


*Figure SA11: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the reaction in ester **18** (A), the electrophile **19** (B) and the free catalyst **20** (C). The product **21** was defined as the output reaction species in the calculations.*[14]

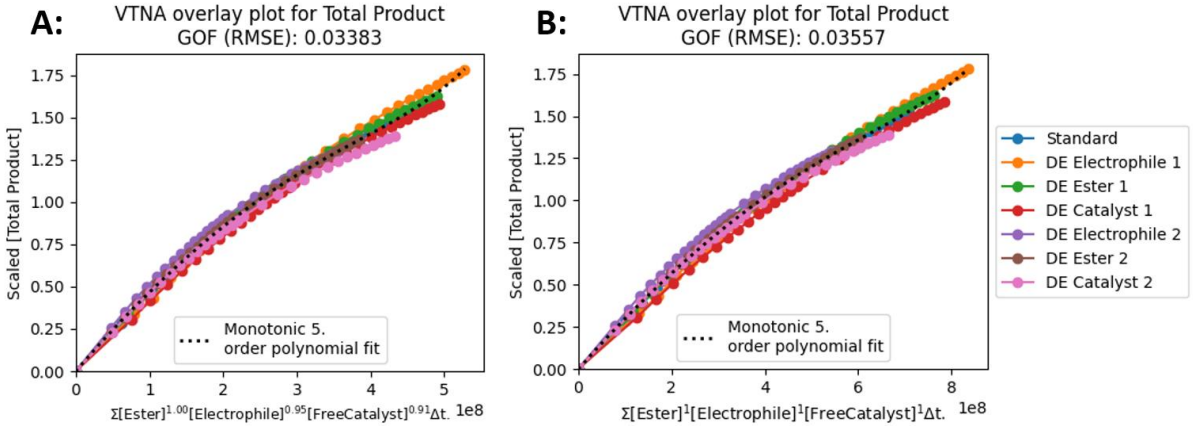
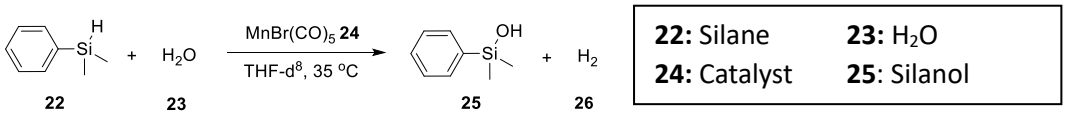


*Figure SA12: Total VTNA overlay plot for the formation of **21** with the reported (A) and calculated (B) order values in ester **18**, the electrophile **19** and the free catalyst **20**. The order values of plot B were calculated via automatic total VTNA.*[14]

A2.5. Entry 5



22: Silane **23:** H_2O
24: Catalyst **25**: Silanol

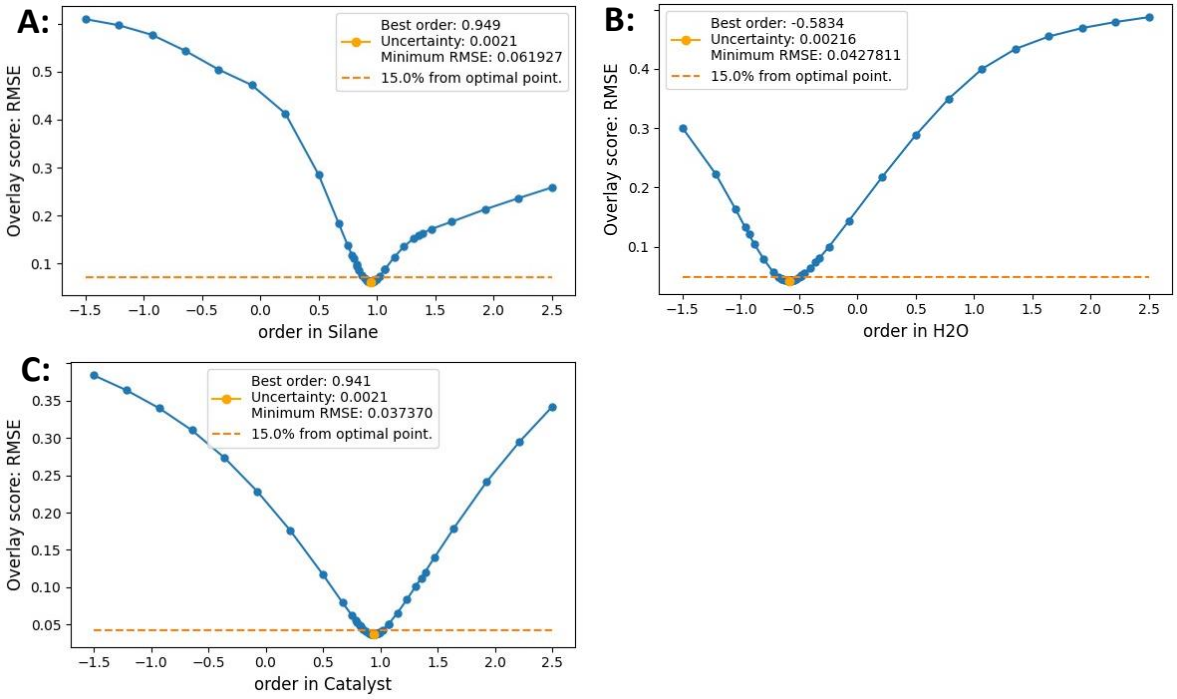


*Figure SA13: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the silane oxidation reaction in silane **22** (A), water **23** (B) and the catalyst **24** (C). The product **25** was defined as the output reaction species in the calculations.*[15]

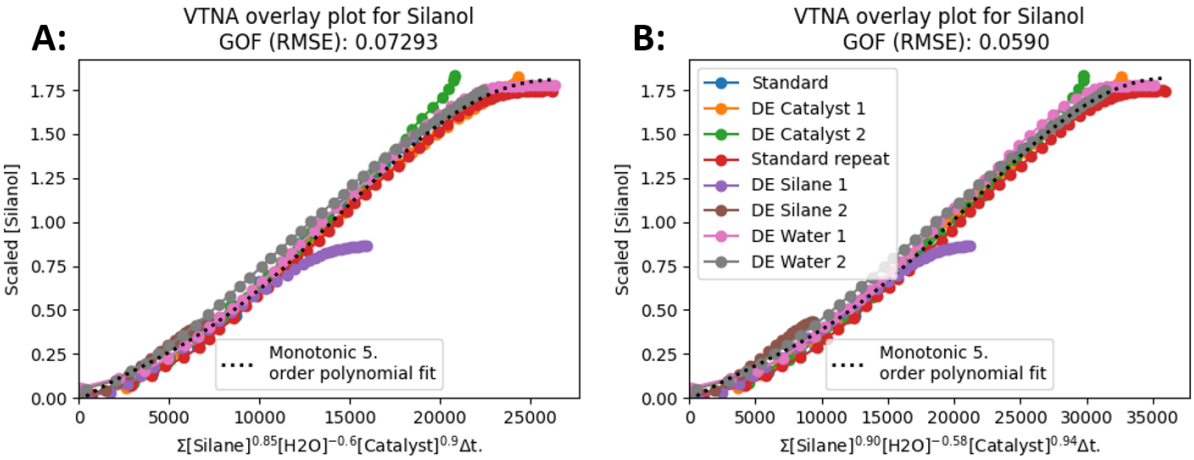
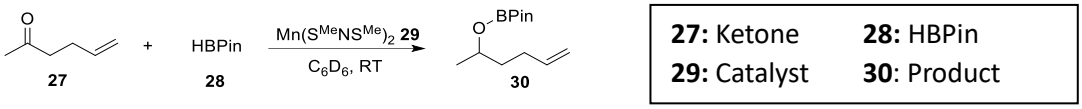


*Figure SA14: Total VTNA overlay plot for the formation of **25** with the reported (A) and calculated (B) order values in **22**, **23** and **24**. The order values of plot B were calculated via automatic total VTNA.*[15] *The legend in plot B also applies to plot A.*

A2.6. Entry 6



27: Ketone **28:** HBPin
29: Catalyst **30**: Product

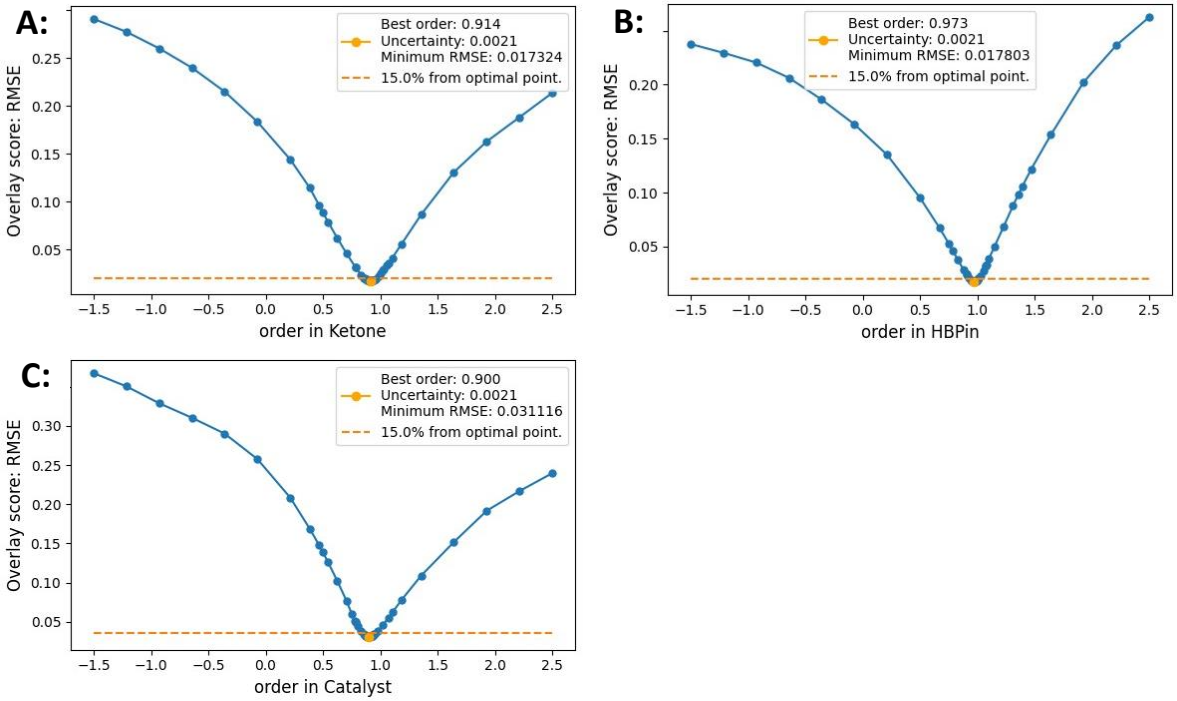


*Figure SA15: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the reduction in ketone **27** (A), HBPin **28** (B) and catalyst **29** (C). The product **30** was defined as the output reaction species in the calculations.* [16]

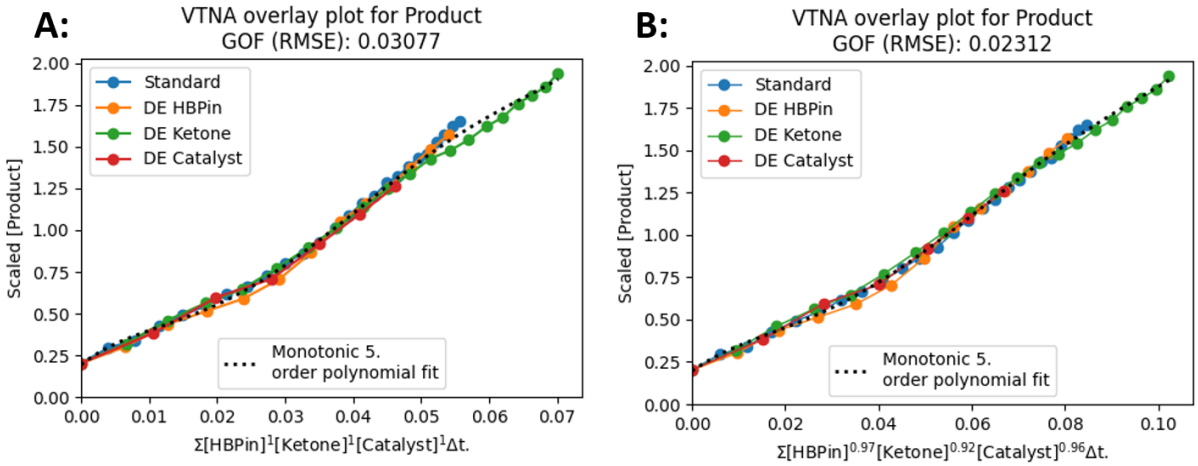
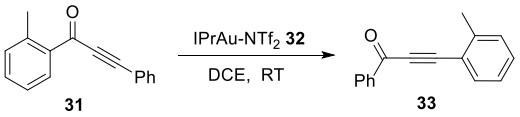


*Figure SA16: Total VTNA overlay plot for the formation of **30** with the reported (A) and calculated (B) order values in ketone **27**, HBPin **28** and catalyst **29.** The order values of plot B were calculated via automatic total VTNA.*[16]

A2.7. Entry 7



31: Ynone **32:** Active catalyst
33: Product

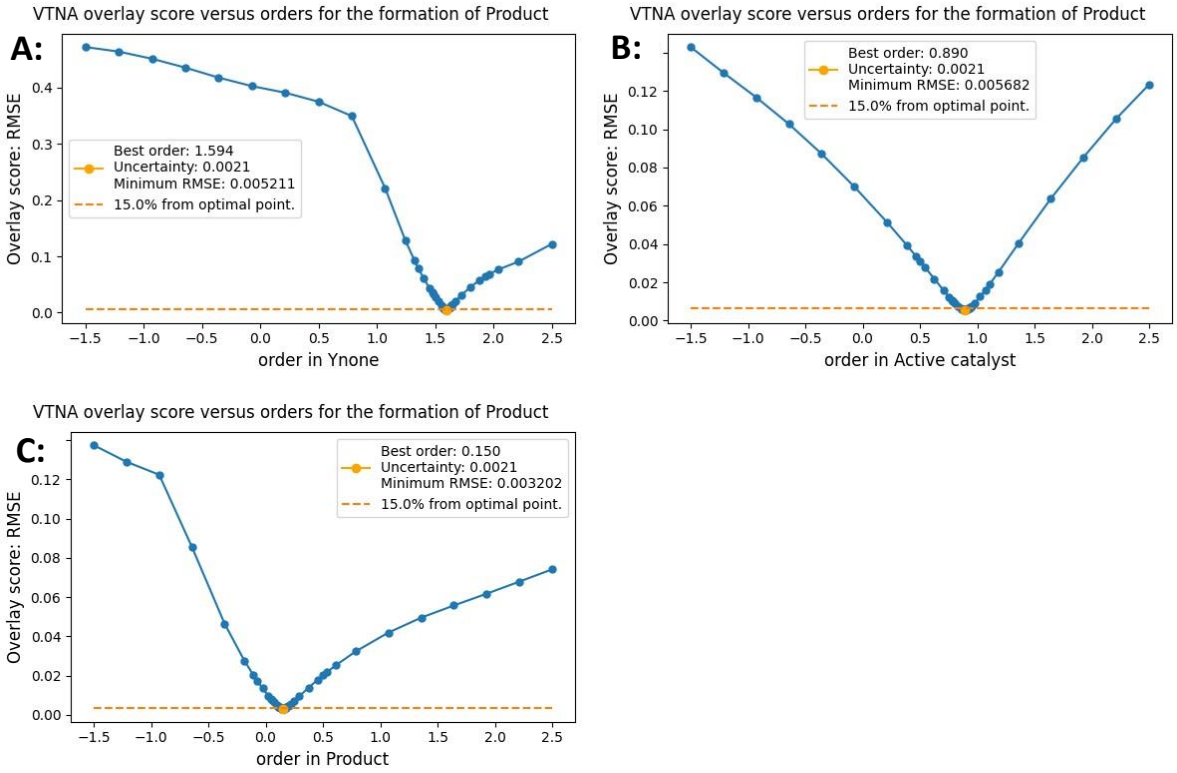


*Figure SA17: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the ynone transposition reaction in ynone **31** (A), active catalyst **32** (B) and product **33** (C). The product **33** was defined as the output reaction species in the calculations.*[7]

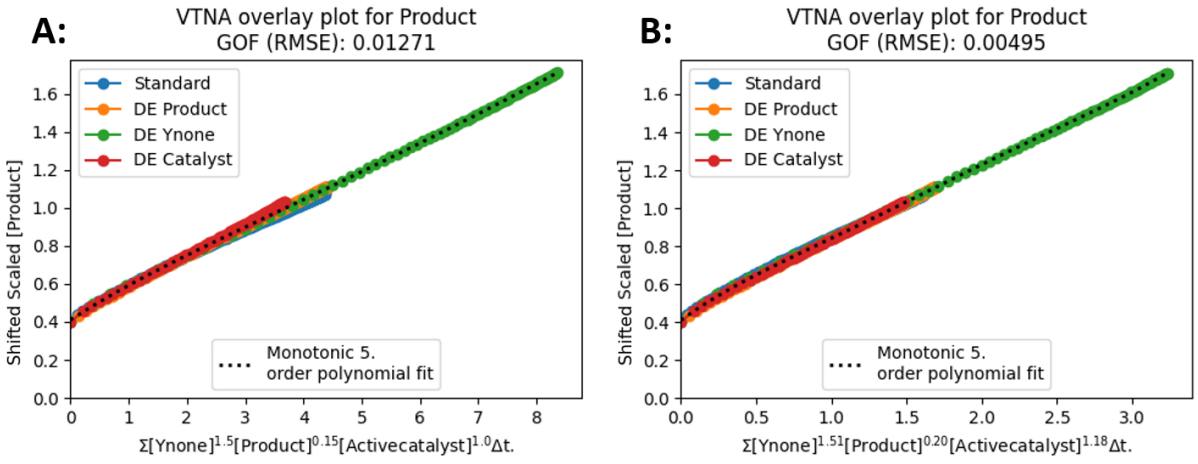
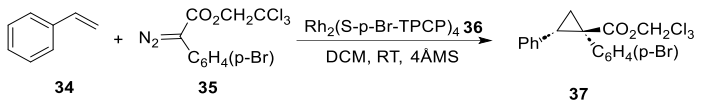


*Figure SA18: Total VTNA overlay plot for the formation of **33** with the reported (A) and calculated (B) order values in in ynone **31**, catalyst **32** and product **33**. The order values of plot B were calculated via automatic total VTNA.*[7]

A2.8. Entry 8



34 35 37

34: Styrene **35:** Diazo compound
36: Catalyst

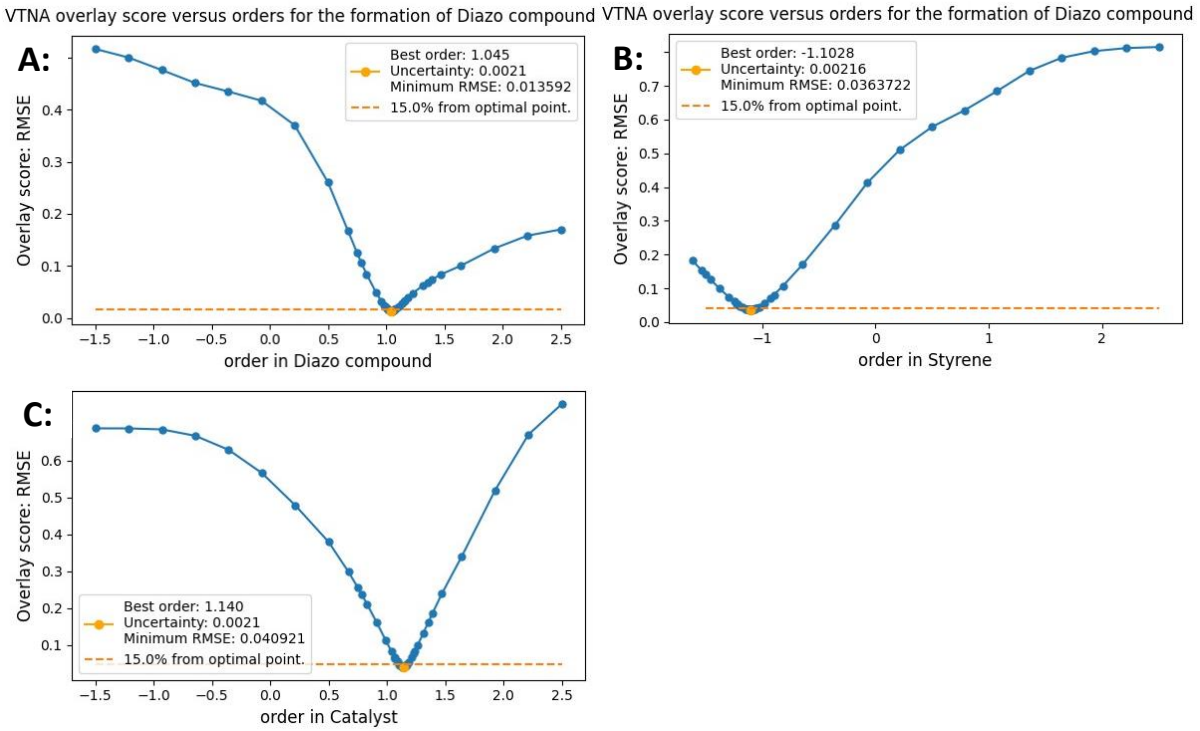


*Figure SA19: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the enantioselective cyclopropanation reaction in DCM in styrene **34**, diazo compound **35** and catalyst **36**. The diazo compound **35** was defined as the output reaction species in the calculations.*[8]

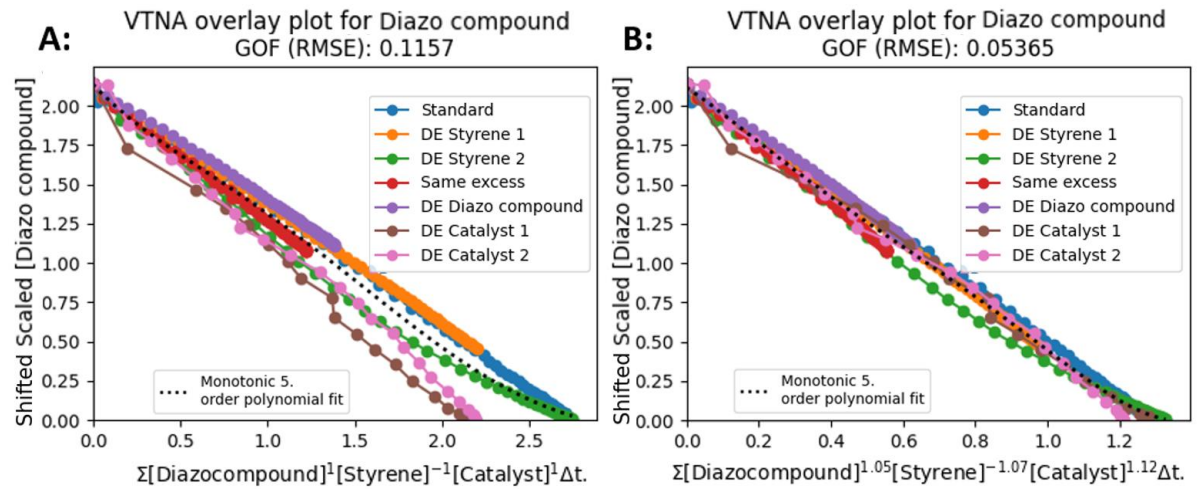
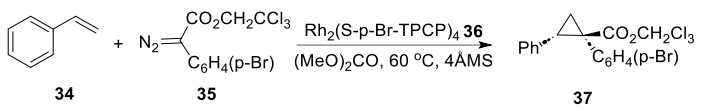


*Figure SA20: Total VTNA overlay plot for the consumption of **35** in DCM with the reported (A) and calculated (B) order values in styrene **34**, diazo compound **35** and catalyst **36**. The order values of plot B were calculated via automatic total VTNA.*[8]

A2.9. Entry 9



34: Styrene **35:** Diazo compound
36: Catalyst

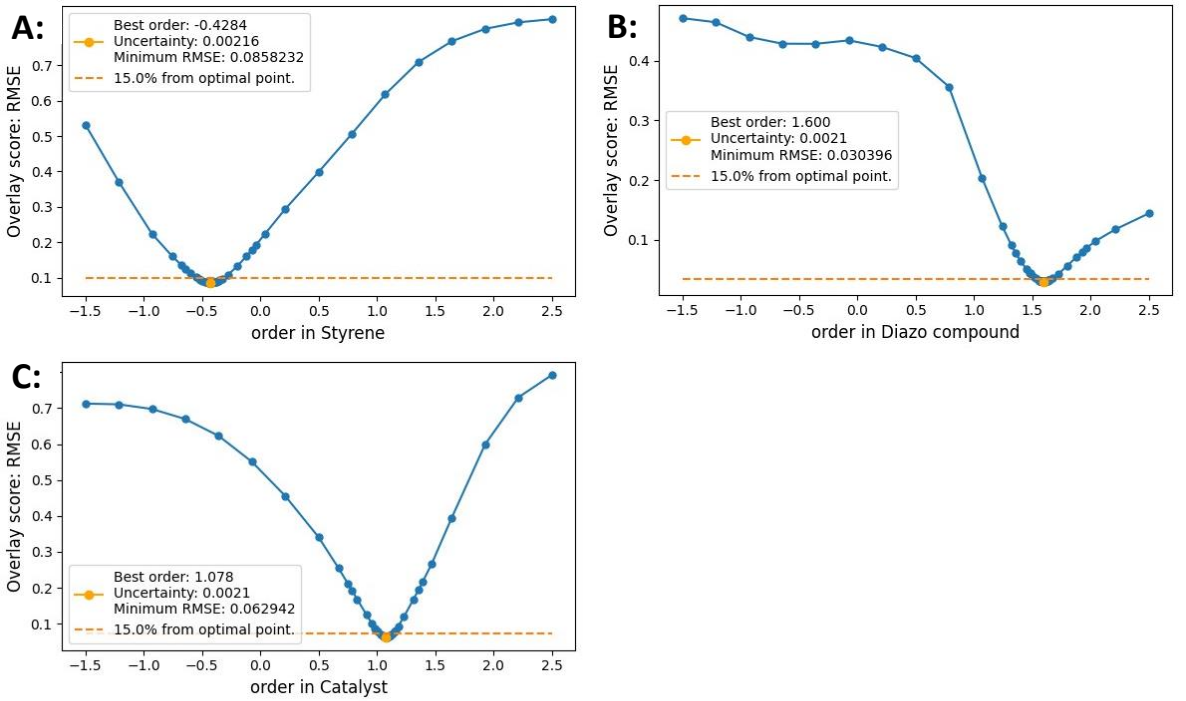


*Figure SA21: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the enantioselective cyclopropanation reaction in dimethyl carbonate in styrene **34** (A), diazo compound **35** (B) and catalyst **36** (C). The diazo compound **35** was defined as the output reaction species in the calculations.*[8]

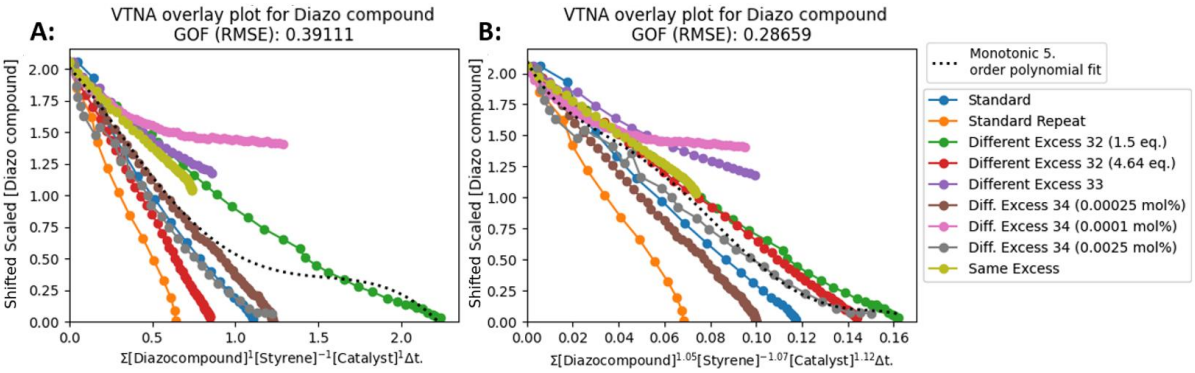


*Figure SA22: Total VTNA overlay plot for the consumption of **35** in dimethyl carbonate with the reported (A) and calculated (B) order values in styrene **34**, diazo compound **35** and catalyst **36**. The order values of plot B were calculated via automatic total VTNA.*[8]

A2.10. Entry 10

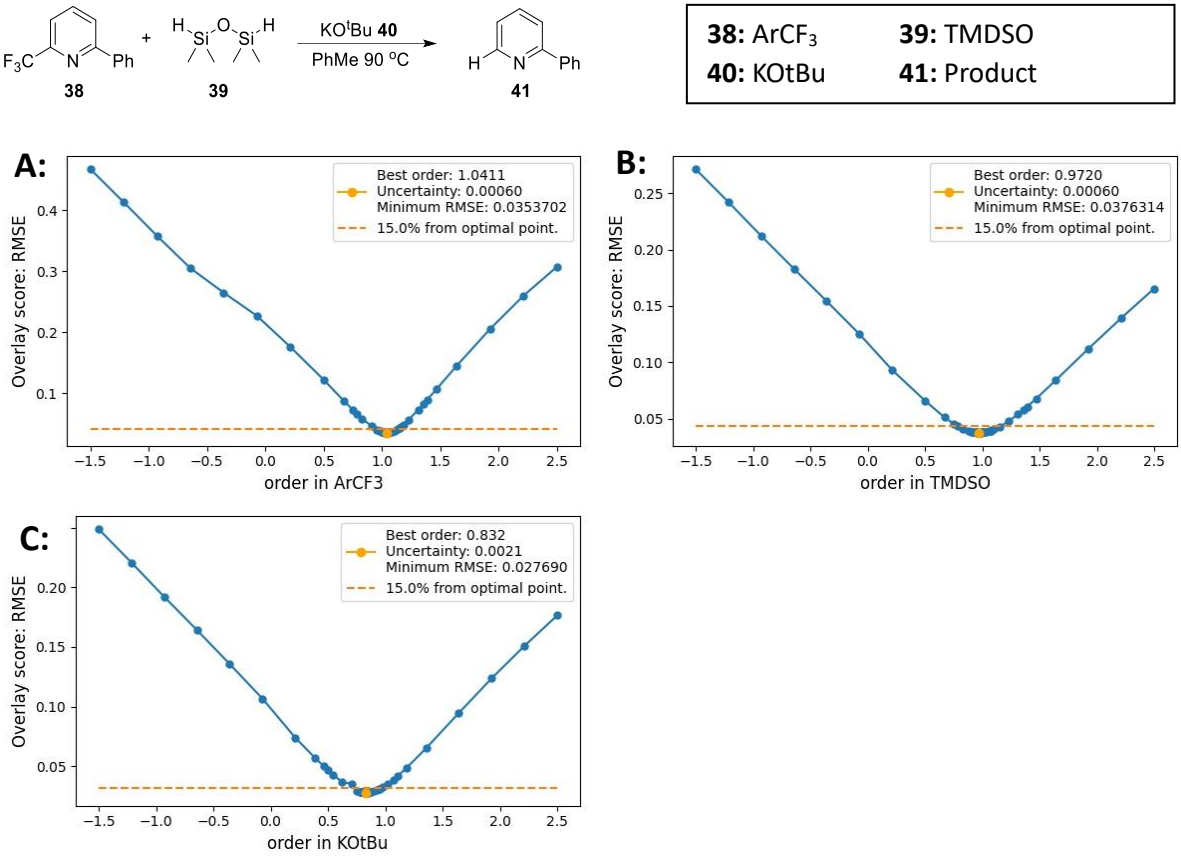


*Figure SA23: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the detrifluoromethylation reaction in $ArCF_3$ **38** (A), TMDSO **39** (B) and KtOBu **40** (C). The product **41** was defined as the output reaction species in the calculations.*[9]

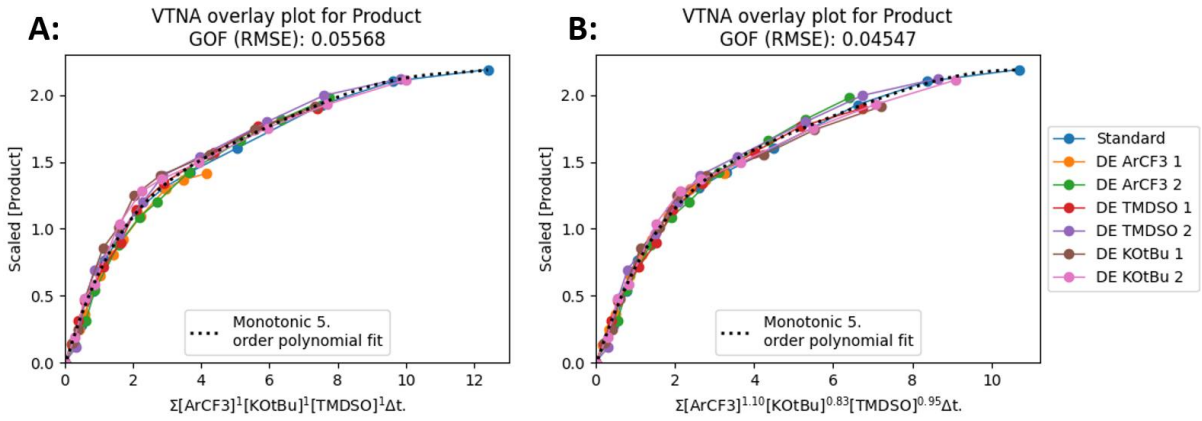
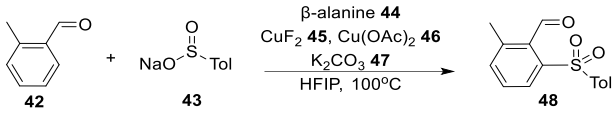


*Figure SA24: Total VTNA overlay plot for the formation of **41** with the reported (A) and calculated (B) order values in $ArCF_3$ **38**, TMDSO **39** and KtOBu **40**. The order values of plot B were calculated via automatic total VTNA.*[9]

A2.11. Entry 11



42: Aldehyde **43:** Sulfinate **44:** TGD
45: CuF_2 **46:** $Cu(OAc)_2$ **47:** K_2CO_3
48: Product

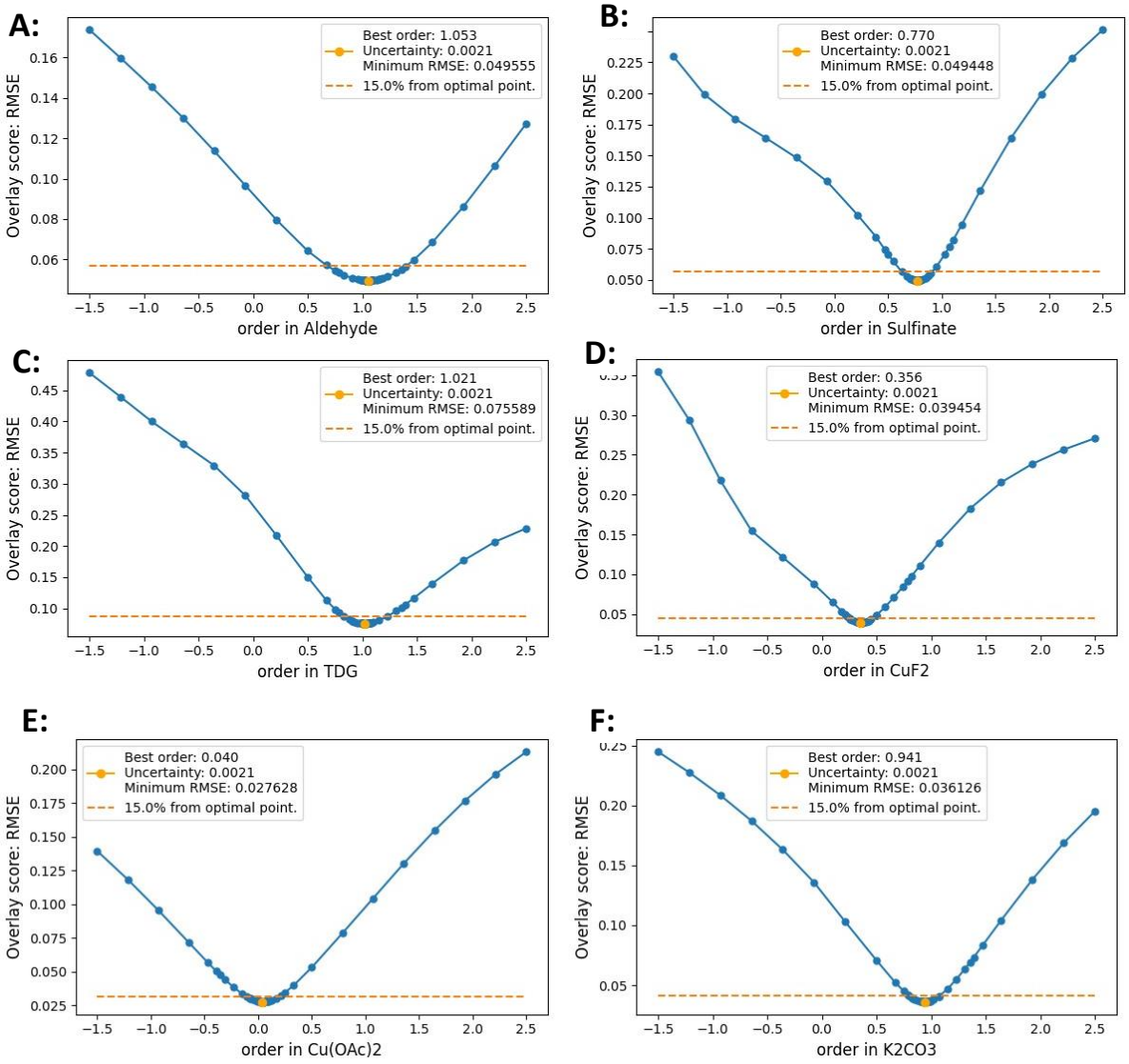


*Figure SA25: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the C-H activation reaction in aldehyde **42** (A), sulfinate **43** (B), transient directing group (TDG) **44** (C), CuF_2 **45** (D), Cu(OAc)2 **46** (E) and K_2CO_3 **47** (F). The product **48** was defined as the output reaction species in the calculations.*[6]

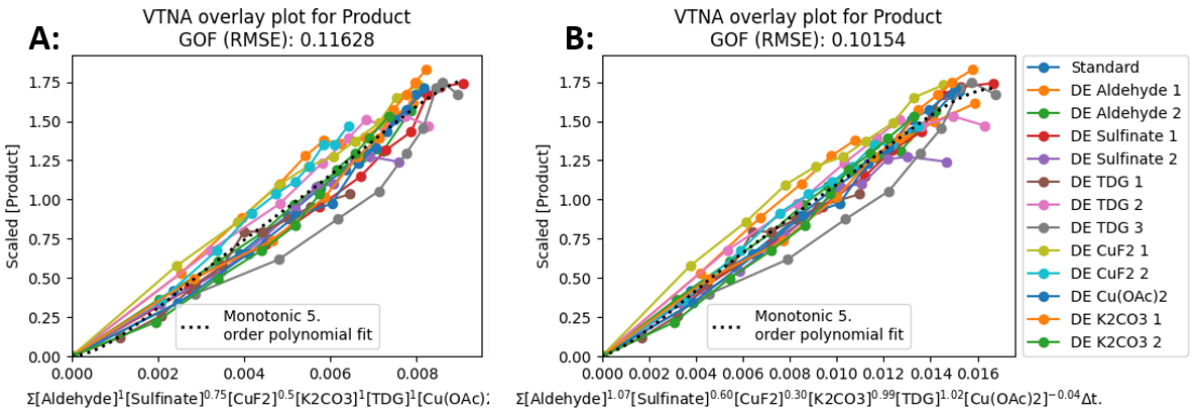
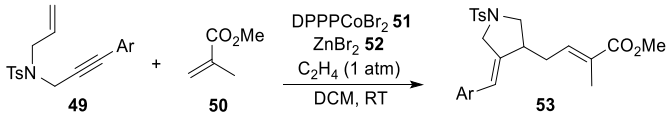


*Figure SA26: Total VTNA overlay plot for the formation of **48** with the reported (A) and calculated (B) order values in aldehyde **42**, sulfinate **43**, transient directing group (TDG) **44**, CuF2 **45**, $Cu(OAc)_2$ **46** and K_2CO_3 **47**. The order values of plot B were calculated via automatic total VTNA.*[6]

A2.12. Entry 12



49: Enyne	**50:** Acrylate	**51:** $DPPPCoBr_2$
52: ZnBr2	**53:** Product	

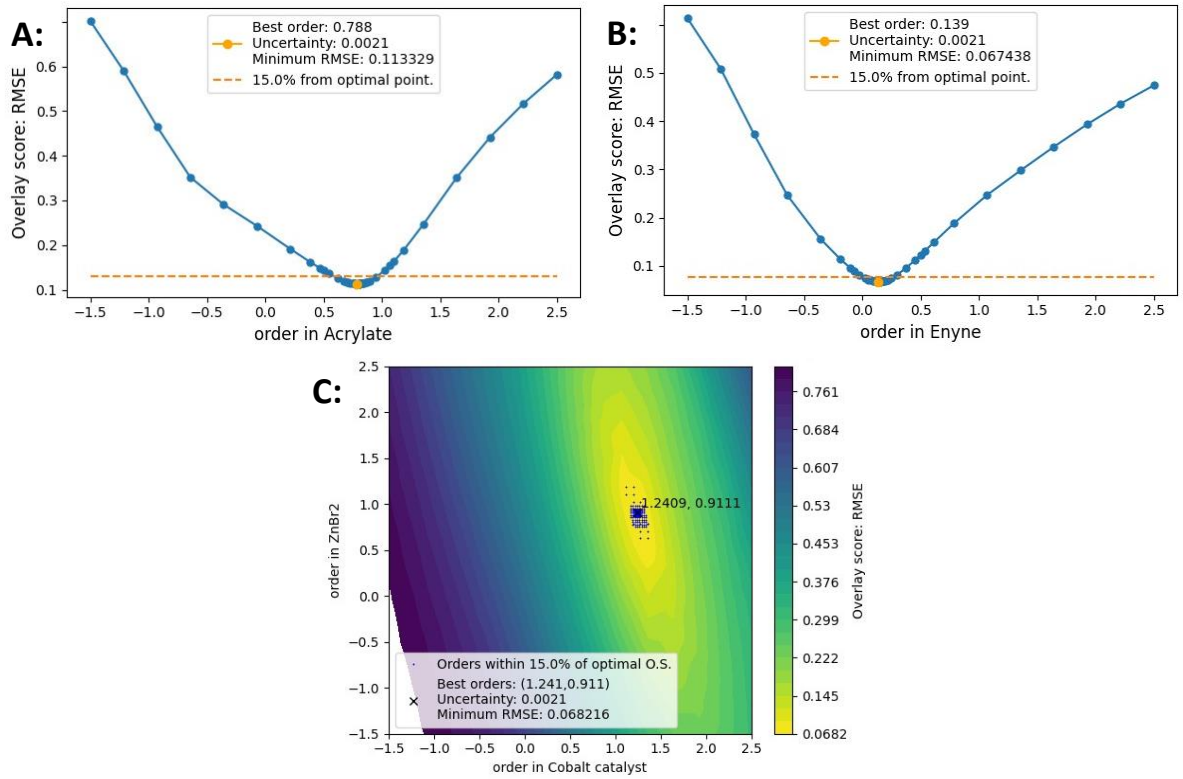


*Figure SA27: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the hydroalkylation reaction in enyne **49** (A), acrylate **50** (B) and in Cobalt catalyst **51** and $ZnBr_2$ **52** (C). The two latter orders were determined in one calculation as the initial concentrations of both were varied between the relevant experiments. Enyne **49** was defined as the output reaction species in the calculations.*[10]

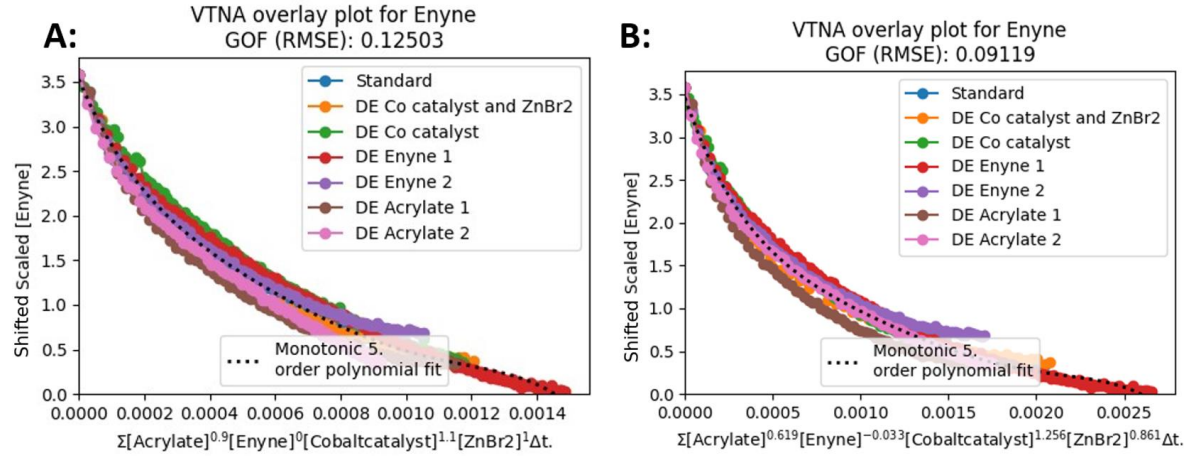
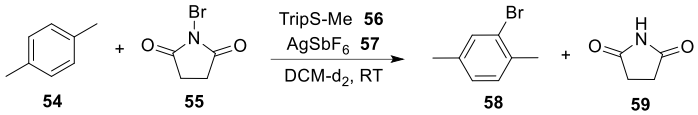


*Figure SA28: Total VTNA overlay plot for the consumption of **49** with the reported (A) and calculated (B) order values in enyne **49**, acrylate **50** and in Cobalt catalyst **51** and $ZnBr_2$ **52**. The order values of plot B were calculated via automatic total VTNA.*[10]

A2.13. Entry 13



54: Xylene **55:** NBS **56:** TripSMe
57: $AgSbF_6$ **58:** Product

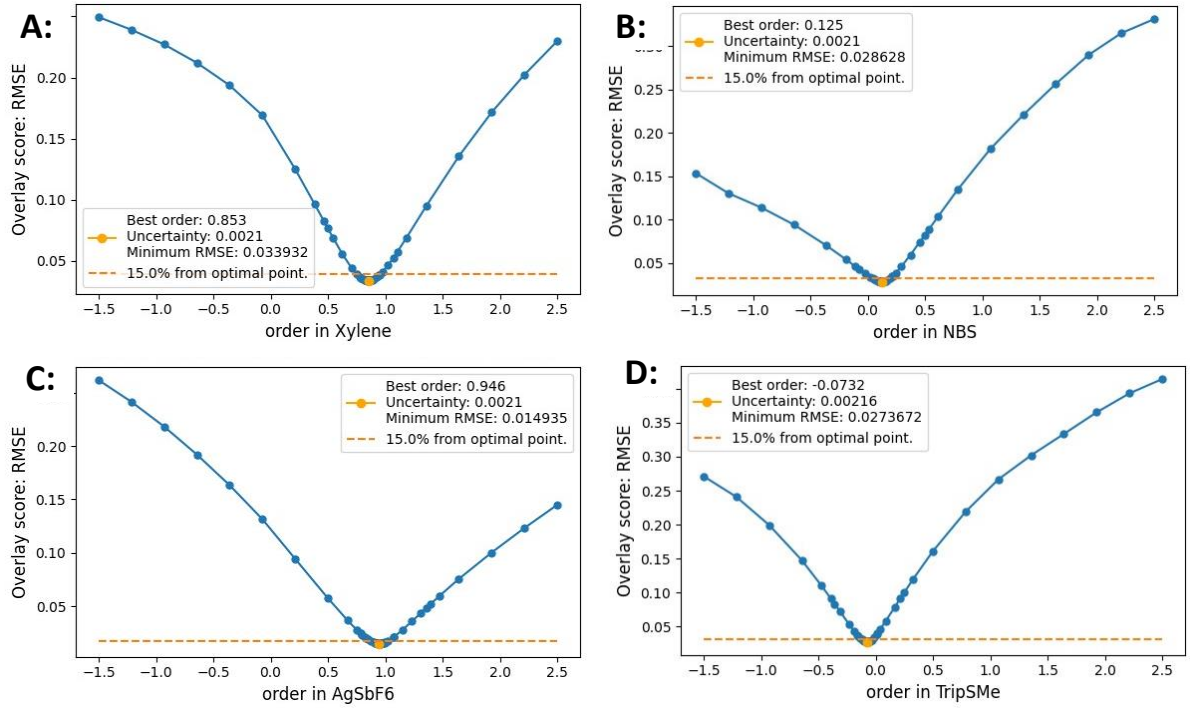


*Figure SA29: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the electrophilic aromatic halogenation reaction in Xylene **54** (A), NBS **55** (B), TripS-Me **56** (C) and $AgSbF_6$ **57** (D). The product **58** was defined as the output reaction species in the calculations.*[17]

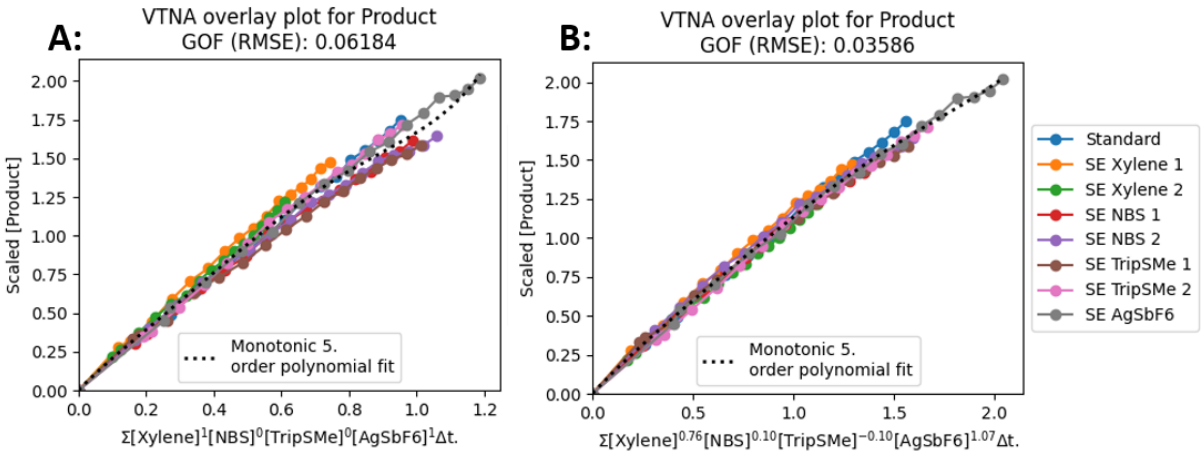
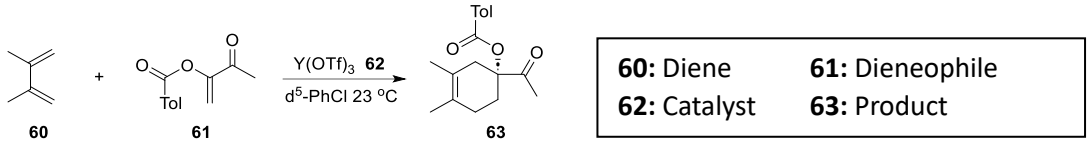


*Figure SA30: Total VTNA overlay plot for the formation of **58** with the reported (A) and calculated (B) order values in in Xylene **54**, NBS **55**, TripS-Me **56** and $AgSbF_6$ **57**. The order values of plot B were calculated via automatic total VTNA.*[17] *The legend to the right of plot B applies to both A and B.*

A2.14. Entry 14



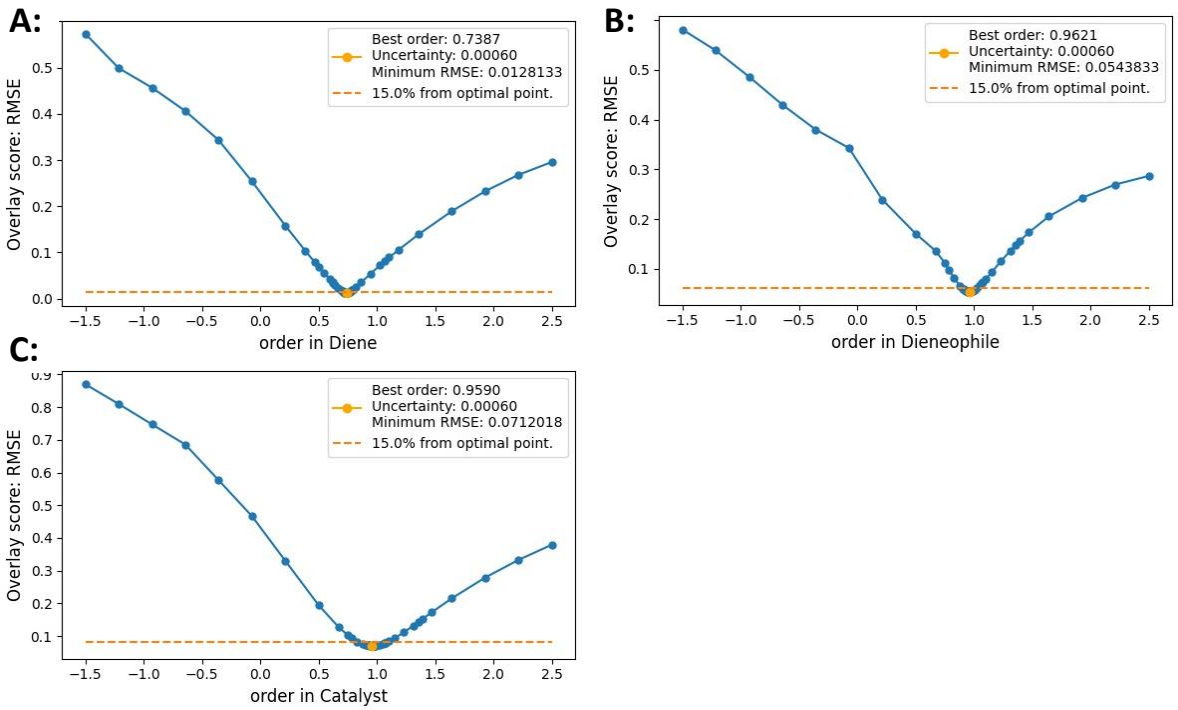


*Figure SA31: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the enantioselective Diels Alder reaction in diene **60** (A), dienophile **61** (B) and catalyst **62** (C). The product **63** was defined as the output reaction species in the calculations.*[18]

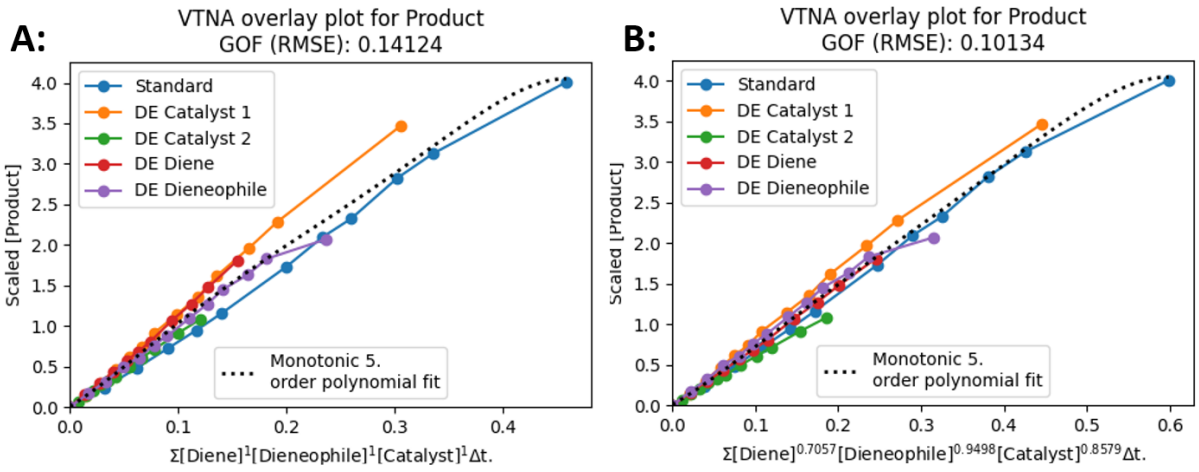


*Figure SA32: Total VTNA overlay plot for the formation of 63 with the reported (A) and calculated (B) order values in diene **60**, dienophile **61** and catalyst **62**. The order values of plot B were calculated via automatic total VTNA.*[18]

A2.15. Entry 15

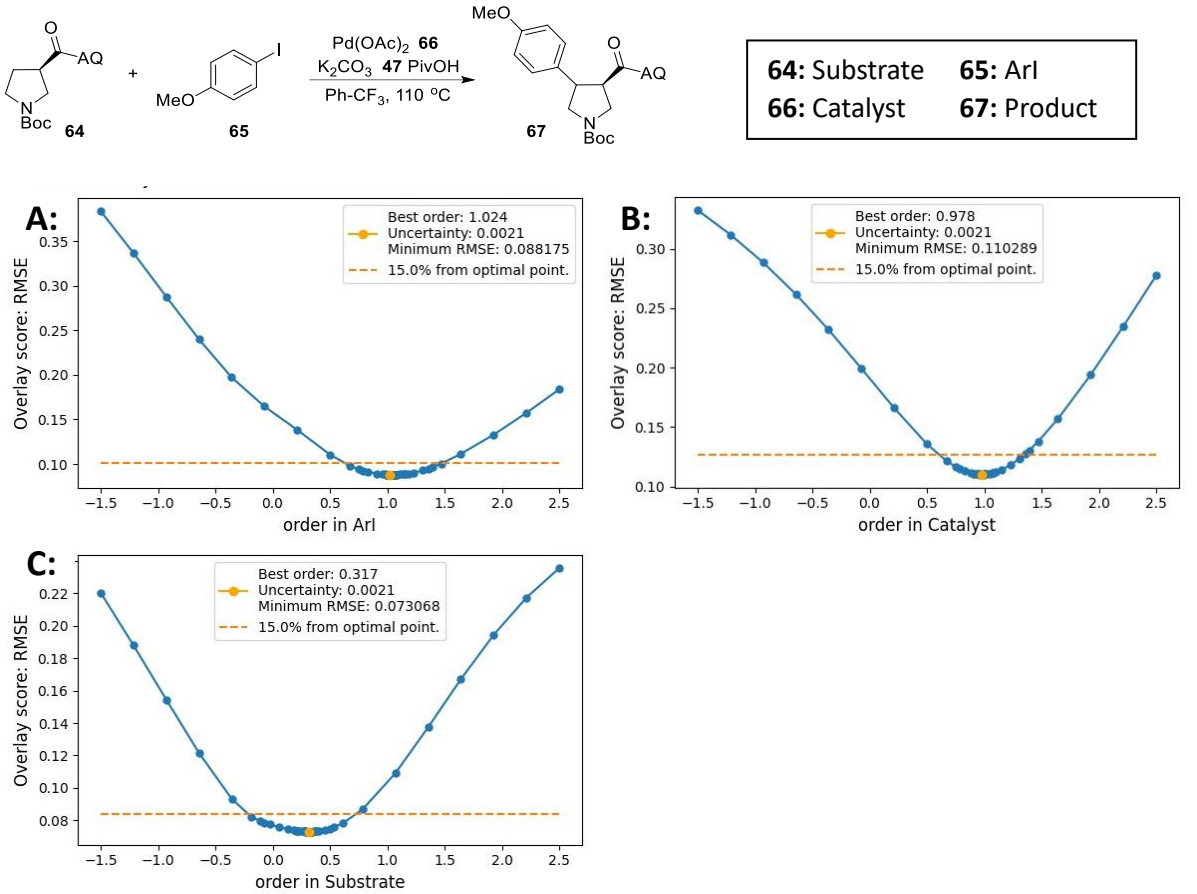


*Figure SA33: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the C-H arylation reaction in substrate **64** (A), ArI **65** (B) and catalyst **66** (C). The product **68** was defined as the output reaction species in the calculations.*[19]

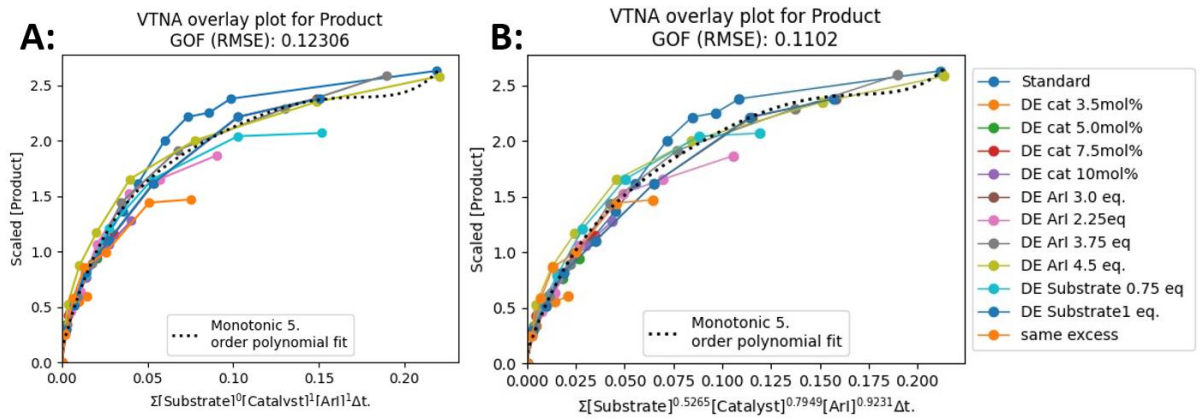
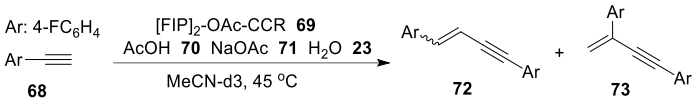


*Figure SA34: Total VTNA overlay plot for the formation of **67** with the reported (A) and calculated (B) order values in substrate **64**, ArI **65** and catalyst **66**. The order values of plot B were calculated via automatic total VTNA.*[19] *The legend to the right of plot B applies to both A and B.*

A2.16. Entry 16



68: Alkyne	**69:** Catalyst	**70:** AcOH
71: NaOAc	**23:** H_2O	

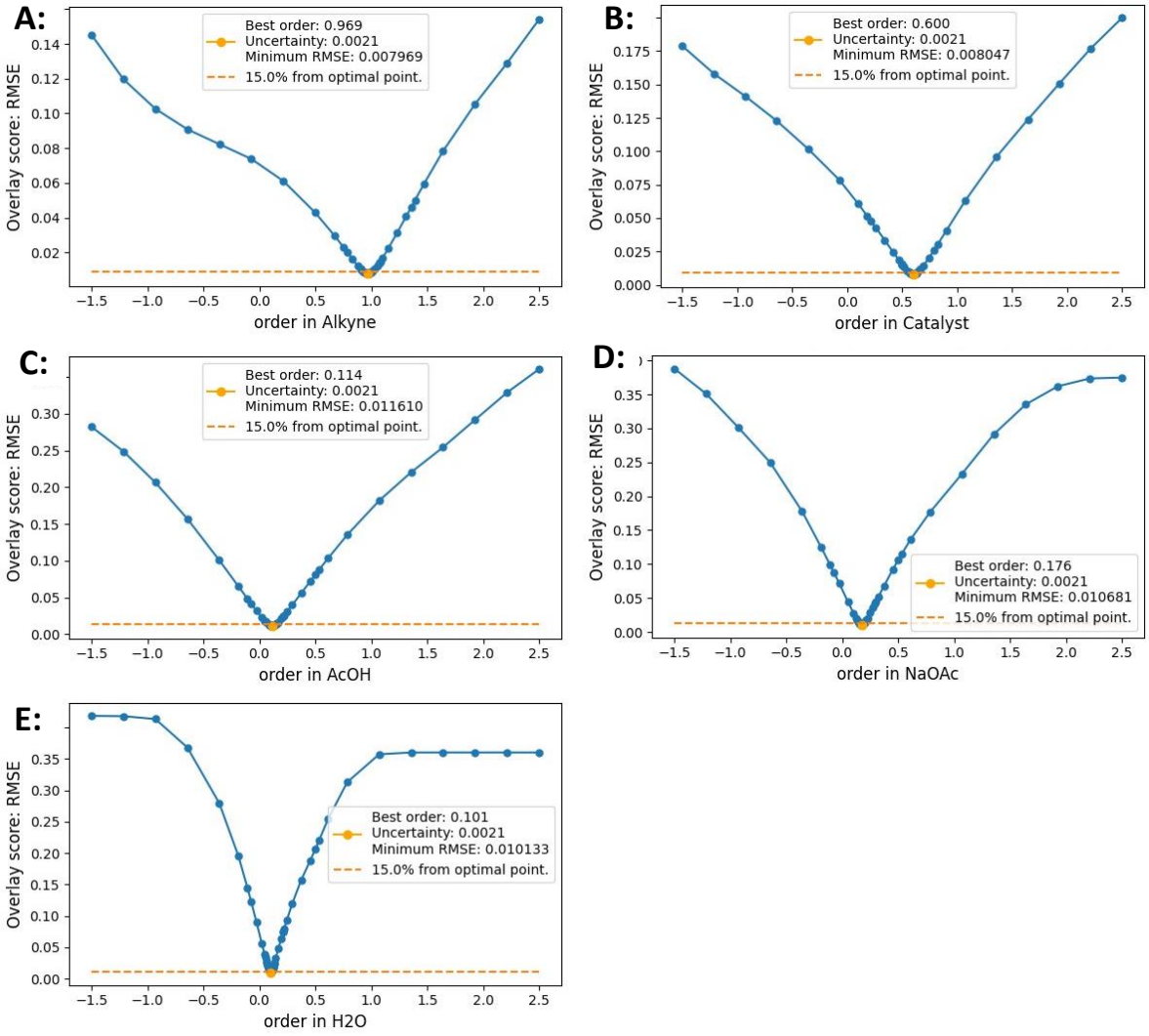


*Figure SA35: Overlay score vs. Order plots for automatic sequential VTNA to obtain the order of the alkyne dimerisation reaction in alkyne **68** (A), catalyst **69** (B), AcOH **70** (C), NaOAc **71** (D) and H_2O **23** (E). Alkyne **68** was defined as the output reaction species in the calculations.*[20]

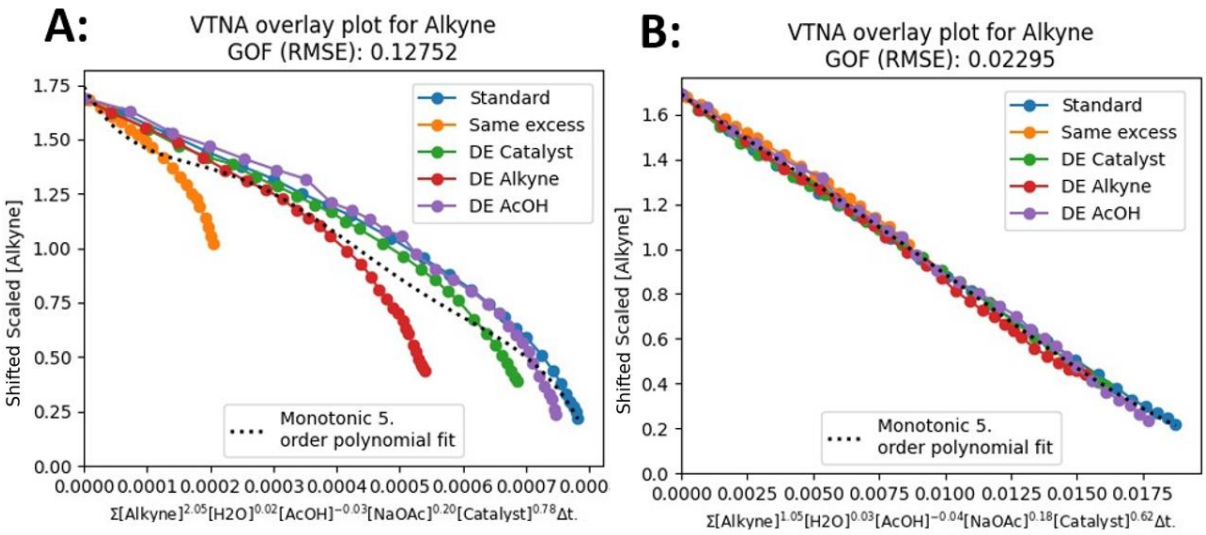


*Figure SA36: Total VTNA overlay plot for the consumption of 68 with the reported (A) and calculated (B) order values in alkyne **68**, catalyst **69**, AcOH **70**, NaOAc **71** and H2O **23.** The order values of plot B were calculated via automatic total VTNA.*[20]

Overlay plot A in Figure SA36 is that corresponding to the reaction order values reported by Peters *et al.*[20] However, the overlay plot shown in the ESI of this publication seems to rather correspond to the same orders in **69**, **70**, **71** and **23** but an order of 1.05 in **68**. The total VTNA overlay plot corresponding to these order values is shown in Figure SA37 and exhibits only a slightly worse RMSE than the calculated order values.

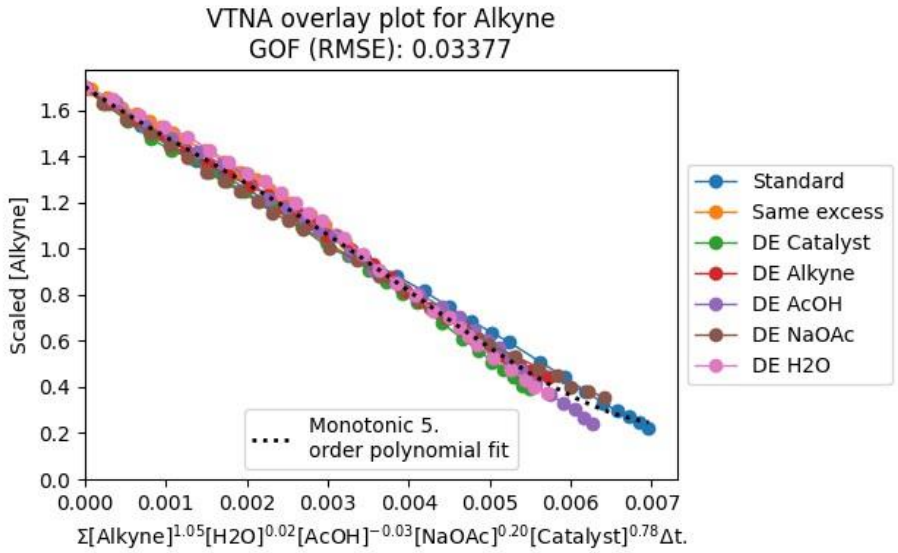


*Figure SA37: Total VTNA overlay plot for the consumption of **68** with the reported order values in **69**, **70**, **71** and **23** but with an order of 1.05 rather than the reported 2.05 in **68**.*[20] *Note that the overlay score is only about 50% higher than that at the calculated order values, but 4 times lower than in Fig. SA36A.*

A3. VTNA overlay plots for orders obtained by Auto-VTNA for complex reaction systems

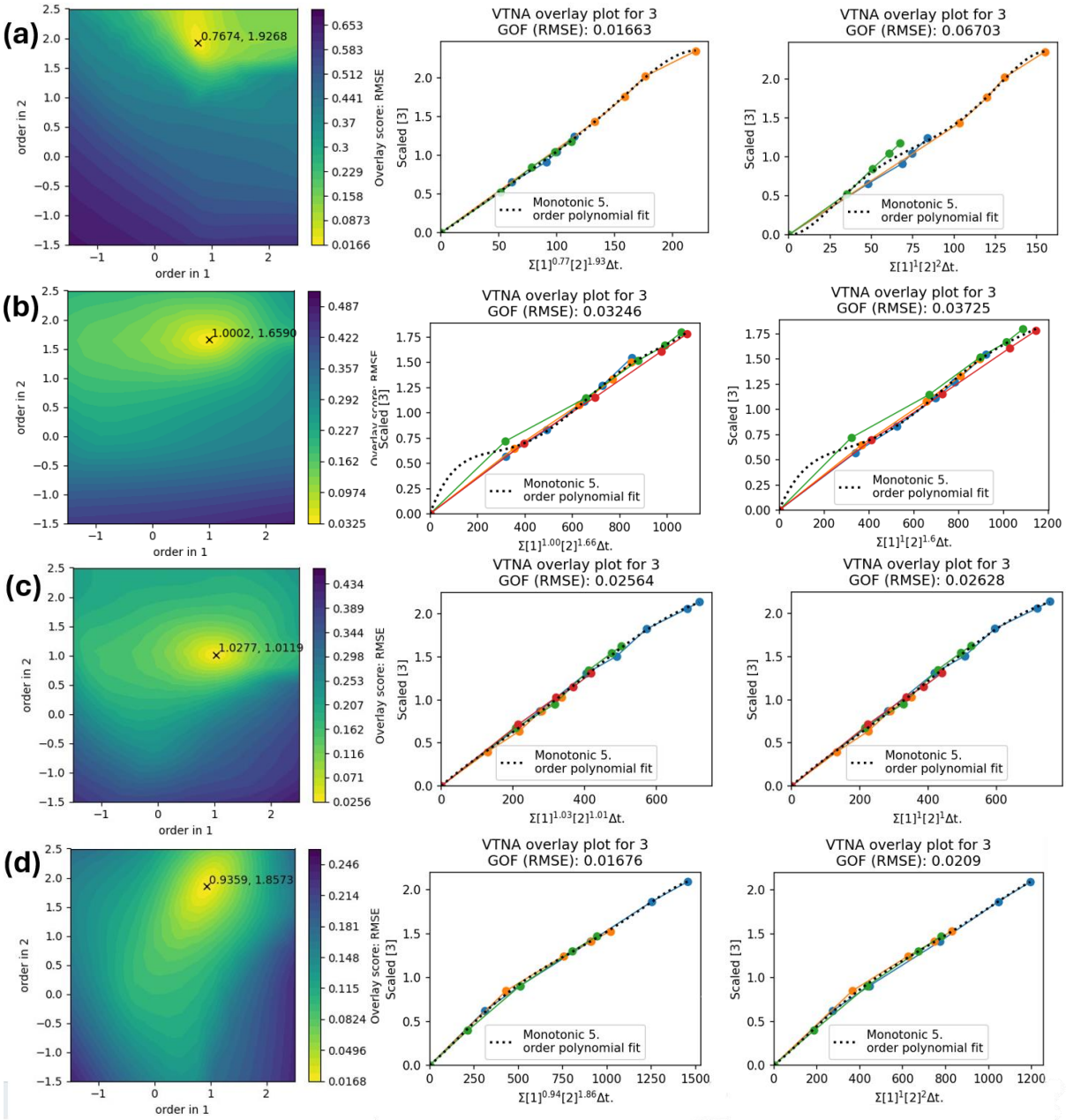


*Figure SA38: Contour plots illustrating the degree of concentration profile overlay across reaction orders between -1.5 and 2.5 in dimethyl itaconate **1** and piperidine **2** for kinetic data in DMSO (a), IPA (b), EtOH (c) and THF (d).[4] The calculated order values and contour plots have been obtained by the standard 5th order monotonic fit with RMSE as the overlay score. The overlay plots corresponding to the calculated and originally reported reaction order values are also shown.*

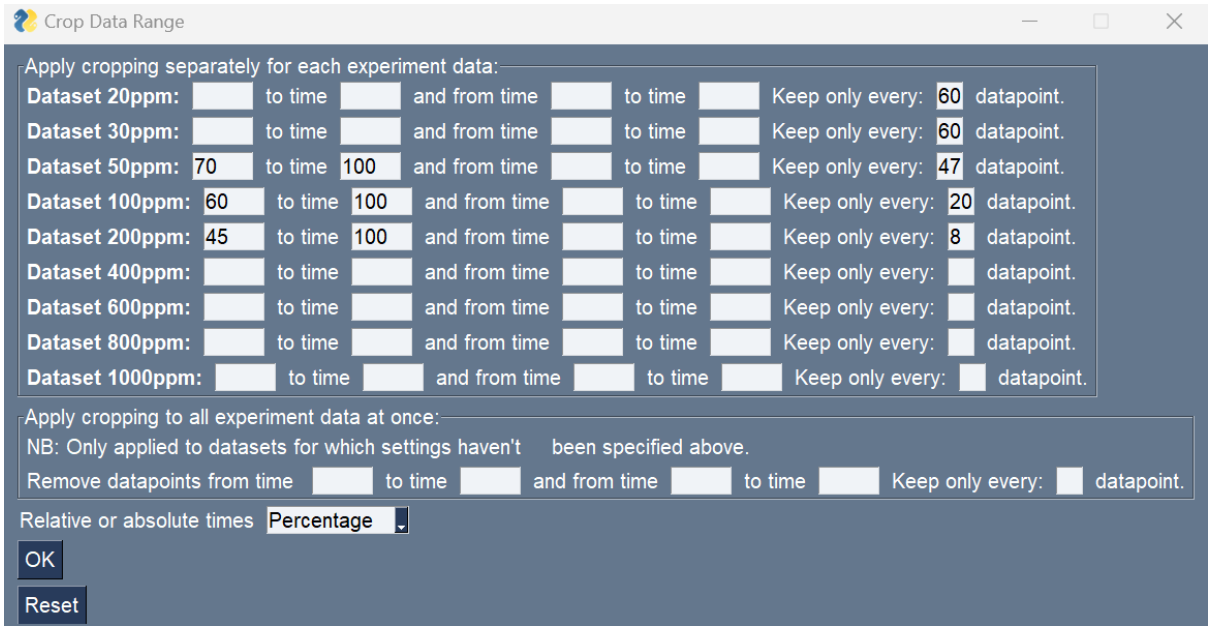


Figure SA39: Kinetic data cropping settings used to perform Auto-VTNA on the kinetic data on the Heck reaction of iodobenzene and methyl acrylate by Newton et al.[5]

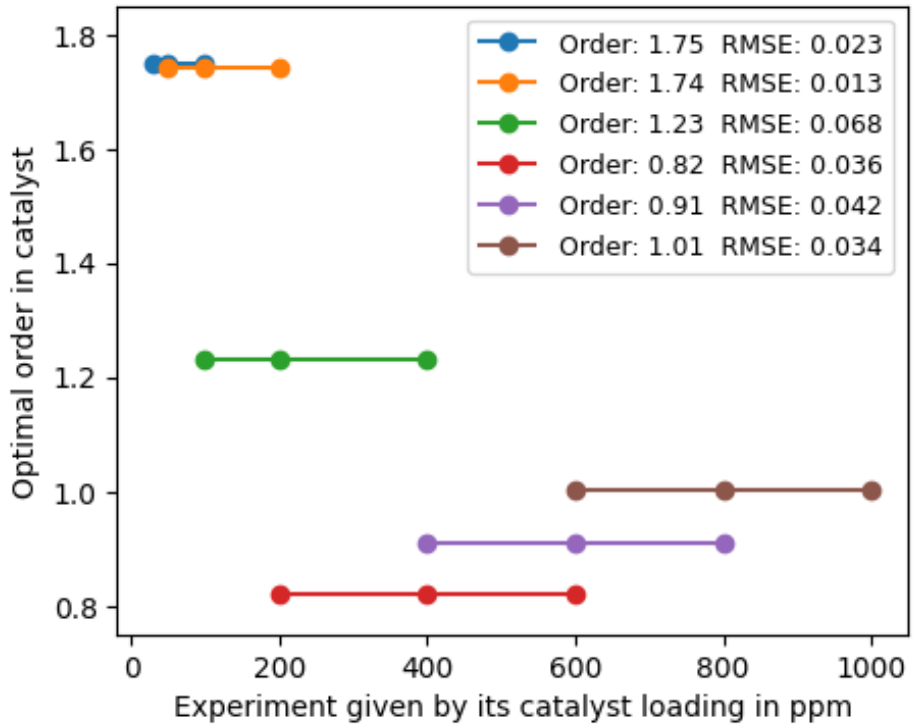


Figure SA40: Calculated reaction orders in catalyst for every 3 consecutive product concentration profiles in terms of catalyst loading.

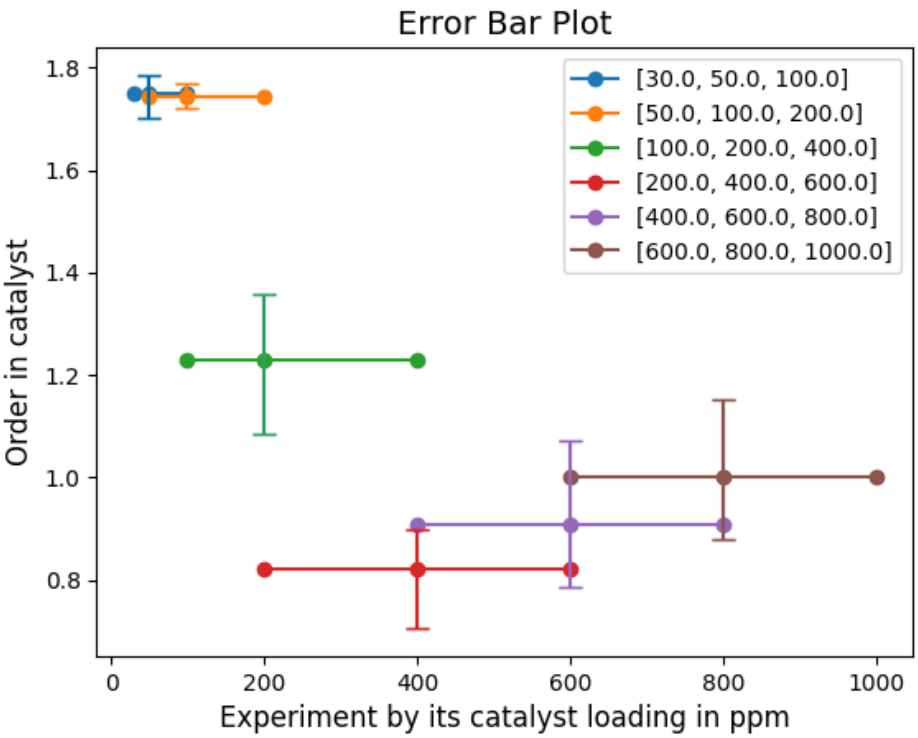


Figure SA41: Calculated reaction orders in catalyst for every 3 consecutive product concentration profiles in terms of catalyst loading with error bars corresponding to order values that give overlay scores within 15% of that of the optimal order value.

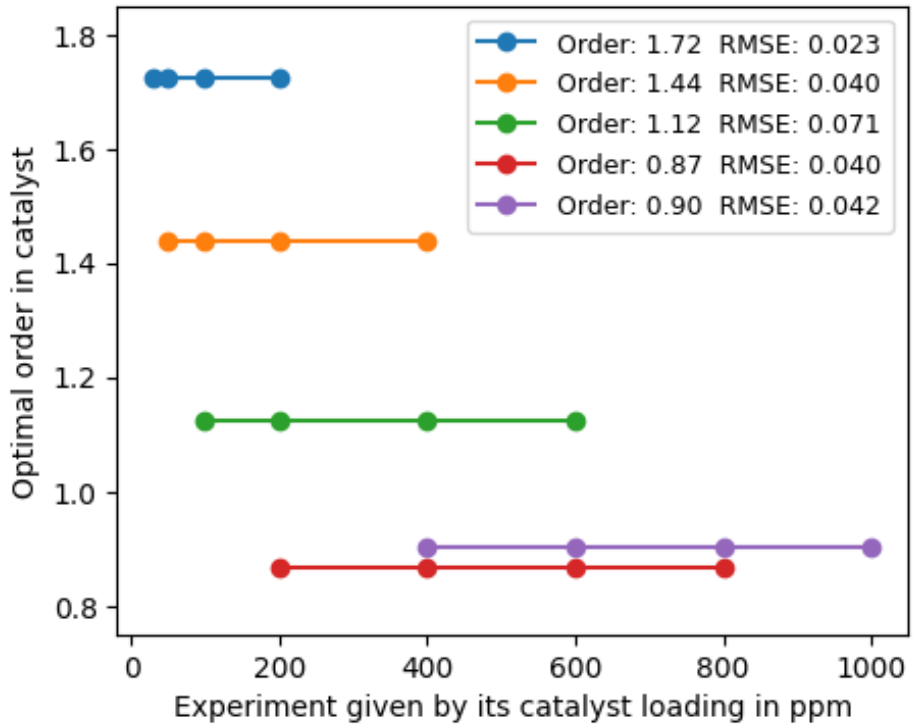


Figure SA42: Calculated reaction orders in catalyst for every 4 consecutive product concentration profiles in terms of catalyst loading

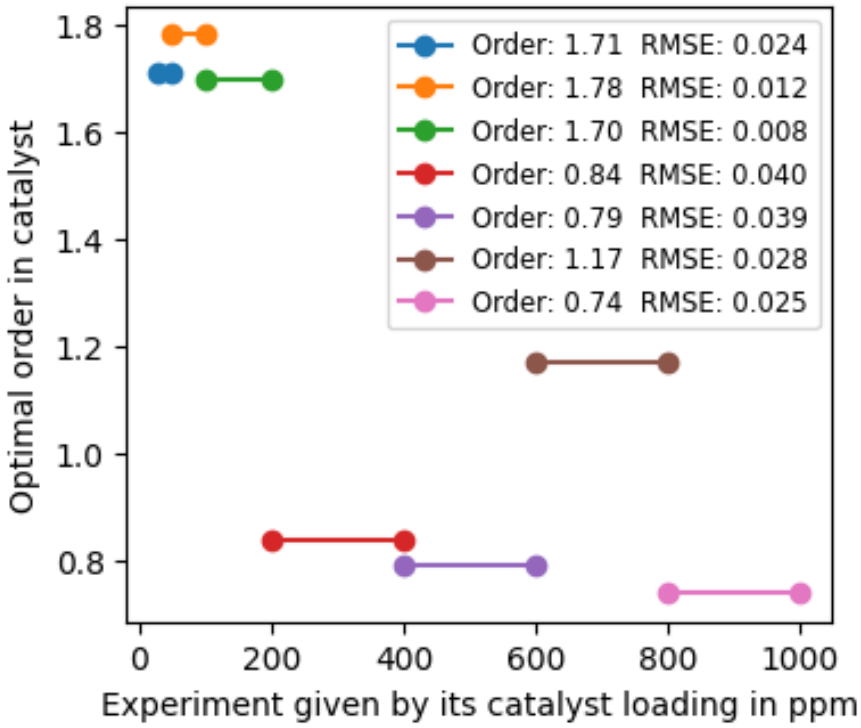


Figure SA43: Calculated reaction orders in catalyst for every 2 consecutive product concentration profiles in terms of catalyst loading.

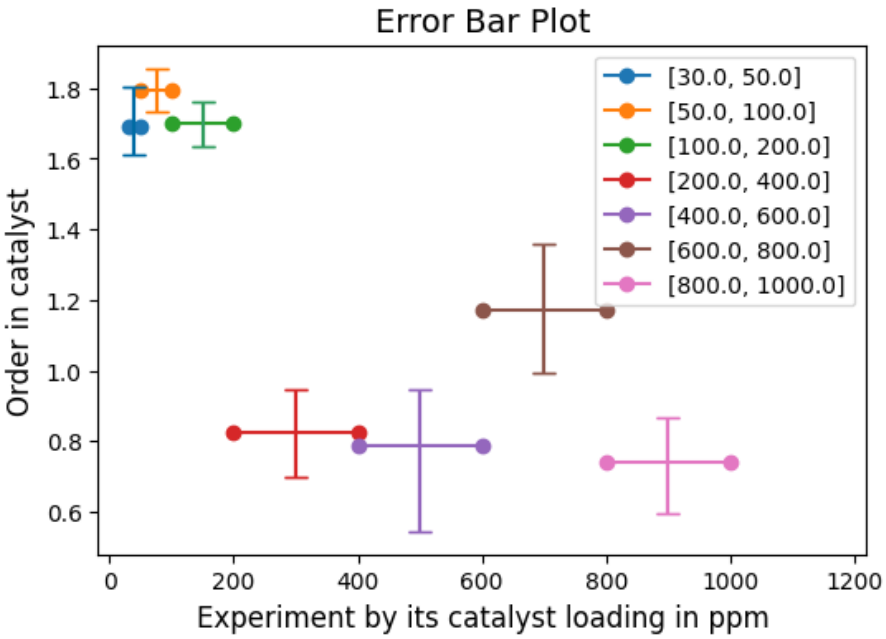


Figure SA44: Calculated reaction orders in catalyst for every 2 consecutive product concentration profiles in terms of catalyst loading with error bars corresponding to order values that give overlay scores within 15% of that of the optimal order value.