

GENESIS: A modular pulse programme generator for NOAH supersequences

Jonathan R. J. Yong,¹ Ēriks Kupče,² Tim D. W. Claridge^{1,*}

¹ *Chemistry Research Laboratory, Department of Chemistry, University of Oxford, Mansfield Road, Oxford, OX1 3TA, U.K.*

² *Bruker UK Ltd., Banner Lane, Coventry, CV4 9GH, U.K.*

* `tim.claridge@chem.ox.ac.uk`

Abstract

(132 words)

NOAH (NMR by Ordered Acquisition using ^1H -detection) supersequences allow multiple 2D NMR datasets to be acquired in greatly reduced experiment durations through the elision of recovery delays. In NOAH experiments, up to five “modules” can be combined, which means that there is a very large number of plausible supersequences (ca. 3800): this renders the traditional method of pulse programme construction by hand wholly inadequate. We introduce here a tool named GENESIS (GENeration of Supersequences In Silico), available via <https://nmr-genesis.co.uk>, which programmatically generates arbitrary NOAH supersequences compatible with Bruker spectrometers. This not only allows users to obtain customised supersequences for specific applications, but also enables us to rapidly and effortlessly disseminate new NOAH modules (e.g. PSYCHE, 2D J) as well as improvements to old modules (e.g. ^{13}C - ^1H HMBC, ^{15}N - ^1H HMQC).

1 Introduction

Sensitivity and speed are the two primary concerns for NMR experimentalists. In the former respect, great advances have been obtained through hardware-based methods such as cryogenic probes and hyperpolarisation. On the other hand, the issue of *speed*, particularly in multidimensional NMR, has proven a fruitful area for NMR pulse sequence and method development. Developments in this area include (but are not limited to) ultrafast NMR,^{1–3} non-uniform sampling (NUS),^{4–6} multiple-FID experiments,^{7–10} and the shortening or elision of recovery delays.^{11–14}

In this paper, we focus on NOAH (NMR by Ordered Acquisition using ^1H -detection) experiments,^{15–22} which fall into the last two categories. NOAH experiments consist of a series of 2D experiments (“modules”), combined into one single “supersequence” which uses only one recovery delay for all modules, as opposed to conventional acquisition which would require one recovery delay per module. This arrangement has a precondition, in that each module must

draw from a different *magnetisation pool*. For example, a ^1H - ^{13}C HSQC module must only excite the ^1H nuclei directly attached to the 1.1%-natural abundance ^{13}C , and should leave all other proton magnetisation (the “bulk magnetisation”) along the equilibrium $+z$ axis. A ^1H - ^1H COSY module (or TOCSY, or NOESY, etc.) can then draw on this bulk magnetisation, with almost no loss in sensitivity and without having to wait for the ^{13}C -bound protons to relax. In general, this partitioning of magnetisation is accomplished using pulse sequence elements such as the zz -filter or the zz isotope-selective pulse (ZIP), which have different net effects on different magnetisation pools.

The combinatorial nature of NOAH experiments means that there are a very large number of conceivable supersequences, ranging from NOAH-2 to NOAH-5 (where the suffix indicates the number of modules). Each module is given a one-letter code, such as ‘B’ for HMBC, ‘S’ for HSQC, ‘M’ for HMQC, ‘C’ for COSY, and so on. Of course, not all combinations are equally effective. In the previous example, if the COSY module were placed first, the ^{13}C -bound proton magnetisation would not survive for use in the HSQC: thus, the HSQC module in a NOAH-2 CS would display severe sensitivity losses, as compared to in a NOAH-2 SC. More subtle considerations are also abound, such as in the SBC sequence,¹⁵ where COSY intensities are modulated by T_2 relaxation and J_{HH} evolution. The alternative BSC arrangement,¹⁶ especially with isotropic “ASAP” mixing applied before the COSY, circumvents this issue and has become the preferred implementation for HMBC/HSQC combinations.¹⁷

Despite this, there are still hundreds of effective supersequences: the original paper alone suggests a figure of 285,¹⁵ and the number of available modules has only grown since then. Constructing every one of these supersequences “by hand” is clearly unreasonable. Consequently, it has not been possible to disseminate an exhaustive series of pulse programmes: we have so far had to limit ourselves to a handful of “typical” supersequences, which may or may not be applicable to users’ needs. Furthermore, each new release of pulse programmes (e.g. in different papers) typically contains small incompatibilities and differences from one another.

To solve this problem, we sought to *programmatically* generate NOAH pulse programmes, in a way which makes new combinations as easy as possible to set up and minimises the chances of human error. We term this approach GENESIS (GENeration of Supersequences In Silico). The implementation of this is a single web page, accessible via <https://nmr-genesis.co.uk>, which can construct virtually any supersequence one might want. The regular user interface is designed to only produce “effective” supersequences: thus, for example, it is not possible to create an SBC supersequence. This is most useful for ordinary users who wish to follow established best practices. For more advanced usage, enabling “developer mode” will remove these limitations, allowing any arbitrary combination of modules to be created.

2 Details

We begin with a brief discussion of how the GENESIS approach works. The pulse programme generation code itself is written in TypeScript (version 4.2.3, Microsoft), which is compiled to JavaScript and then executed directly in a client’s browser; there is no server-side code. The web browser shows a list of modules for users to choose from, using accessible and familiar names such as HMBC, HSQC, etc. Internally, these are mapped to a series of `NOAHModule` objects, each of which contain module-specific information such as its abbreviation (usually one letter, but some are longer), parameter definitions, the pulse programme for the module itself, and the AU programme to use for processing.

Using this information, GENESIS then constructs the pulse programme in three steps:

- the *preamble*, which contains header comments including information about the pulse programme and constituent modules, as well as parameter definitions.
- the *main section*, which contains the actual instructions corresponding to the pulse sequence. Appropriate looping and incrementation of parameters such as phase cycles, t_1 delays, and gradient pulses for echo–antiecho selection is introduced at this stage.
- the *postamble*, which contains the phase cycles themselves, as well as other comments such as parameter descriptions (displayed in TopSpin’s **ased** parameter setup screen), gradient and shaped pulse information (which allow direct setup using the `gppp` and `wvm` commands), and other details such as which modules were used in the pulse programme and the GENESIS version number (which enables full reproducibility in the pulse programme generation).

(Maybe a figure pointing out these three sections?)

The website further contains a series of frequently asked questions and instructions on how to set NOAH experiments up; and also offers download links for the AU scripts used for processing, as well as the Python script used for toggling non-uniform sampling on/off.

2.1 Standardisation of parameter meanings

An immediate problem of directly concatenating pulse programme texts from different modules is that a given parameter in one module may take on a different value in another module. As a necessary step to prevent such clashes, we have fully standardised all parameters used in the GENESIS pulse programmes. These include pulse widths (`p`), delays (`d`), constants (`cnst`), z -gradient amplitudes (`gpz`), and phase cycles (`ph`). Thus, `d2` is always $1/(2 \cdot {}^1J_{\text{CH}})$, and `d4` is always $1/(4 \cdot {}^1J_{\text{CH}})$. Where possible, we have chosen meanings that are similar to those in the standard library of Bruker pulse programmes. However, there are invariably some small deviations: for example, the Δ' delay in the ^{13}C sensitivity-enhanced HSQC (which can range from $1/(8 \cdot {}^1J_{\text{CH}})$ to $1/(4 \cdot {}^1J_{\text{CH}})$) is `d24` in the standard library, but we have chosen `d6` because `d24` is reserved for $1/(4 \cdot {}^1J_{\text{NH}})$.

In place of module-specific delays which are often called `DELTA1`, ... in standard library sequences, we have chosen to define new identifiers with more human-readable names inside the pulse programme itself. Thus, the delays in a HSQC sequence might be called `DC_HSQC1`, ..., where the first `C` indicates the indirect-dimension nucleus (^{13}C).

While this was primarily implemented in order to facilitate pulse programme construction, this standardisation also makes it far easier for users to set up a series of different supersequences, as virtually every parameter value can simply be reused. In principle, it is possible to set up a single “prototype” parameter set which already has every parameter set up; to run a different NOAH experiment, one need only change `PULPROG`, `TD1`, and `NBL`. (Why haven’t we done this? We could make four different parameter sets for NOAH-2 through NOAH-5, and attach them somewhere.)

2.2 Choice of module ‘versions’

Another potential issue is that different versions of a pulse sequence often exist. A simple example is the *zz*-HMBC,^{16,18} where the *zz*-filter element needs to retain ^{13}C -bound and optionally ^{15}N -bound ^1H magnetisation, depending on what modules follow. Since the user only specifies that they want a HMBC module and not the exact details of the *zz*-filter, the GENESIS code must in effect make this choice for the user. The sensitivity-enhanced HSQC (seHSQC, abbreviated ‘S⁺’) presents a more nuanced case. If the seHSQC module is followed by one or more homonuclear modules such as a COSY or NOESY, then the ZIP-seHSQC^{21,22} should be used, as this preserves the requisite magnetisation for the later modules. However, if the seHSQC module is used last in the sequence, then the original Cavanagh–Rance–Kay seHSQC^{23,24} should be used in order to maximise sensitivity. If necessary, this “intelligent” module selection can be circumvented by entering “developer mode”, where the user is free to pick exactly which version of a module they want to include. (Figure X depicts the decision process for the HMBC and seHSQC modules...)

2.3 Recent improvements

A substantial benefit of the GENESIS approach is that it provides a single source for all NOAH pulse sequences and processing scripts, which are always in sync with one another. This allows us to rapidly disseminate new NOAH developments to users, independently from TopSpin’s own release cycle, and without needing to write an entire paper about them. The initial GENESIS release already contains a number of such improvements, some of which we detail below.

2.3.1 ^{15}N HMQC gradient scheme

(Need a figure here showing pulse sequence and before/after spectra)

We have recently described the occurrence of “wing artefacts” in homonuclear modules, which arise from bulk magnetisation which evolves during either half of t_1 in a preceding heteronuclear module.²² These artefacts can be removed in an elegant manner by ensuring that each half of

t_1 in a HMQC/HSQC/seHSQC module contains coherence transfer pathway (CTP) gradients of equal sign and magnitude, which makes sure that any bulk magnetisation undergoing net evolution during t_1 is dephased.

At the same time, it is also important that the final gradient in the heteronuclear module have as large an amplitude as possible, since this gradient is responsible for dephasing bulk magnetisation that is not returned to $+z$ just prior to the detection period. In order to accomplish this, the previous ^{15}N HMQC module used bipolar opposing gradients in either half of t_1 , a scheme which allows the final refocusing CTP gradient g_2 to have an amplitude of $4\gamma_{\text{N}}/\gamma_{\text{H}}$ (Fig XX). Although this leads to excellent artefact suppression in the ^{15}N HMQC itself, wing artefacts are apparent in downstream modules, because the opposing gradients cancel each other out and do not enforce any CTP selection on the bulk magnetisation.

The solution to this is to only use two gradients during t_1 (one in either half), but to *lengthen* their duration such that the final gradient g_2 provides sufficient dephasing in the HMQC itself (Fig XX). This strategy was previously described for the ^{15}N seHSQC;²² we have also applied it to the HMQC with success.

2.3.2 Suppression of one-bond artefacts in HMBC

(Need a figure here showing pulse sequence and before/after spectra)

The zz -HMBC module is ordinarily placed first in a supersequence, where the zz -filter element serves to preserve magnetisation of protons directly coupled to ^{13}C and/or ^{15}N .^{16,18} Specifically, the zz -filter acts as a 90° excitation pulse on uncoupled protons, while leaving coupled protons along $+z$. This is largely accomplished in practice, as evidenced by the fact that the intensities in subsequent HSQC-type modules are barely perturbed. However, not all of the coupled magnetisation is perfectly retained: in particular, the zz -filter also generates a degree of *antiphase* magnetisation of the form $2\text{H}_x\text{C}_z$ due to $^1J_{\text{CH}}$ mismatch ... ?. This antiphase magnetisation is later refocused during the low-pass J-filter (LPJF) to give in-phase magnetisation, eventually ending up as one-bond correlation artefacts in the HMBC spectrum. In fact, the degree of $^1J_{\text{CH}}$ artefacts present in the NOAH HMBC was typically greater than those in a standard HMBC (Fig xx).

A simple solution to this is to add a ^{13}C 90° pulse at the end of the zz -filter: this converts any antiphase magnetisation to double- or zero-quantum magnetisation, which is later destroyed by the LPJF. This idea has previously been used by Luy and coworkers in CLIP-HSQC experiments to remove antiphase contributions prior to FID detection.²⁵ In the event, this small modification proved to have a large impact, almost completely suppressing the $^1J_{\text{CH}}$ artefacts; the NOAH zz -HMBC now even outperforms the standard third-order LPJF HMBC in this aspect (Fig xx).

2.3.3 PSYCHE and 2DJ modules

In previous work,²² we described how the sensitivity of some ^{15}N modules, particularly the HMQC, could be improved through the use of k -scaling.²⁶ This entails a reduction in the number of t_1 increments (by a factor of k), in return for a corresponding increase in the number of transients per increment, with no overall change in the experimental time.

A simple extension of this protocol to *homonuclear* ^1H – ^1H modules enables 2D J-resolved spectroscopy, or pseudo-2D pure shift techniques, to be incorporated into NOAH supersequences. In both cases, the number of t_1 increments needed (16–32) is far smaller than the typical number required for a 2D experiment (128–256). In particular, at present, we have implemented a family of PSYCHE experiments: namely the original pseudo-2D PSYCHE, the triple spin echo (TSE)-PSYCHE experiment which provides improved robustness towards strong coupling, and the PSYCHE 2DJ experiment.^{27–29}

In PSYCHE spectra, the flip angle of the chirp or saltire pulses used in the J-refocusing element provides the experimentalist with a choice to make: a larger flip angle provides greater sensitivity, but at the cost of increased artefacts.²⁹ One advantage of acquiring PSYCHE spectra in NOAH-type supersequences is that, due to the increased number of transients, sensitivity is often not at a premium. Thus, the user can choose a smaller flip angle (ca. 10°) in order to maximise spectral purity instead, while not losing any actual spectrometer *time* (since multiple spectra are being acquired simultaneously).

(Might be nice to show an example of a NOAH-4 experiment incl. PSYCHE module...)

2.3.4 splitx.au processing

NOAH data processing is done using the `splitx.au` AU programme; this is responsible for creating NBL separate datasets containing the data for each module, and processing each dataset using module-specific AU programmes (e.g. `noah_hsqc` for ^{13}C HSQC data). Previously, the module-specific AU programmes had to be specified as the `USERP1`, `USERP2`, ... series of processing parameters. Since the GENESIS approach allows module-specific information such as these to be embedded within the pulse programme itself, this means that (with a small modification to the `splitx.au` AU programme) we can parse the pulse programme to obtain the requisite list of AU programmes. The user therefore does not have to specify it explicitly, which makes setting up multiple different supersequences a much smoother process. It is still possible to specify the `USERPx` parameters, and these will override the AU programmes parsed from the pulse programme, but in practice there is no real reason to do so.

2.3.5 Non-uniform sampling implementation

With the GENESIS pulse programmes we also introduce a new and more user-friendly implementation of non-uniform sampling (NUS). NOAH experiments do not work “out of the box” with TopSpin’s conventional NUS setup routine: some special adjustments have to be made by

manually generating the list of increments to be sampled and adjusting the t_1 delays accordingly. Previously, this was accomplished using a Python script which created a new pulse programme for each supersequence, e.g. `noah3_BSC.nus`.¹⁷ We have modified this approach such that the *same* pulse programme can be used for both uniform and non-uniform sampling, where NUS is controlled by an acquisition flag `-DNUS`. Although a (different) Python script is still required for initialisation, this means that it is no longer necessary to keep two separate instances of the same pulse sequence in TopSpin, and reduces the difficulty of turning NUS on and off.

3 Conclusion

To be written.

References

- (1) Frydman, L.; Scherf, T.; Lupulescu, A. *Proc. Natl. Acad. Sci. U. S. A.* **2002**, *99*, 15858–15862, DOI: [10.1073/pnas.252644399](https://doi.org/10.1073/pnas.252644399).
- (2) Frydman, L.; Lupulescu, A.; Scherf, T. *J. Am. Chem. Soc.* **2003**, *125*, 9204–9217, DOI: [10.1021/ja030055b](https://doi.org/10.1021/ja030055b).
- (3) Tal, A.; Frydman, L. *Prog. Nucl. Magn. Reson. Spectrosc.* **2010**, *57*, 241–292, DOI: [10.1016/j.pnmrs.2010.04.001](https://doi.org/10.1016/j.pnmrs.2010.04.001).
- (4) Mobli, M.; Hoch, J. C. *Prog. Nucl. Magn. Reson. Spectrosc.* **2014**, *83*, 21–41, DOI: [10.1016/j.pnmrs.2014.09.002](https://doi.org/10.1016/j.pnmrs.2014.09.002).
- (5) Kazimierczuk, K.; Orekhov, V. *Magn. Reson. Chem.* **2015**, *53*, 921–926, DOI: [10.1002/mrc.4284](https://doi.org/10.1002/mrc.4284).
- (6) Gołowicz, D.; Kasprzak, P.; Orekhov, V.; Kazimierczuk, K. *Prog. Nucl. Magn. Reson. Spectrosc.* **2020**, *116*, 40–55, DOI: [10.1016/j.pnmrs.2019.09.003](https://doi.org/10.1016/j.pnmrs.2019.09.003).
- (7) Nolis, P.; Pérez-Trujillo, M.; Parella, T. *Angew. Chem. Int. Ed.* **2007**, *46*, 7495–7497, DOI: [10.1002/anie.200702258](https://doi.org/10.1002/anie.200702258).
- (8) Kupče, Ě.; Freeman, R. *J. Am. Chem. Soc.* **2008**, *130*, 10788–10792, DOI: [10.1021/ja8036492](https://doi.org/10.1021/ja8036492).
- (9) Nagy, T. M.; Gyöngyösi, T.; Kövér, K. E.; Sørensen, O. W. *Chem. Commun.* **2019**, *55*, 12208–12211, DOI: [10.1039/c9cc06253j](https://doi.org/10.1039/c9cc06253j).
- (10) Nagy, T. M.; Kövér, K. E.; Sørensen, O. W. *J. Magn. Reson.* **2020**, *316*, 106767, DOI: [10.1016/j.jmr.2020.106767](https://doi.org/10.1016/j.jmr.2020.106767).
- (11) Schanda, P.; Van Melckebeke, H.; Brutscher, B. *J. Am. Chem. Soc.* **2006**, *128*, 9042–9043, DOI: [10.1021/ja062025p](https://doi.org/10.1021/ja062025p).
- (12) Kupče, Ě.; Freeman, R. *Magn. Reson. Chem.* **2007**, *45*, 2–4, DOI: [10.1002/mrc.1931](https://doi.org/10.1002/mrc.1931).
- (13) Schulze-Sünninghausen, D.; Becker, J.; Luy, B. *J. Am. Chem. Soc.* **2014**, *136*, 1242–1245, DOI: [10.1021/ja411588d](https://doi.org/10.1021/ja411588d).
- (14) Nagy, T. M.; Kövér, K. E.; Sørensen, O. W. *Angew. Chem. Int. Ed.* **2021**, *60*, 13587–13590, DOI: [10.1002/anie.202102487](https://doi.org/10.1002/anie.202102487).

- (15) Kupče, Ě.; Claridge, T. D. W. *Angew. Chem. Int. Ed.* **2017**, *56*, 11779–11783, DOI: [10.1002/anie.201705506](https://doi.org/10.1002/anie.201705506).
- (16) Kupče, Ě.; Claridge, T. D. W. *Chem. Commun.* **2018**, *54*, 7139–7142, DOI: [10.1039/c8cc03296c](https://doi.org/10.1039/c8cc03296c).
- (17) Claridge, T. D. W.; Mayzel, M.; Kupče, Ě. *Magn. Reson. Chem.* **2019**, *57*, 946–952, DOI: [10.1002/mrc.4887](https://doi.org/10.1002/mrc.4887).
- (18) Kupče, Ě.; Claridge, T. D. W. *J. Magn. Reson.* **2019**, *307*, 106568, DOI: [10.1016/j.jmr.2019.106568](https://doi.org/10.1016/j.jmr.2019.106568).
- (19) Kupče, Ě.; Mote, K. R.; Webb, A.; Madhu, P. K.; Claridge, T. D. W. *Prog. Nucl. Magn. Reson. Spectrosc.* **2021**, *124–125*, 1–56, DOI: [10.1016/j.pnmrs.2021.03.001](https://doi.org/10.1016/j.pnmrs.2021.03.001).
- (20) Kupče, Ě.; Frydman, L.; Webb, A. G.; Yong, J. R. J.; Claridge, T. D. W. *Nat. Rev. Methods Primers* **2021**, *1*, No. 27, DOI: [10.1038/s43586-021-00024-3](https://doi.org/10.1038/s43586-021-00024-3).
- (21) Hansen, A. L.; Kupče, Ě.; Li, D.-W.; Bruschweiler-Li, L.; Wang, C.; Brüschweiler, R. *Anal. Chem.* **2021**, *93*, 6112–6119, DOI: [10.1021/acs.analchem.0c05205](https://doi.org/10.1021/acs.analchem.0c05205).
- (22) Yong, J. R. J.; Hansen, A. L.; Kupče, Ě.; Claridge, T. D. W. *J. Magn. Reson.* **2021**, *329*, 107027, DOI: [10.1016/j.jmr.2021.107027](https://doi.org/10.1016/j.jmr.2021.107027).
- (23) Palmer III, A. G.; Cavanagh, J.; Wright, P. E.; Rance, M. *J. Magn. Reson.* **1991**, *93*, 151–170, DOI: [10.1016/0022-2364\(91\)90036-s](https://doi.org/10.1016/0022-2364(91)90036-s).
- (24) Kay E., L.; Keifer, P.; Saarinen, T. *J. Am. Chem. Soc.* **1992**, *114*, 10663–10665, DOI: [10.1021/ja00052a088](https://doi.org/10.1021/ja00052a088).
- (25) Enthart, A.; Freudenberger, J. C.; Furrer, J.; Kessler, H.; Luy, B. *J. Magn. Reson.* **2008**, *192*, 314–322, DOI: [10.1016/j.jmr.2008.03.009](https://doi.org/10.1016/j.jmr.2008.03.009).
- (26) Pérez-Trujillo, M.; Nolis, P.; Bermel, W.; Parella, T. *Magn. Reson. Chem.* **2007**, *45*, 325–329, DOI: [10.1002/mrc.1973](https://doi.org/10.1002/mrc.1973).
- (27) Foroozandeh, M.; Adams, R. W.; Meharry, N. J.; Jeannerat, D.; Nilsson, M.; Morris, G. A. *Angew. Chem. Int. Ed.* **2014**, *53*, 6990–6992, DOI: [10.1002/anie.201404111](https://doi.org/10.1002/anie.201404111).
- (28) Foroozandeh, M.; Adams, R. W.; Kiraly, P.; Nilsson, M.; Morris, G. A. *Chem. Commun.* **2015**, *51*, 15410–15413, DOI: [10.1039/c5cc06293d](https://doi.org/10.1039/c5cc06293d).
- (29) Foroozandeh, M.; Morris, G. A.; Nilsson, M. *Chem. Eur. J.* **2018**, *24*, 13988–14000, DOI: [10.1002/chem.201800524](https://doi.org/10.1002/chem.201800524).

4 SI

4.1 Number of combinations

In this section we count the total number of “plausible” combinations, available from the non-developer mode version of the GENESIS website, using the inclusion–exclusion principle. As of version **CURRENTVERSION** of the website, there are five categories of modules:

- HMBC (2 choices, including “none”)
- ^{15}N – ^1H (3 choices, including “none”)
- ^{13}C – ^1H #1 (5 choices, including “none”)
- ^{13}C – ^1H #2 (8 choices, including “none”)
- ^1H – ^1H (19 choices, including “none”)

To a first approximation, there are therefore $2 \cdot 3 \cdot 5 \cdot 8 \cdot 19 = \mathbf{4560}$ combinations (since we included the “none” options, this figure includes “lesser” supersequences such as NOAH-4 and lower.). However, this does contain some invalid combinations, namely:

1. All five “none” modules selected (**1**)
2. “NOAH-1” combinations with only one module: $1 + 2 + 4 + 7 + 12 = \mathbf{26}$. These are technically valid experiments (and the pulse programmes generated by the website *will* function correctly), but they are no different from standard 2D experiments so the NOAH description does not truly apply to these.
3. “NOAH-6” combinations where there is one experiment in each of the first four categories, and a “double” experiment (e.g. COSY + NOESY) in the last category: $1 \cdot 2 \cdot 4 \cdot 7 \cdot 5 = \mathbf{280}$. These are perfectly sound from a scientific perspective, but cannot be executed in current versions of TopSpin as the parameter NBL (number of modules) has a maximum value of 5.
4. Duplicate combinations where the same ^{13}C – ^1H module (e.g. HSQC) was selected either from the third or the fourth column. There are four such modules, so we must subtract $2 \cdot 3 \cdot 4 \cdot 19 = \mathbf{456}$ combinations.
 - Note, however, that some of these duplicate combinations were *already* rejected in step (2) above. Thus, we need to add back **4** “NOAH-1” combinations that we double-counted.

This yields a total number of $\mathbf{4560} - \mathbf{1} - \mathbf{26} - \mathbf{280} - \mathbf{456} + \mathbf{4} = \mathbf{3801}$ NOAH supersequences. Note that this does not take into account options that are set using acquisition flags, such as multiplicity editing in HSQC-type experiments and inversion of “indirect” responses in HSQC-COSY and HSQC-TOCSY spectra.

Of course, in “developer mode”, this number is far larger: it is on the order of $(7 + 5 + 7 + 21 +$

$23)^5 = 992\,436\,543$, though this is a slight overestimate for similar reasons as above. However, the vast majority of these sequences are either not optimal (e.g. ^1H – ^1H module not placed at the end of the supersequence, or same magnetisation pool being drawn on too many times) or simply pointless (same module repeated multiple times in a supersequence).