

SolidSpin

固体 NMR シミュレーションプラグイン

内容

| | | |
|-----|---|----|
| 1 | はじめに | 3 |
| 2 | インストールとセットアップ | 4 |
| 2.1 | プラグインマネージャでのプラグイン有効化 | 4 |
| 2.2 | プラグインの設定 | 5 |
| 3 | 静的 CSA シミュレーションとフィッティング | 8 |
| 3.1 | 簡単なシミュレーション | 8 |
| 3.2 | 実験スペクトルに基づくシミュレーション | 9 |
| 3.3 | 実験スペクトルのフィッティング。 | 11 |
| 4 | CSA のシミュレーションとフィッティング | 15 |
| 5 | スピン N/2 四極子スペクトルの二次四極子スペクトル幅のフィッティング。 | 18 |
| 6 | 入力ファイルの書式 | 21 |
| 6.1 | シミュレーションタイトル・パラメータの定義 | 23 |
| 6.2 | spinsys, par, proc pulseq セクション | 25 |
| 6.3 | proc main セクション | 26 |

このドキュメントは、JASON 固体 NMR シミュレーションプラグインの概要を説明し、ソフトウェアの使い方を理解していただくことを目的としています。ユーザーは、NMR 処理と JASON に関する実用的な知識を持っていることを前提としています。JASON は Windows オペレーティングシステムで動作します。

1 はじめに

SolidSpin は、広く使われている固体 NMR シミュレーションプログラム SIMPSON と JASON を使いやすいグラフィカルユーザーインターフェースを介して連携するプラグインです。SIMPSON についてはこちらの論文をご参照ください: <https://doi.org/10.1006/jmre.2000.2179>

SIMPSON はコマンドラインユーティリティです。シミュレーション内容は Tcl 言語を使い記述します。固体 NMR を含む様々な NMR 測定を量子力学レベルで数値シミュレーションできます。様々な測定方法をリーズナブルな計算コストでシミュレート可能です。

SolidSpin プラグインでは、プラグイン内から操作するシミュレーションパラメータを定義するセクションや、JASON と SIMPSON 間でパラメータを交換するコードが含まれる、拡張された SIMPSON の入力ファイルを使用します。

入力ファイルのいくつかの例がプラグインに含まれており、さらに独自の入力ファイルを作成、使用することで幅広い解析に利用可能です。

厳密で一般的な量子力学シミュレーションは、単純化された、しばしば解析的に定義された解を持つような特定のケースでは、最適な解ではないかもしれません。しかし、このアプローチは、固体 NMR 解析において、妥当な性能を維持しながら、高いレベルの柔軟性とパワーを提供します。

このマニュアルでは、プラグインの設定方法と、2 つの主要なタスクに対するプラグインの使用方法を説明します：

1. 1次元固体 NMR スペクトルのシミュレーション
2. 実測の 1次元固体 NMR スペクトルとシミュレーションスペクトルのフィッティングによる NMR パラメータ推定

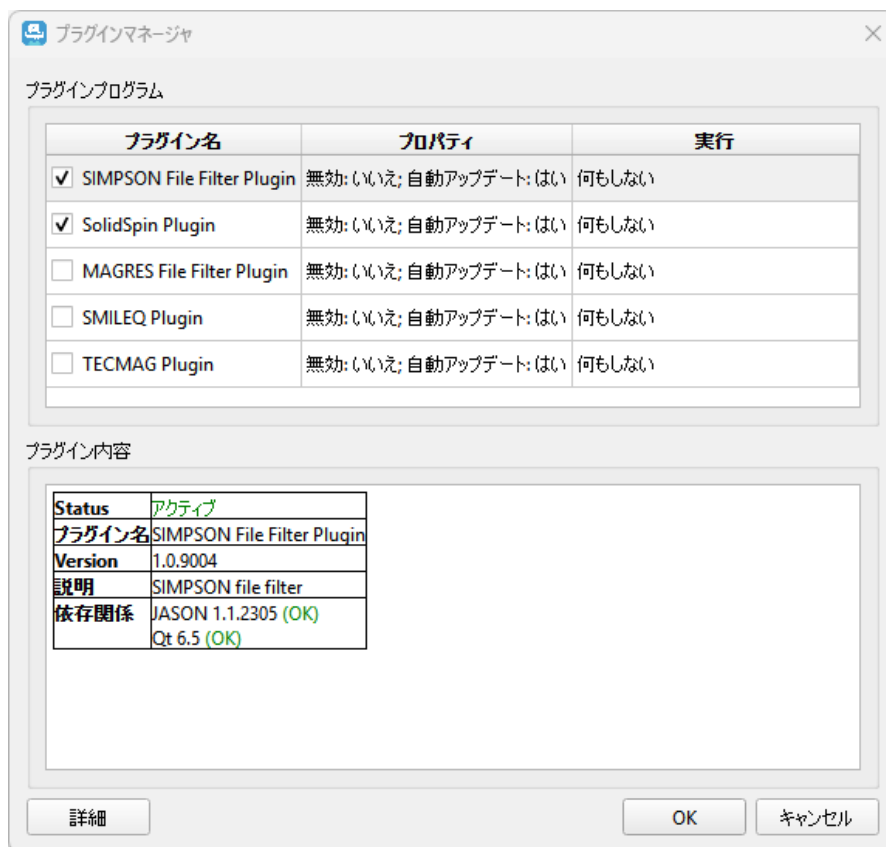
2 インストールとセットアップ

プラグインのインストールは、プラグインマネージャで SolidSpin NMR シミュレーションと SIMPSON ファイル形式のプラグインの両方を有効化し、SolidSpin NMR プラグインの設定をする、という 2 つのステップから成ります。

2.1 プラグインマネージャでのプラグイン有効化

プラグインマネージャは、メインメニューボタン(3 つの点のボタン)からアクセスでき、起動している JASON インスタンスで利用可能なすべてのプラグインを表示します。

プラグインは、使用前に有効化する必要があります。SolidSpin プラグインは、JASON に同梱されており、プラグインマネージャの利用可能なプラグインのリストにあります。リストの「SolidSpin Plugin」の横のチェックボックスをチェックして、有効にしてください。さらに、「SIMPSON File Filter Plugin」が、SIMPSON によって生成されたファイルを読み込むために必要であり、これも有効化する必要があります：

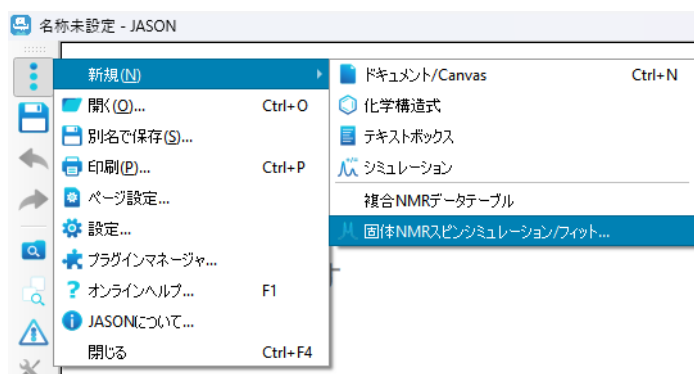


OK をクリックすると、JASON の再起動を促すプロンプトが表示されますので、再起動します。これにより新しく有効化されたプラグインを読み込むことができます。

2.2 プラグインの設定

SIMPSON は、日本電子の外部で開発されたオープンソースのソフトウェアです。そのため日本電子は SIMPSON の配布やサポートは行いません。ユーザーは、SIMPSON を別途インストールし、JASON SolidSpin プラグインを、設定ウィザードを用いて設定する必要があります。

プラグインが正しくインストールされると、JASON のメインメニュー（3つの点）に、「**新規 -> 固体 NMR スピンシミュレーション/フィット...**」という項目が追加されます：



初めてプラグインを起動すると、ウィザードが開始されます：



まず、プラグインに SIMPSON の実行ファイルの場所を指定する必要があります。SIMPSON がインストールされていない場合は、SIMPSON Web というリンク(リンクは本マニュアル作成時のものです)をクリックし、表示される SIMPSON のダウンロードサイトからプログラムを入手してください。そのページの指示に従ってインストールしてください。

SIMPSON のインストールが完了したら、ウィザードのフィールドに SIMPSON の実行ファイル (Windows では simpson.exe) の場所を入力してください。場所が自動で決定されない場合は、参照ボタンを使ってファイルシステムをブラウズして実行ファイルを指定してください。

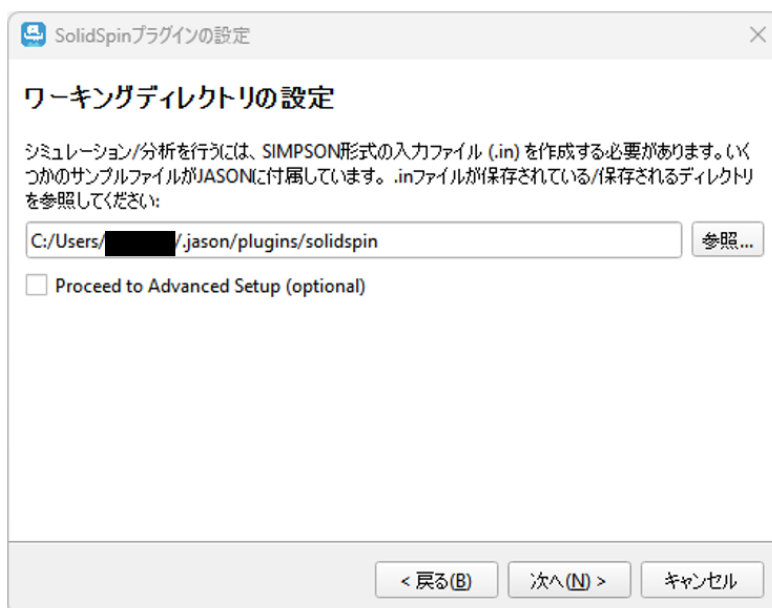
次へをクリックして次のステップに進みます。

このステップでは「ワーキングディレクトリ」を設定する必要があります。このディレクトリには次の役割があります:

- シミュレーションを記述する SIMPSON 入力ファイルの指定
- シミュレーション結果ファイルの一時保存

ユーザーがワーキングディレクトリの読み取り/書き込み権限を持っていることが必要です。

デフォルトの場所は、プラグインライブラリがインストールされているディレクトリです。このディレクトリにはサンプルスクリプト (.in ファイル) も含まれます。



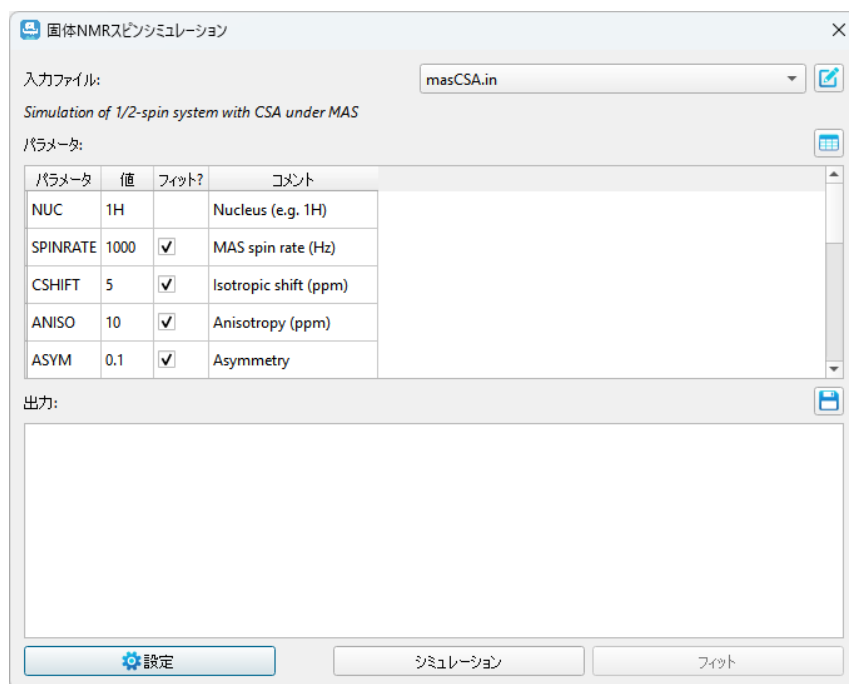
このマニュアルでは、新しいディレクトリを作成し、この新しいディレクトリにサンプルをコピーすることを推奨します。

高度なセットアップが必要な場合は、**Proceed to Advanced Setup** ボックスにチェックを入れ、**N次へ** をクリックします。高度なセットアップでは、SIMPSON 実行時のコマンドライン引数を指定することができますが、一般的な使用では設定不要です。

ウィザードの次のページで、設定ステップが正常に実行されたことを確認し、JASON を再起動します。

「**新規 -> 固体 NMR スピンシミュレーション/フィット...**」をクリックすると、シミュレーションダイアログが開きます。

このセットアップウィザードは、シミュレーションダイアログの左下隅にある**設定**ボタンをクリックすることで、いつでも再実行できます。

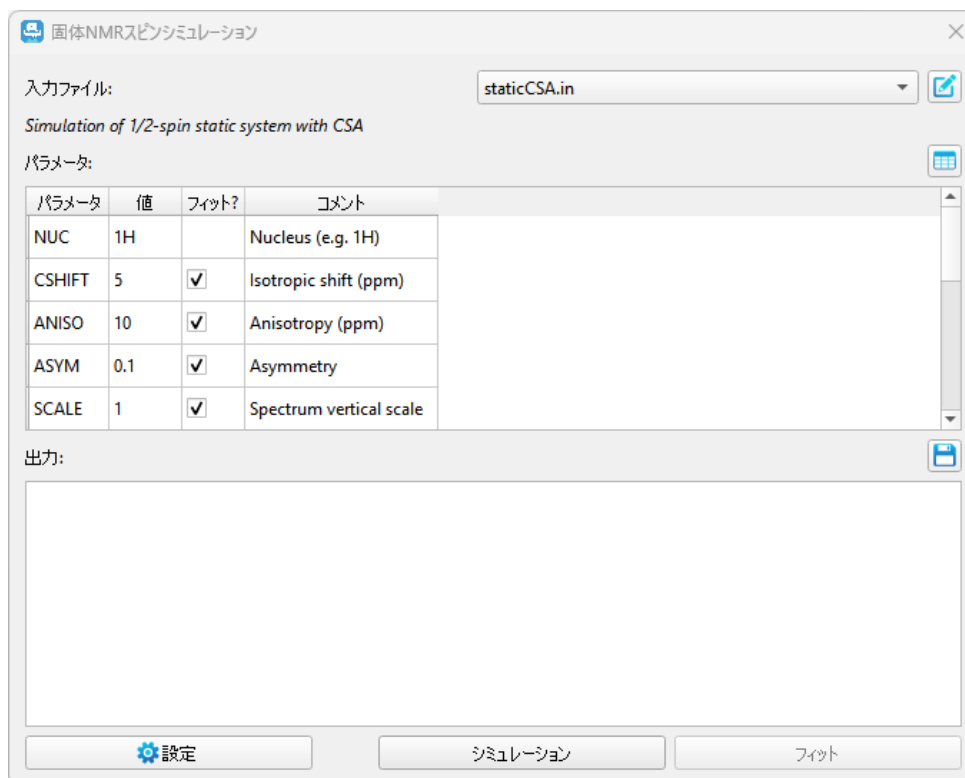


3 静的 CSA シミュレーションとフィッティング

3.1 簡単なシミュレーション

最初の例は、CSA パウダーパターンの簡単なシミュレーションです。

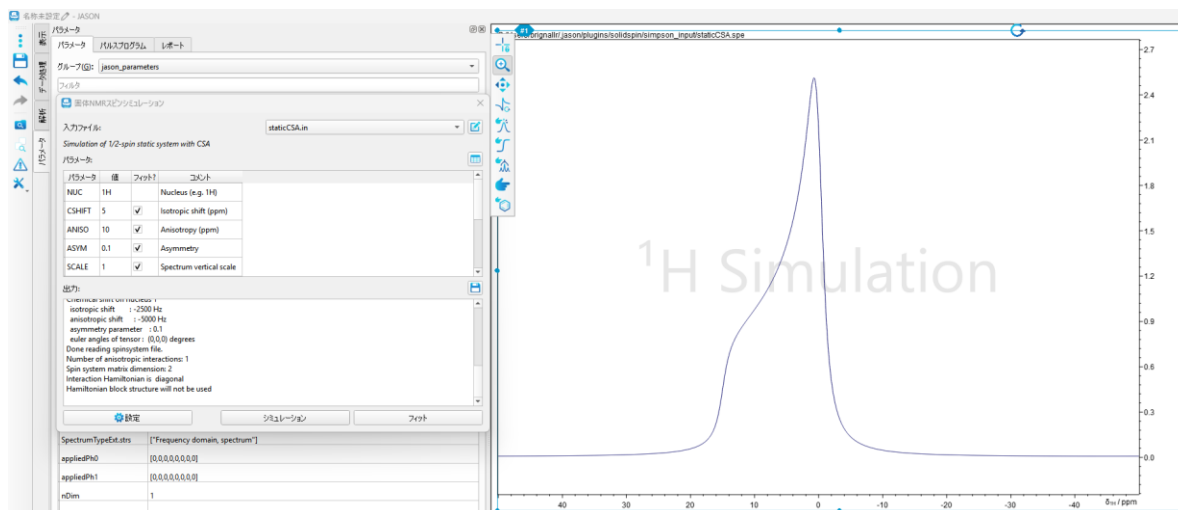
- メインメニュー (...) **新規** -> **固体 NMR スピンシミュレーション/フィット...**でシミュレーションダイアログを開きます。
- ドロップダウンメニューから**入力ファイル** staticCSA.in を選択します。ドロップダウンメニューにファイルが表示されない場合は、セットアップウィザードを再実行して、プラグインに提供されたサンプル入力ファイルの場所を指定してください。
- 入力ファイルは、入力ファイルドロップダウンメニューの隣にある **編集** ボタンを押すことで、表示または編集できます。ボタンをクリックするとファイルを開く指定したテキストエディタを選択するよう指示されます。入力ファイルを開いているテキストエディタを閉じると、ダイアログは自動的に更新されます。
- 入力ファイルを選択すると、ダイアログが更新され、staticCSA.in ファイルで指定されたパラメータが表示されます：



パラメータテーブルには、シミュレーションに使用するパラメータが表示されます。表の「値」の列は編集可能です。値をダブルクリックして、必要なパラメータを入力します。**フィット?**列は、実験スペクトルのフィッ

ティングでそのパラメータを使用するか設定するのに使用します。そのため単純なシミュレーションでは設定する必要はありません。この使い方については、このマニュアルの後半で説明します。

シミュレーションボタンをクリックするとシミュレーションが開始されます。**出力フィールド**には、SIMPSON からのコマンドライン出力（実行エラーを含む）が表示されます。シミュレーションが終了すると、シミュレートされたスペクトルが JASON キャンバスに新しいアイテムとしてロードされます：



シミュレーションダイアログは、ウィンドウ右上の X をクリックして閉じることができます。

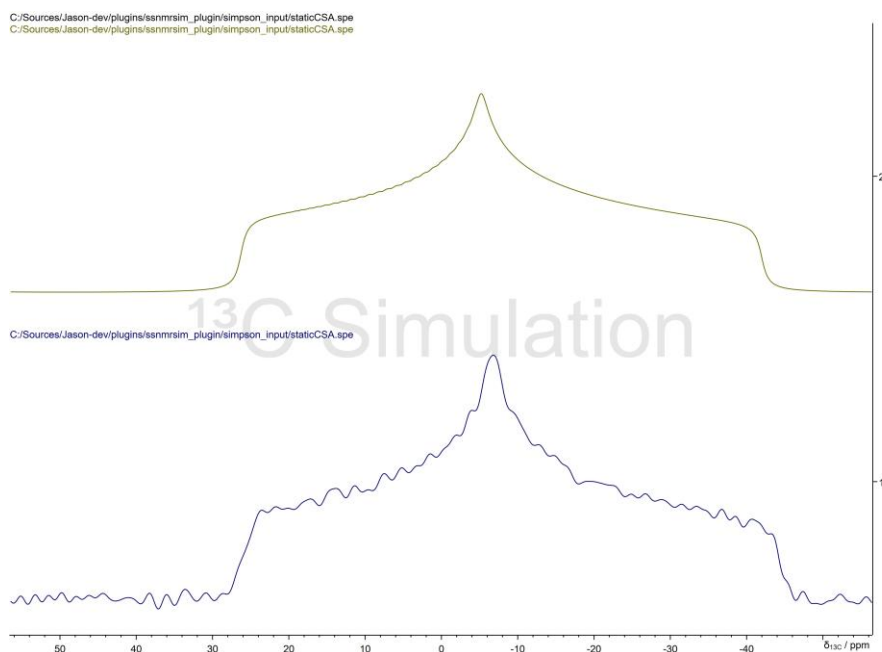
3.2 実験スペクトルに基づくシミュレーション

固体 NMR シミュレーションのより典型的なケースの一つに、シミュレートしたスペクトルを実験スペクトルと比較することが挙げられます。このセクションでは、それを JASON で行う方法を、プラグインに添付されているスペクトルファイル *gly_static.spe* を使って紹介します。

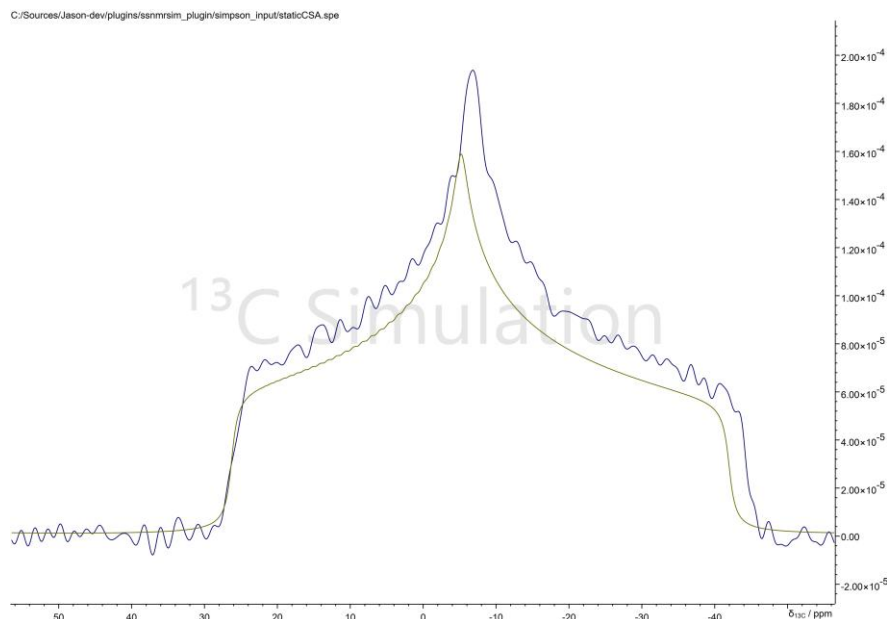
- *gly_static.spe* を開く(このスペクトルはプラグインの使用方法を例示することを目的としており、化学シフトのリファレンスは適切に設定されていません)。
- メインメニュー (...) 新規 -> 固体 NMR スピンシミュレーション/フィット... でシミュレーションダイアログを開きます。 *gly_static.spe* がアクティブであれば、スペクトルは青枠で強調表示されます。アクティブでない場合は、スペクトルをクリックします。
- 入力ファイル *staticCSA.in* をド롭ダウンメニューから選択します。
- シミュレーションプラグインは、表中のいくつかのパラメータを実験パラメータと一致するように埋めます。自動入力されるパラメータには次のものが含まれます： NUC, FREQ, SQ, NP, REF。
- 以下のパラメータを修正します：

| | |
|---------|----------|
| CSHIFT | -7 |
| ANISO | -35 |
| ASYM | 0.9 |
| SCALE | 2e-4 |
| CRYSFIL | zcw28656 |
| LB | 200 |

- シミュレーションボタンをクリックしてシミュレーションを開始します。
- シミュレーションが終了すると、シミュレートされたスペクトルがロードされ、実験スペクトルと垂直スタックで表示されます：



- 別の方法として、例えば、JASON スタックパネルのドロップダウンメニューからスペクトルの重ね合わせを選択することで、スペクトルを重ね合わせモードで表示することができます。スタックパネルの詳細については、JASON のマニュアルを参照してください



実験に近いパラメータを選びましたが、完全には一致していないことがわかります。次のセクションでは、実験スペクトルのフィット方法について説明します。

3.3 実験スペクトルのフィッティング。

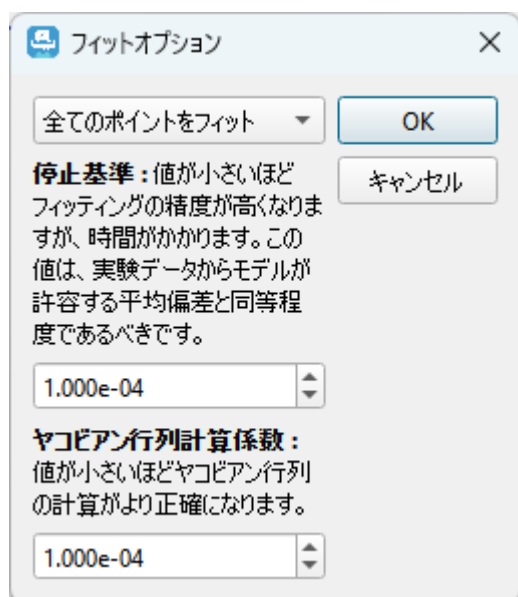
SolidSpin プラグインは、JASON の非線形フィッティングルーチンを使って、シミュレーションが実験スペクトルに合うまで、指定されたパラメータを自動的に調整します。非線形フィッティングルーチンは、パラメータの初期値に敏感です。真の値に近いパラメータを選択することで、ローカルミニマムを回避し、フィッティングを高速化できます。

前節 3.2 の手順を実行し、初期パラメータを設定します。シミュレートしたスペクトルが実験スペクトルに十分近くなるまでパラメータを手動で調整することを推奨します。

フィッティングするパラメータを選択します。非線形フィッティングは、どのパラメータをフィッティングに使用するかによって計算時間やグローバルミニマムに収束するか、などが変化します。どのパラメータを調整すべきか慎重に検討してください。フィッティングするパラメータに対して、フィット？ 欄のボックスにチェックを入れてください。チェックボックスは、入力ファイルで指定されたパラメータに対してのみ表示されます。入力ファイルのフォーマットについては、このマニュアルの次のセクションで説明します。

スタティック CSA の例では、フィッティング指定可能なパラメータ(CSHIFT, ANSIO, ASYM, SCALE) はすべてチェックしておくことができます。

フィットをクリックしてフィットを開始します。フィッティングパラメータのダイアログが表示されます：



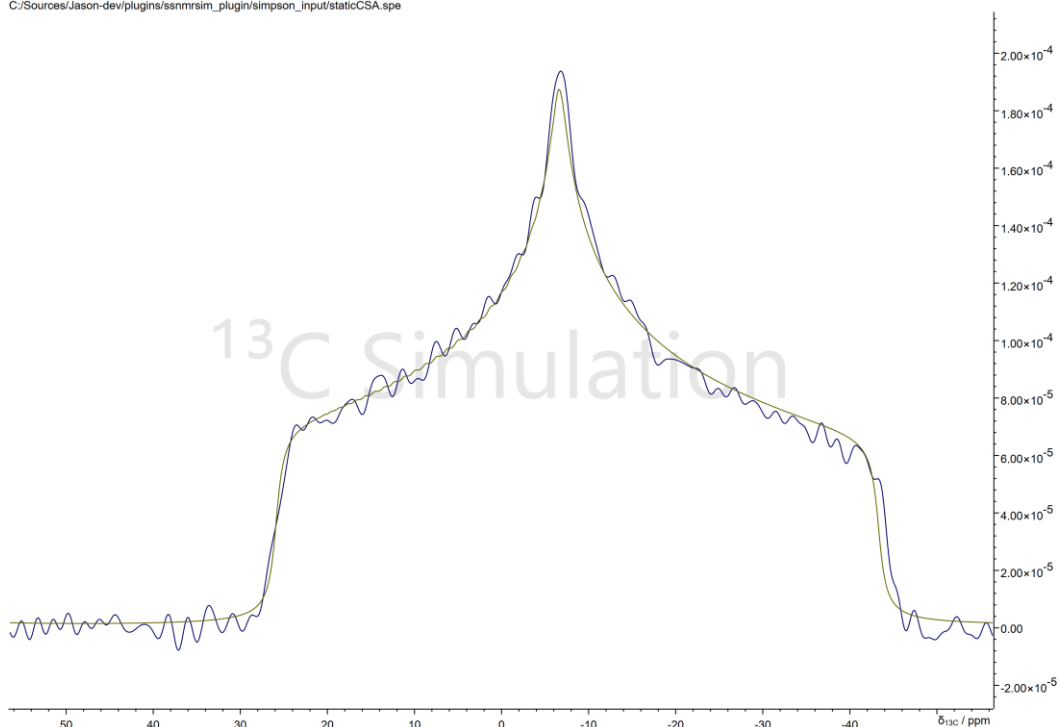
全てのポイントをフィットオプションが選択されていることを確認します。このオプションを選択すると、実験スペクトルのすべてのポイントが、フィッティングの過程でシミュレーション結果と比較されます。ドロップダウンメニューの他のオプションについては、後で説明します。

他の2つのパラメータは、フィットの品質を指定します。

OK をクリックし、フィッティングが終わるまで待ちます。フィッティングルーチンの動作している間、シミュレートされたスペクトルがキャンバスの項目で更新され、フィッティングの進行状況が表示されます。必要であれば、**シミュレーションを終了**ボタンをクリックして、フィッティングルーチンを中断することができます。

フィッティングが正常に終了すると、ダイアログが表示されます。最終スペクトルはキャンバス項目に表示されます：

C:/Sources/Jason-dev/plugins/ssnmrsim_plugin/simpson_input/staticCSA.spe



パラメータテーブルが更新され、最適化された値とその誤差を含む**フィット値**と**σフィット**の列が追加されます：

固体NMRスピンドルシミュレーション

入力ファイル: staticCSA.in

Simulation of 1/2-spin static system with CSA

パラメータ:

| パラメータ | 値 | フィット? | コメント | フィット値 | σフィット |
|--------|--------|-------------------------------------|-------------------------|-------------|-------------|
| NUC | 13C | | Nucleus (e.g. 1H) | | |
| CSHIFT | -7 | <input checked="" type="checkbox"/> | Isotropic shift (ppm) | -8.04356 | 0.0114733 |
| ANISO | -35 | <input checked="" type="checkbox"/> | Anisotropy (ppm) | -35.2887 | 0.0174508 |
| ASYM | 0.9 | <input checked="" type="checkbox"/> | Asymmetry | 0.9214 | 0.000823345 |
| SCALE | 0.0002 | <input checked="" type="checkbox"/> | Spectrum vertical scale | 0.000238894 | 2.19105e-07 |

出力:

```

Chemical shift on nucleus 1
isotropic shift : 1206.53 Hz
anisotropic shift : 5293.3 Hz
asymmetry parameter : 0.9214
euler angles of tensor: (0,0,0) degrees
Done reading spinsystem file.
Number of anisotropic interactions: 1
Spin system matrix dimension: 2
Interaction Hamiltonian is diagonal
Hamiltonian block structure will not be used
    
```

設定 編集

フィッティング後、表は読み取り専用モードになっています。フィッティングをやり直したい場合は、編集ボタンを押して編集モードを再度有効にする必要があります。このときフィットの列は削除されます。初期パラメータを維持するか、最後にフィッティングした値を次のフィッティングサイクルの初期パラメータとして使用するか選択できます。

キャンバスに**テーブルを作成**ボタン(表右上のボタン)を押すと、パラメータ表がキャンバスに出力されます。

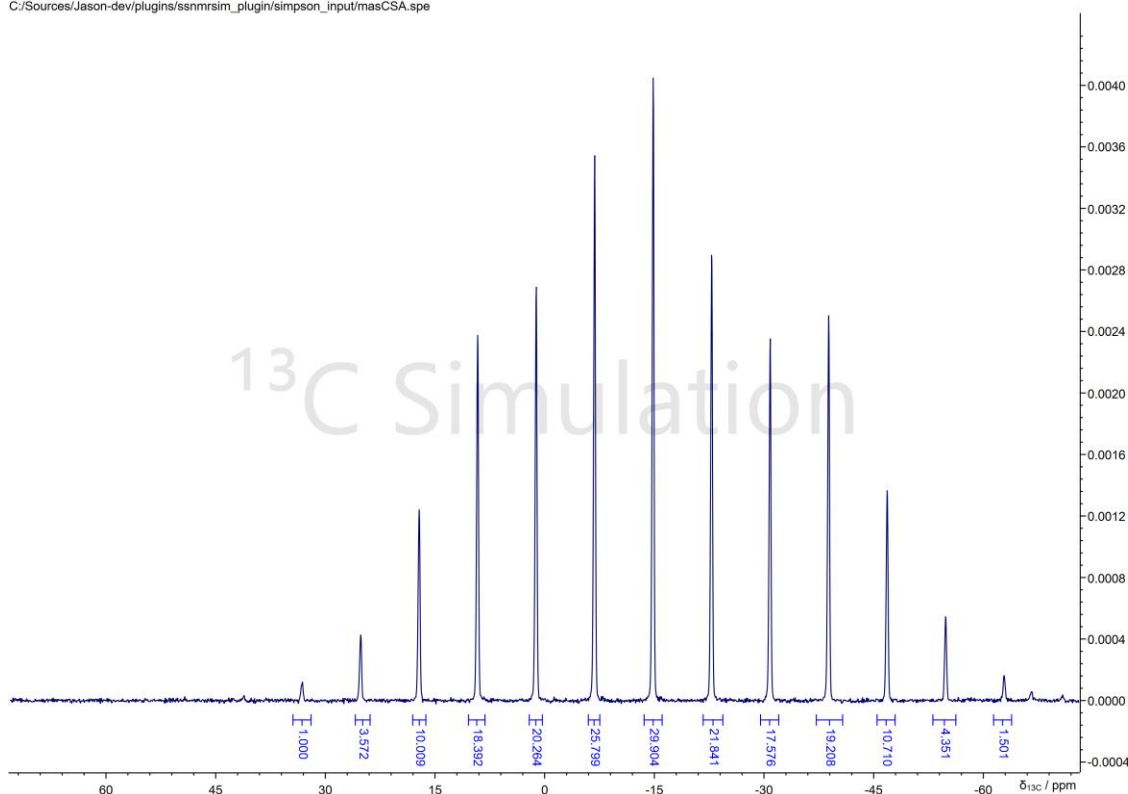
4 CSA のシミュレーションとフィッティング

この例では、スピン-1/2 スペクトルの MAS サイドバンドパターンを CSA でフィッティングします。多くのステップはスタティック CSA シミュレーションと同様です。よって、ここでは *13C_gly_CP_1kHz.spe* を使用し重要なステップと機能に焦点を当てます。

13C_gly_CP_1kHz.spe を開きます。全スペクトルポイントを使用したフィッティングが可能ですが、このスペクトルは複数のシャープなサイドバンドシグナルで構成されているためフィッティングが不安定になり、多くのローカルミニマムが発生する可能性があります。そこで、サイドバンド全体の強度のみをフィッティングに使用する特別なアプローチを使用します。そのためには、いくつかの準備が必要です。

JASON の手動積分ツールを選択し、各サイドバンド周辺の領域を積分します。見やすくするために、**表示パネルで積分曲線**をオフにすることができます。積分されたスペクトルはこのようになります：

C:/Sources/Jason-dev/plugins/ssnmrsim_plugin/simpson_input/masCSA.spe



各積分の値は、各スピニングサイドバンドの強度に対応します。これらの積分値の絶対値（スクリーンショット上の 13 個の積分値）のみがフィッティングルーチンで使用されます。この方法では、例えば複数の成分が混在した複雑なスペクトルから、特定の成分のサイドバンドパターンを選択することも可能です。

- メインメニュー (...) **新規 -> 固体 NMR スピンミュレーション/フィット...**でシミュレーションダイアログを開きます。
- ドロップダウンメニューから入力ファイル masCSA.in を選択します。
- 以下のパラメータを変更します：

| | |
|--------|--------|
| CSHIFT | -14.85 |
| ANISO | -35 |
| ASYM | 0.9 |
| SCALE | 2e-4 |
| LB | 28 |

- **シミュレーション**をクリックしてシミュレーションを開始し、得られたパターンが実験スペクトルに十分近いことを確認します。

ここで積分値を使用するため、フィッティングに含める必要があるパラメータを考えてみます。

積分値を使用するため、CSHIFT（等方化学シフト）や SPINRATE（MAS 速度）のわずかなずれは、シミュレートされたサイドバンドが実験値に十分近い限り、フィッティングに大きな影響を与えません。したがって、実験スペクトルからこれらの値を推定し、**Fit?**欄のチェックを外して固定する方がよいです。

LB(指数関数窓の強さ)と SCALE(スペクトル強度のスケール)には積分値に関して逆相関の関係があります。例えば、LB を小さくし、SCALE をある程度大きくすることで、同じ積分値が得られます。このような相関のあるパラメータは望ましくなく、フィッティングを不安定にします。そこで、このフィッティングでは LB 値を固定します(LB と SCALE を実験データから推定するのは比較的簡単です、またこれらの正確な一致は今回の目的においては重要ではありません)。

したがって、フィッティングは ANISO、ASYM、SCALE のみに対して行います。次の手順でフィッティングを行います：

- フィットをクリックします
- 表示されるダイアログで、上述したように積分値のみをフィットするために、**積分値をフィット**を選択します。
- 停止基準とヤコビアンに関するパラメータをそれぞれ 1e-6 に設定するなどして、調整するとよいでしょう。
- フィットを実行します

ここまでで CSA パラメータだけが調整されます。スキップされたパラメータ(CSHIFT, SPINRATE, LB)の推定を含む、より質の高いフィットが必要な場合は、実験スペクトルのポイントを使用した解析に進みます。このとき**全てのポイントをフィット**または**積分範囲内でフィット**のオプションを利用可能です。**積分**

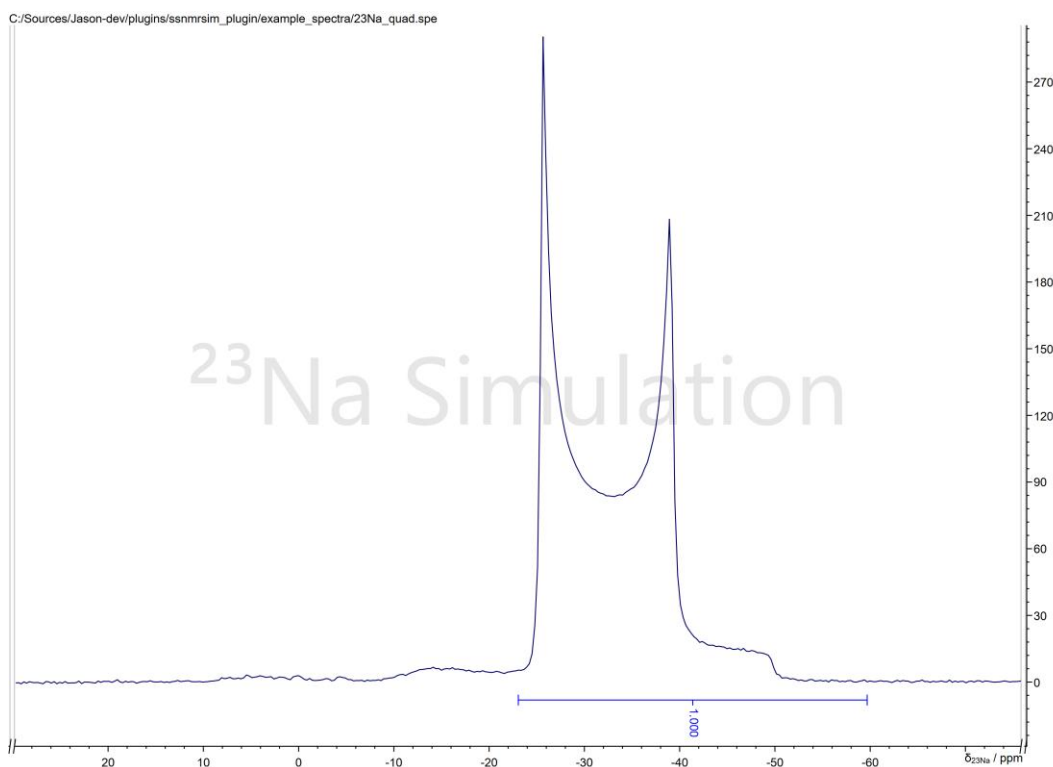
範囲内でフィットオプションは、ユーザー定義の積分を使用します。ただし、**積分値をフィット**のように 1 つの積分値を使用するのではなく、積分範囲内のすべてのスペクトルポイントを使用します。

5 スピン N/2 四極子スペクトルの二次四極子スペクトル幅のフィッティング。

この例では、 ^{23}Na の MAS スペクトルの中心遷移をフィッティングします。多くのステップは前述の CSA のシミュレーションと同様です。そのため、この例で重要な特定のステップと機能に焦点を当てます。ここでは、23Na_quad.spe の解析を**積分範囲内でフィット**を使用して行います。

まず 23Na_quad.spe を開きます。このスペクトルの観測幅は広がっています。そこで、フィッティングの前に 2 つの準備作業を行います：

- フィッティング範囲を指定するために、以下のように中心遷移のシグナル周辺を積分します：



シグナルの左側はシミュレーションで再現できず、フィッティング範囲に含めるとその結果に影響を与える可能性があるため除外しています。右側の積分は、ベースラインを含むように少し広めに選択しています。

上のスクリーンショットでは、表示パネルの表示範囲の切り落としを使い、手動で不要な部分を切り落としています。カットは、横軸のカットマーカー(//)で示されるように、スペクトルの左右の部分を隠しています。これは、フィッティング中に中心遷移のピークのみを表示するために行っています。この作業は実際のデータには影響を与えません。

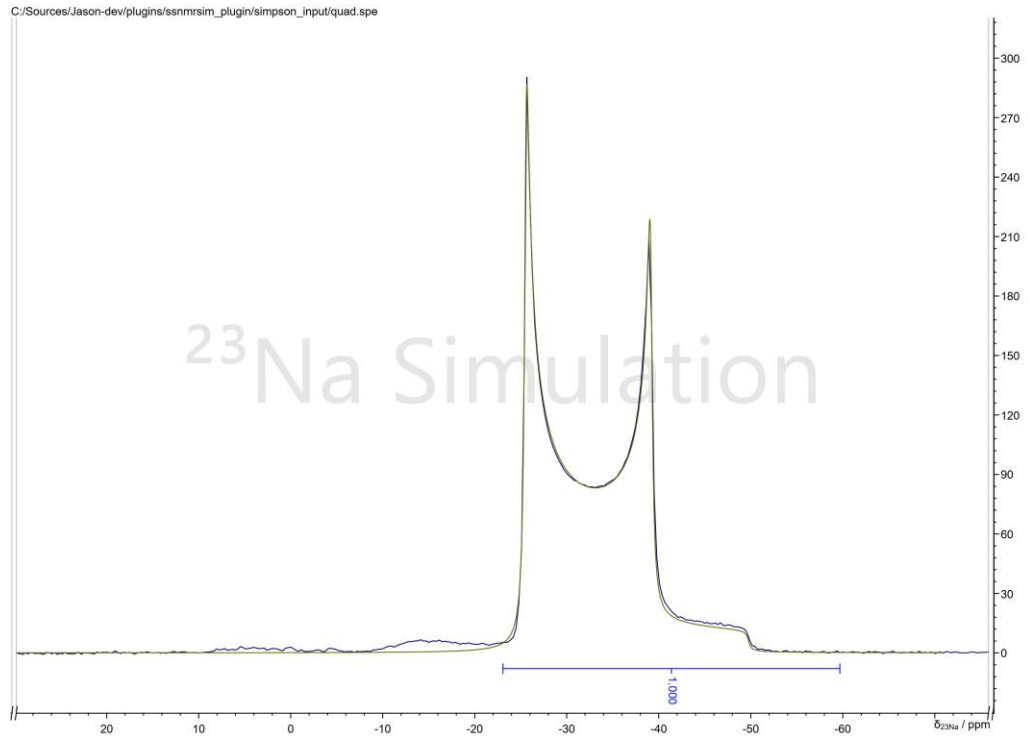
次の操作でフィッティングを行います：

- ドロップダウンメニューから quad.in 入力ファイルを選択します
- 以下の通りにパラメータを変更します：

| | |
|--------|-------|
| CSHIFT | -22 |
| ORDER | 2 |
| CQ | 2.2 |
| ETA | 0.0 |
| SCALE | 10 |
| LB | 20 |
| SW | 20000 |

シミュレートされたスペクトルは、実験スペクトルよりも狭いスペクトル幅を持つことに注意してください。一般に、シミュレーションでのスペクトル幅は、実験スペクトルと一致させる必要はありません。2つのスペクトルの共通範囲のみがフィッティングのターゲット関数に使用されます。シミュレーションと実験スペクトルは、実験スペクトルのポイントと一致するように信号が補間されます。

- **シミュレーション**を実行して、シミュレートしたパターンが実験スペクトルに十分近いことを確認してください。
- フィッティングパラメータに CSHIFT、CQ、SCALE、LB を設定します。スペクトルの狭い範囲でフィッティングするため SPINRATE は結果に大きな影響を与えません。そのため実験値とほぼ等しい固定パラメータとして設定します。ETA も同様にフィットできますが、この例では 0.0 で信号の特徴を再現できるため、0.0 に固定します。
- **フィット**をクリックする。
- **積分範囲内でフィット**を選択します。準備作業で指定した積分範囲がフィッティングに使用されます。これによりシミュレーションで再現できない部分はフィッティングの評価から除外されます。停止基準とヤコビアンに関するパラメータのデフォルト値（ $1e-4$ ）はそのままにします。
- フィッティングの結果は以下ようになります：



キャンバスに**テーブルを作成**ボタン(表右上のボタン)を押すと、パラメータ表がキャンバスに出力されます。

6 入力ファイルの書式

カスタム入力ファイルを作成し、ワーキングディレクトリに追加することで、SIMPSON の全ての機能を利用することができます。

SolidSpin プラグインの入力ファイルは、SIMPSON の標準の.in ファイルに、プラグインとの連携に必要な記述を加えたものです。この拡張により、プラグイン上でのパラメータ表示、あるいはフィッティングで使用するパラメータの設定変更が可能となります。このマニュアルでは、この記述方法にのみ焦点を当てます。.in ファイルの説明は本マニュアルの範囲外であり、次に挙げる SIMPSON 開発者の論文を参照してください:

<https://doi.org/10.1006/jmre.2000.2179>

<http://dx.doi.org/10.1016/j.jmr.2014.07.002>

<https://doi.org/10.1016/bs.arnmr.2019.12.001>

プラグイン上で表示・設定するパラメータは、.in ファイル内で「##」から始まる拡張記法を使用して定義します。これらは SIMPSON ではコメントとして扱われるため、シミュレーション中は無視されます。JASON プラグインは、これらの行を解析し、シミュレーションタイトルとパラメータ値を取得します。

SIMPSON と JASON は、一時的なシステム変数（シミュレーション中にのみ有効な変数）を介してパラメータを交換します。シミュレーションを実行する際、SolidSpin プラグインはシステム変数を作成し、SIMPSON を実行します。SIMPSON の.in ファイルには、これらのシステム変数の値を読み込み、シミュレーションで使用するようコードを記述します。

フィッティングルーチンは、JASON 内部のフィッティングエンジンを使用します。JASON プラグインは複数のシミュレーションを、パラメータを変更しながら実行します。SIMPSON 内蔵のフィッティングエンジンを使用しないため、スペクトルをシミュレートするだけの簡単な入力ファイルだけが必要です。そのため.in ファイルが簡素化されます。

masCSA.in を例に説明します :

```
##title: <i>Simulation of 1/2-spin system with CSA under MAS</i>
```

```
##par:NUC;1H;;fixed;Nucleus (e.g. 1H)
```

```
##par:SPINRATE;1000.0;10.0,1000000.0,1;variable;MAS spin rate (Hz)
```

```

##par:CSHIFT;5.0;-100000.0,100000.0,1;variable;Isotropic shift (ppm)
##par:ANISO;10.0;-10000000.0,10000000.0,1.0;variable;Anisotropy (ppm)
##par:ASYM;0.1;0.0,1.0,0.05;variable;Asymmetry
##par:SCALE;1.0;0.0,1000000000000000.0,0.001;variable;Spectrum vertical
scale
##par:LB;50;0.0,10000.0,1.0;variable;Linebroadening (Hz)
##par:FREQ;500.0;1.0,3000.0,100.0;fixed;Proton frequency (MHz)
##par:CRYSTFILE;rep320;;fixed;Crystal file
##par:SW;50000;10.0,20000000.0,100.0;fixed;Spectrum width (Hz)
##par:NP;8192;1024,524288,1024;fixed;Number of points in FID
##par:REF;0;-20000000.0,20000000.0,10.0;fixed;Reference shift (Hz)

```

```

spinsys {
  channels 1H
  nuclei 1H
  shift 1 0 10p 0 0 0 0
}

```

```

par {
  spin_rate      -1

  np              2048
  proton_frequency -1
  start_operator  I1x
  detect_operator I1p
  method          direct
  crystal file    -1
  gamma_angles   8
  sw              -1
  variable tsw    -1
  verbose         1111
}

```

```

proc pulseseq {} {
  global par

  acq_block {
    delay $par(tsw)

```

```

}
}

proc main {} {
  global par
  global spinsys
  set spinsys(channels) $::env(NUC)
  set spinsys(nuclei) $::env(NUC)
  #spectrometer frequency for the isotope NUC
  set sfrq [resfreq $::env(NUC) [expr $::env(FREQ)*1e6]]
  #Isotropic shift in Hz with the adjustment of spectrum reference
  set adjshft [expr $::env(CSHIFT)*($sfrq+$::env(REF))*1e-6-$::env(REF)]
  set spinsys(shift) "1 $adjshft $::env(ANISO)p $::env(ASYM) 0 0 0"

  set par(spin_rate) $::env(SPINRATE)
  set par(proton_frequency) [expr $::env(FREQ)*1e6]
  set par(crystal_file) $::env(CRYSFIL)
  set par(sw) [expr {floor($::env(SW)/$par(spin_rate))*$par(spin_rate)}]
  set par(tsw) [expr {1e6/$par(sw)}]

  set f [fsimpson]
  fset $f -ref $::env(REF)
  faddlb $f $::env(LB) 0
  fzerofill $f $::env(NP)
  fft $f
  fphase $f -scale $::env(SCALE)
  fsave $f $par(name).spe
}

```

6.1 シミュレーションタイトル・パラメータの定義

SolidSpin プラグインの入力ファイルは、ユーザーに表示されるシミュレーションタイトル、パラメータを定義するセクションから始まります。このセクションの各行は「##」で始まる拡張記法で構成されています。# は SIMPSON のコメント識別子として使用されるため、SIMPSON はこのセクションを無視します。拡張記法とコメントを分けるために、#を二重に使用しています。

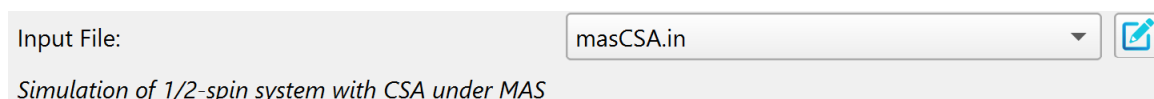
拡張記法によるシミュレーションタイトルの定義例：

`##title: <i>Simulation of 1/2-spin system with CSA under MAS</i>`

書式は次の通りです :

`##title: <user-defined title for the simulation>.`

シミュレーションファイルを選択すると、シミュレーションダイアログにタイトルが表示されます。入力ファイルの機能を説明するために、任意のテキストを使用できます。HTML タグも使用できます。上の例では、タイトルに `<i>…</i>` タグを使用したことで次のようにイタリック体で表示されます。:



`par:` で始まる行は、**パラメータ**テーブルの行の各要素を定義します。その書式は次の通りです :

`##par: <parameter name>; <default`

`value>; <minimum>, <maximum>, <step>; <variable/fixed>; <comment>`

- `<parameter name>`: パラメータ名とシステム変数名。上述したように、SIMPSON と SolidSpin プラグインは、一時的なシステム変数を介して、パラメータの値を交換します。そのためパラメータ名にはシステム変数名と同じ制限があります。トラブルを避けるために、特殊文字を使用せず、すべて大文字の単一単語を名前に使用することを推奨します。
- `<value>` は、プラグインダイアログに表示されるデフォルト値です。数値または文字列です。アクティブな実験スペクトルを選択してシミュレーションを開始した場合、いくつかの予約パラメータの値は実験スペクトルから取得され、デフォルト値の代わりに使用されます。
 - NUC: 核種 (例: 13C)
 - SW: スペクトル幅 (Hz)
 - FREQ: 分光器周波数 (MHz)
 - REF: 化学シフト基準オフセット (Hz)。
 - SPINRATE: MAS 速度等の試料回転数 (Hz)
 - NP: スペクトルポイント数。
- `<minimum>`, `<maximum>`, `<step>` は、パラメータの最小値と最大値を指定します。`<step>` は、GUI で上下の矢印を使って値を変更するときのステップ幅として使われます。`<minimum>` と `<maximum>` はフィッティングのパラメータ制約としても使用され、フィッティング中、パラメータはこの範囲内に制限されます。`<minimum>`, `<maximum>`, `<step>` は他の要素と異なり「, 」で区切る必要があります。文字列パラメータなどでは省略可能です。
- `<variable/fixed>`: パラメータに「variable」キーワードが使用されている場合、フィッティング中にパラメータを調整することができます。「フィット?」欄にチェックボックスが表示され、フィッティング中にパラメータを調整するか、一定にするかを選択できます。「fixed」キーワードを指定したパラメータは常に一定に保たれます。

- <comment>はユーザー定義の文字列で、パラメータの用途を説明します。

par 行の記述例は次の通りです：

```
##par:SPINRATE;1000.0;10.0,1000000.0,1;variable;MAS spin rate (Hz)
```

この記述ではパラメータとシステム変数名を SPINRATE と定義します。デフォルト値は 1000.0 です。SPINRATE は予約されたパラメータ名の 1 つであるため、シミュレーション時に実験スペクトルを選択すると、その値は MAS 速度の実験値に置き換えられます。許容される値の範囲は 10.0-1000000.0 です。GUI でのインクリメント幅は 1.0 です。パラメータは変数として指定されており、フィッティングの変数として設定することができます。**Parameters table** **パラメータテーブル**の**コメント**欄に「MAS spin rate (Hz)」という文字列が表示されます。

```
##par:CRYSTFILE;rep320;;fixed;Crystal file
```

この記述ではパラメータとシステム変数 CRYSTFILE を定義します。デフォルト値は「rep320」という文字列です。これは SIMPSON で定義されている「クリスタルファイル」の名前の一つです。文字列パラメータでは、許容値の範囲と GUI でのインクリメント幅は省略されます。パラメータは固定です。フィッティングのための変数として定義できるのは数値パラメータのみであり、文字列パラメータは「fixed」を指定する必要があります。パラメータテーブルの**コメント**欄に「Crystal file」という文字列が表示されます。

6.2 spinsys, par, proc pulseseq セクション

これらのセクションは、標準の SIMPSON ファイルからほとんど変更されていません。これらのセクションで定義されている多くのパラメータ値は、SolidSpin プラグインからシステム変数経由で渡される値で上書きする必要があります。この例では、proc main セクションの後半で行います。proc main は、シミュレーションが呼び出されたときに実行される Tcl コードで、この時点でパラメータの値を設定しておく便利です。

読みやすくするために、後で変更する spinsys と par 内部のパラメータを『-1』などでマークすることをお勧めします。次に例を示します：

```
crystal_file -1
```

または

```
sw          -1
variable tsw -1
```

6.3 proc main セクション

上述したように、このセクションはプラグインを使用するために若干の変更が加えられています。proc main は、シミュレーションが呼び出されたときに実行される Tcl コードです。proc main セクション実行時に SolidSpin プラグインで設定したパラメータを、システム変数を介して設定します。

システム変数にアクセスするための Tcl 言語の構文は以下の通りです：

```
::env(<variable_name>)
```

proc main セクションの冒頭にいくつか例があります：

```
set spinsys(nuclei) ::env(NUC)
...
set par(spin_rate) ::env(SPINRATE)
set par(proton_frequency) [expr ::env(FREQ)*1e6]
```

最後の行は、Tcl コマンド expr の使用方法の例として挙げています。この例では、ユーザーが MHz で入力した周波数を SIMPSON が要求する Hz 単位に変換しています。

特殊なケースについても説明します。次の例では、スペクトルウィンドウに特別な計算を使用しています：

```
set par(sw) [expr {floor(::env(SW)/$par(spin_rate))*$par(spin_rate)}]
```

数値シミュレーションを効率的に行うためには、シミュレーションパラメータの値を制限する必要があります。例えば、FID のサンプリング間隔（スペクトル幅の逆数）を MAS ローター周期の整数倍に保つことで計算を高速化できます。しかし、実験スペクトルのフィッティングを行う場合、実験スペクトルのスペクトル幅がこの条件を満たさないことがあります。

SolidSpin プラグインでは、実験スペクトルとシミュレーションスペクトルの帯域幅をそろえる必要があるという条件を取り除くことで、この問題を克服しています。フィッティングにおいて、シミュレーションスペクトルの帯域幅は実験スペクトルの解析対象領域がカバーされていれば十分です。シミュレーションと実験スペクトルのスペクトル幅が一致しない場合、補間によるシミュレーションスペクトルのリサンプリングが適用されます。このとき 2 つのスペクトルの共通範囲のみがリサンプリングされます。この領域内で線形補間が適

用され、実験点の位置におけるシミュレーション値が得られます。リサンプリングされたシミュレーション点は、フィッティングルーチンで使用されます。

上記のコードでは、シミュレーションで使用される `par(sw)` は、`$$::env(SW)` として実験スペクトルのスペクトル幅から推定されますが、MAS 速度の整数倍に最も近くなるように調整されます。これにより、スペクトル幅と MAS 速度の正確な値に関わらず、高速にシミュレーションを行うことができます。例えば、MAS 速度のフィッティングでもこの方法で可能です。補間により、実験とシミュレーションのスペクトルのデータ点の間にある小さなずれに対処します。

また、等方化学シフト値とスペクトルのリファレンシングについても特別な配慮がなされています。関連するコードは次の通りです：

```
#Isotropic shift in Hz with the adjustment of spectrum reference
set adjshft [expr $$::env(CSHIFT)*($sfrq+$$::env(REF))*1e-6-$$::env(REF)]
set spinsys(shift) "1 $adjshft $$::env(ANISO)p $$::env(ASYM) 0 0 0"
```

SIMPSON のシミュレーションスペクトルは、*main* セクションにある次のコマンドでシミュレーション後にリファレンシングされます。

```
fset $f -ref $$::env(REF)
```

上記の REF パラメータは、スペクトルの中心の位置を Hz 単位で設定します。シミュレーションを実験スペクトルと比較するために実行した場合、値は自動で推定され設定されます。

しかし、SIMPSON でのリファレンシングがシミュレーション完了後に実行されるため問題が発生します、すなわち *spinsys (shift)* エントリーで使用される等方化学シフトはリファレンシングされていません ($REF=0.0$)。そのため、これらの値は実験スペクトルで観測された化学シフトと一致しません。

そのため、シミュレートするスピンの化学シフトを実験値でリファレンシングするために次のコードを使用します：

```
set adjshft [expr $$::env(CSHIFT)*($sfrq+$$::env(REF))*1e-6-$$::env(REF)]
```

変数 *adjshft* (リファレンシング後シフト) には、Hz 単位で表されたリファレンシング後シフトが代入されます。

まず、リファレンシング済み化学シフトを、 $$$sfrq + $$::env(REF)$ (どちらの変数も Hz 単位) を基準周波数として Hz に変換します：

```
$$::env(CSHIFT)*($sfrq+$$::env(REF))*1e-6
```

さらに $$$::env(REF)$ でシフトさせます：

```
 $\\$::env(CSHIFT)*($sfrq+\\$::env(REF))*1e-6-\\$::env(REF).$ 
```

計算された値は Hz 単位になっているため、『p』接尾辞を付けずにシミュレーションに使用します:

```
set spinsys(shift) "1 $adjshft $::env(ANISO)p $::env(ASYM) 0 0 0"
```