
Twin Screw Simulator (Ver.9.0.0)

TSSカスタマイズ機能 補足資料

- ・ユーザ定義ルーチン / Fortran compiler の利用方法 pp. 1- 2
- ・開発環境の設定例 ① Intel Fortran pp. 3- 9
- ・開発環境の設定例 ② GFortran pp.10-18
- ・サンプルプログラム pp.19-40
- ・変数一覧 pp.41-48

ユーザ定義ルーチン / Fortran compiler の利用方法

TSS Ver.9.0.0 で利用可能な開発環境 / 下表2種類のいずれかを利用してください.

開発環境 (compiler)	解析で使用するマトリクスソルバ	使用PCのOS (Operating System)
① Intel Fortran	SMSソルバライブラリ (TSS標準ソルバ)	Windows
② GFortran (GNU Fortran)	当社自社開発ソルバ	Windows (MinGWを使用)

(参考) TSS Ver.9.0.0 / GPU対応版 (Ver.8.0.0 から提供) はユーザ定義ルーチンのサポート対象外になりますが、ユーザ定義ルーチン以外の解析には使用可能です。

開発環境 (compiler)	解析で使用するマトリクスソルバ	使用PCのOS (operating system)
NVIDIA nvfortran	CUDAライブラリ (TSS/GPU版標準ソルバ)	Linux (PGI Fortranは2021年に販売停止)

* TSS/GPU対応版が利用可能なPC環境は、下記2点を満たす必要があります。

1. NVIDIA グラフィックボードが搭載されていること。
2. GPUのCUDA Computer Capability (グラフィックスボードのハードウェアバージョン) が7.5以下であること。

当社社開発環境 GPU: Nvidia GeForce GTX 1660 Ti, CUDA Computer Capability: 7.5, Compiler: PGI Community Edition

TSS Ver.9.0.0 フォルダ構成

TwinScrewSimulatorVer9.0.0¥bin¥x86 内には3つのSystemフォルダが存在します.



フォルダの内容と対象ユーザ

標準仕様

⇒ pp.3-9

Systemver.9.0.0 【TSS/標準版】：通常解析を実施されるユーザ、①Intel Fortranを利用されるユーザ
Systemver.9.0.0_GPU 【TSS/GPU版】：GPUを利用されるユーザ(利用可能なPC環境はp.1参照)
Systemver.9.0.0_GFORT : ②GFortranを利用されるユーザ

⇒ pp.10-18

開発環境の設定例 ① Intel Fortran

1) Intel® oneAPI HPC Toolkit をダウンロードする.

入手可能なウェブサイト(下図に抜粋)

<https://www.intel.com/content/www/us/en/developer/articles/news/free-intel-software-developer-tools.html>

Free Intel® Software Development Tools

Published: 01/12/2021
Last Updated: 02/01/2021

Intel® oneAPI Toolkits - Free for All Developers

Intel® oneAPI Toolkits are the next generation of standards-based Intel® Software Development Tools for building applications across diverse architectures. All Intel® oneAPI Toolkits products are available at no cost. The Intel oneAPI Toolkits do not require license files and the terms of use are based on the End User License Agreement. Support is available via Intel Developer Zone community forums.

Native Code Toolkits

Intel® oneAPI Base Toolkit

Get started with this foundational kit that enables developers of all types to build, test, and deploy performance-driven, data-centric applications across CPUs, GPUs, and FPGAs. For specialized workloads, use the Base Kit with one or more add-on toolkits.

[Get the Base Kit](#) + Add a Domain-Specific Toolkit

Add Domain-Specific Toolkits for Specialized Workloads

Intel® oneAPI HPC Toolkit

Deliver fast DPC++, C++, Fortran, OpenMP, and MPI applications that scale.

[Get the Base Kit](#) + [Get the HPC Kit](#)

Intel® oneAPI IoT Toolkit

Build high-performing, efficient, reliable solutions that run at the network's edge.

[Get the Base Kit](#) + [Get the IoT Kit](#)

Intel® oneAPI Rendering Toolkit

Accelerate High-Fidelity Rendering and Visualization Applications with Powerful Libraries.

[Get the Base Kit](#) + [Get the Render Kit](#)

コマンドプロンプト上でコンパイル
する場合にはHPC Kit のみ
インストールします(当社実施環境).

インテル社製品のダウンロードには、
インテル社のサイトでアカウントを
作成する必要があります。

開発環境の設定例 ① Intel Fortran

2) Intel® oneAPI HPC Toolkit をインストールする.

【補足情報】

- ・ Intel® oneAPI HPC Toolkit を使用するためには、Microsoft社の Visual Studio をインストールする必要があります。
(当社使用バージョン: Microsoft Visual Studio Community 2017)
- ・ インテルソフトウェア開発製品の正規販売代理店であるエクセルソフト株式会社のウェブサイトから評価版をインストールすることも可能です。30日間のサポートサービスを受けられます。

エクセルソフト株式会社のウェブサイト(下図に抜粋)

<https://www.xlsoft.com/jp/products/intel/download/eval.html>

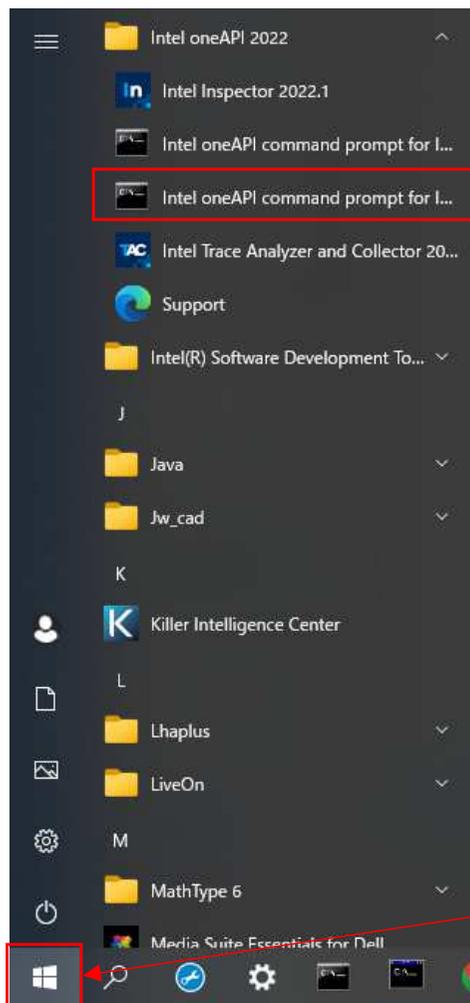
The screenshot shows the Intel Software website interface. At the top, there is a navigation bar with the Intel Software logo and a search bar. Below the navigation bar, there are several menu items: ホーム (Home), 製品 (Products), 購入 (Purchase), 技術情報 (Technical Information), サポート (Support), 評価する (Evaluate), and お問い合わせ (Contact Us). The main content area displays the breadcrumb trail: ホーム > ダウンロード > 評価版申し込み手順. The title of the page is "インテル® ソフトウェア開発製品 評価版/コミュニティ・ライセンスの申し込み手順". Below the title, there is a paragraph of text: "インテル® ソフトウェア開発製品の評価版/コミュニティ・ライセンスのお申し込み手順、および注意事項を説明いたします。必ず事前にご確認ください。". A highlighted box contains the following text: "■ インテル® oneAPI ツールキット評価版サポートサービスに関して
弊社にて、インテル® oneAPI ツールキットの評価版ユーザー様からのお問い合わせを日本語で承っております。評価中にご不明な点がございましたら、こちらのフォームからお問い合わせください。
お問い合わせの際には、本ページの申し込み手順で発行された、ハイフンを含めた 13 桁のシリアル番号をご連絡ください。
インテル® oneAPI ツールキット評価版サポートサービスのご利用期間は、お申し込み後、30日間となっております。お早めにご評価ください。
インテル® oneAPI ツールキット評価版サポートサービスのご利用規約は、こちらのページをご覧ください。
評価版ユーザー様が、インテル社へ直接お問い合わせをご希望の際は、インテル社 Web フォーラム (英語) をご利用ください。"

Below the highlighted box, there are two links: ▶ 評価版 申し込み手順 and ▶ よくあるご質問 (FAQ).

At the bottom of the screenshot, there is a blue banner with the text: 評価版/コミュニティ・ライセンス申し込み手順.

開発環境の設定例 ① Intel Fortran

3) Intel® oneAPI HPC Toolkit をインストール後, Windows 画面左下のスタートボタンから Intel oneAPI command prompt for Intel64 をクリックしてコマンドプロンプトを起動します.



2. Intel oneAPI command prompt for Intel64 をクリックする.
(Intel32 も存在するため注意)

1. Windowsスタートボタン
をクリックする.
(当社実施環境 windows10)

3. コマンドプロンプトが起動する. (x64の表記を確認)

```
Intel(r) oneAPI Tools
:: initializing oneAPI environment...
:: Initializing Visual Studio command-line environment...
Visual Studio version 15.9.22 environment configured.
"C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\
Visual Studio command-line environment initialized for: 'x64'
: compiler -- latest
: debugger -- latest
: dev-utilities -- latest
: inspector -- latest
: itac -- latest
: mpi -- latest
: tbb -- latest
:: oneAPI environment initialized ::
C:\Program Files (x86)\Intel\oneAPI>
```

開発環境の設定例 ① Intel Fortran

- 4) Intel oneAPI command prompt for Intel64 のコマンドプロンプト上で、ユーザーチンのソースコードが格納されているTSS内のフォルダ (¥bin¥x86¥Systemver.9.0.0¥TwinFlow2022oneAPI_user) に移動します。



```
Intel(r) oneAPI Tools
:: initializing oneAPI environment...
:: Initializing Visual Studio command-line environment...
Visual Studio version 15.9.22 environment configured.
"C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\"
Visual Studio command-line environment initialized for: 'x64'
: compiler -- latest
: debugger -- latest
: dev-utilities -- latest
: inspector -- latest
: itac -- latest
: mpi -- latest
: tbb -- latest
:: oneAPI environment initialized ::

C:\Program Files (x86)\Intel\oneAPI>cd C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user>
```

cd の後に、¥TwinFlow2022oneAPI_user までのパス(C:ドライブからの絶対パス)を入力して、キーボードのEnterキーを押すと、カレントディレクトリが ¥TwinFlow2022oneAPI_userに変更されます。

開発環境の設定例 ① Intel Fortran

5) ¥TwinFlow2022oneAPI_user 内に存在するソースコードを用途向きに書き直します(任意のエディタを使用)。

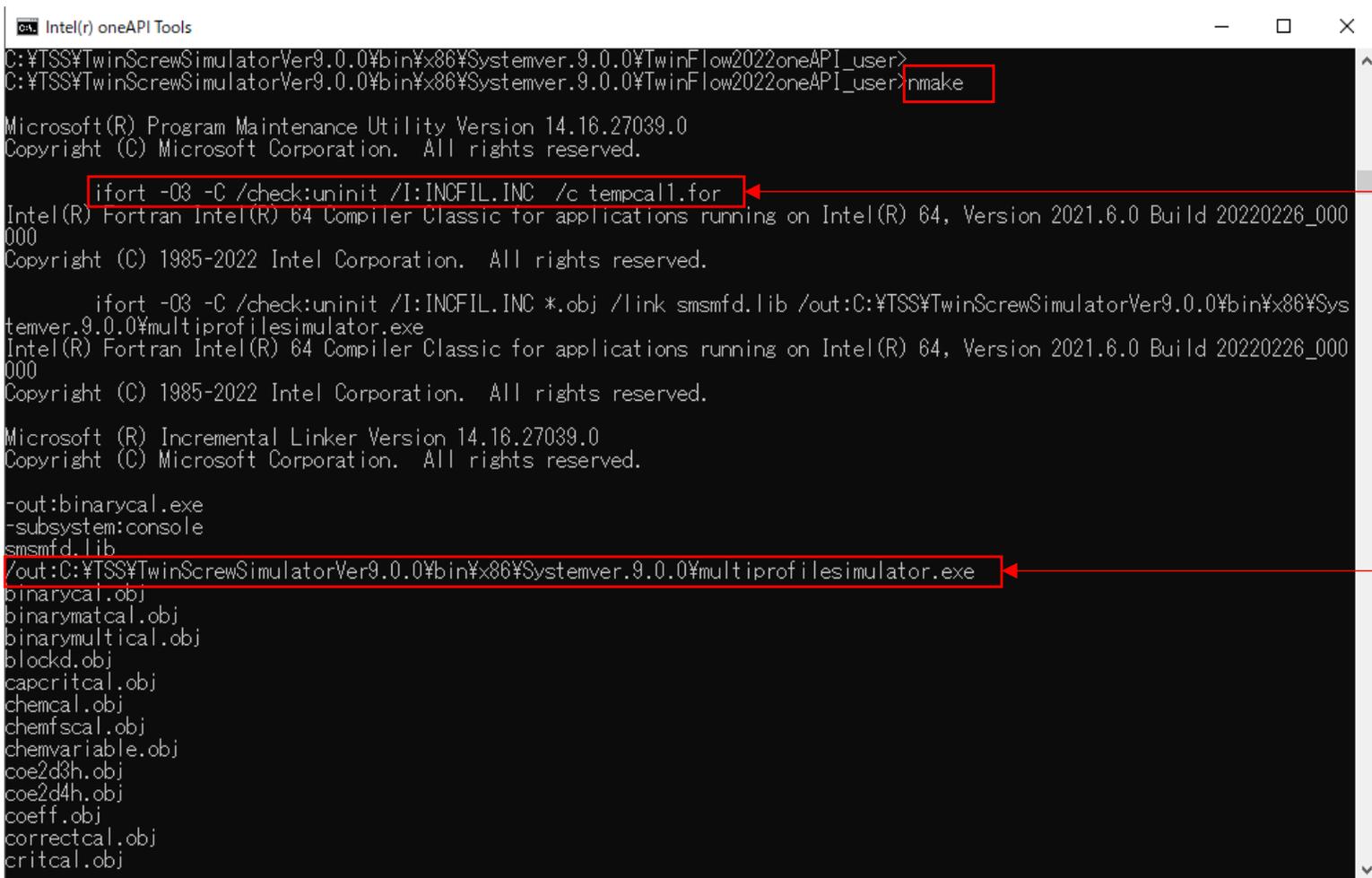
名前	更新日時	種類
Makefile	2022/08/18 13:26	ファイル
smsmfd.dll	2015/05/21 18:25	アプリケーション拡張
smsmfd.lib	2015/05/21 18:25	Object File Library
dalloc.mod	2022/08/18 13:21	MOD ファイル
INCFIL.INC	2022/08/08 15:28	Include File
viscal.for	2022/08/18 13:29	Fortran Source
tempcal1.for	2022/08/08 15:18	Fortran Source
initialsetforchem.for	2022/07/21 15:44	Fortran Source
chemvariable.for	2022/07/21 18:21	Fortran Source
chemfscal.for	2022/07/21 17:03	Fortran Source
vofttraceimplicit.obj	2022/08/18 13:22	3D Object
vofttrace.obj	2022/08/18 13:21	3D Object
viscal.obj	2022/08/18 13:29	3D Object
velgra3d.obj	2022/08/18 13:22	3D Object
unfillcal.obj	2022/08/18 13:21	3D Object
twintosingle.obj	2022/08/18 13:21	3D Object
thicksetstl.obj	2022/08/18 13:22	3D Object
thickset2.obj	2022/08/18 13:22	3D Object

ユーザがカスタマイズ可能な
サブルーチンのソースコード
(ファイル名.for)

(注)ソースコード以外に格納されている
ファイル(.obj など)を変更したり
削除するとコンパイルができなく
なりますので注意して使用ください。

開発環境の設定例 ① Intel Fortran

6) ソースコードを編集後、コマンドプロンプト上で `nmake` と入力してキーボードのEnterキーを押すと、`makefile` を利用したコンパイルが実行されます。



```
Intel(r) oneAPI Tools
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\TwinFlow2022oneAPI_user>nmake

Microsoft(R) Program Maintenance Utility Version 14.16.27039.0
Copyright (C) Microsoft Corporation. All rights reserved.

ifort -O3 -C /check:uninit /I:INCFIL.INC /c tempcall.for
Intel(R) Fortran Intel(R) 64 Compiler Classic for applications running on Intel(R) 64, Version 2021.6.0 Build 20220226_000
Copyright (C) 1985-2022 Intel Corporation. All rights reserved.

ifort -O3 -C /check:uninit /I:INCFIL.INC *.obj /link smsmfd.lib /out:C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Sys
temver.9.0.0\multiprofilesimulator.exe
Intel(R) Fortran Intel(R) 64 Compiler Classic for applications running on Intel(R) 64, Version 2021.6.0 Build 20220226_000
Copyright (C) 1985-2022 Intel Corporation. All rights reserved.

Microsoft (R) Incremental Linker Version 14.16.27039.0
Copyright (C) Microsoft Corporation. All rights reserved.

-out:binarycal.exe
-subsystem:console
smsmfd.lib
/out:C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0\multiprofilesimulator.exe
binarycal.obj
binarymatcal.obj
binarymultical.obj
blockd.obj
capcritical.obj
chemical.obj
chemfscal.obj
chemvariable.obj
coe2d3h.obj
coe2d4h.obj
coeff.obj
correctcal.obj
critical.obj
```

変更したソースコードのみ
コンパイルされます。

コンパイルに成功すると、
¥TwinFlow2022oneAPI_user
フォルダ内に実行プログラム
multiprofilesimulator.exe
が作成されます。

コンパイルに失敗した場合は
エラーメッセージが出力され
るので、ソースコードを修正
後に再度 `nmake` を実施して
ください。

開発環境の設定例 ① Intel Fortran

- 7) ¥TwinFlow2022oneAPI_user フォルダ内の multiprofilesimulator.exe の更新日時がコンパイルした日時に変更されていることを確認後, Systemver.9.0.0フォルダ内に存在する multiprofilesimulator.exe を上書き保存して更新すると, TSSのGUIから解析実行(Analysis/Execute)した際に更新されたmultiprofilesimulator.exe が実行されます。

上書き保存して更新

(参考) 更新前のmultiprofilesimulator.exe を別名でコピーしておく, または別の場所にコピーしておく, 更新前の実行プログラムを残しておくことができます。

TSS(GUI)から解析実行すると, Systemver.9.0.0フォルダ内に存在するmultiprofilesimulator.exe が実行されます。

開発環境の設定例 ② GFortran

1) MinGW (Minimalist GNU for Windows, GUNコンパイラのWindows移植版) をダウンロードする。

入手可能なウェブサイト(下図に抜粋)

<https://sourceforge.net/projects/mingw-w64/files/mingw-w64/>

Home / Browse / Development / Compilers / MinGW-w64 - for 32 and 64 bit Windows / Files

MinGW-w64 - for 32 and 64 bit Windows Files

A complete runtime environment for gcc
Brought to you by: [jon_y](#), [ktietz70](#), [nightstrike](#)

Summary | **Files** | Reviews | Support | Wiki | Mailing Lists | Tickets | News | Discussion | Donate | Code | Git

[Download Latest Version](#)
mingw-w64-v10.0.0.zip (16.0 MB) | [Get Updates](#)

Home / mingw-w64

Name	Modified	Size	Downloads / Week
Parent folder			
mingw-w64-release	2022-04-04		59,900
mingw-w64-testing	2013-09-20		5
mingw-w64-snapshot	2011-09-26		10
Toolchain patches	2010-09-02		6
Totals: 4 Items			59,921

MinGW-W64 Online Installer

- MinGW-W64-install.exe

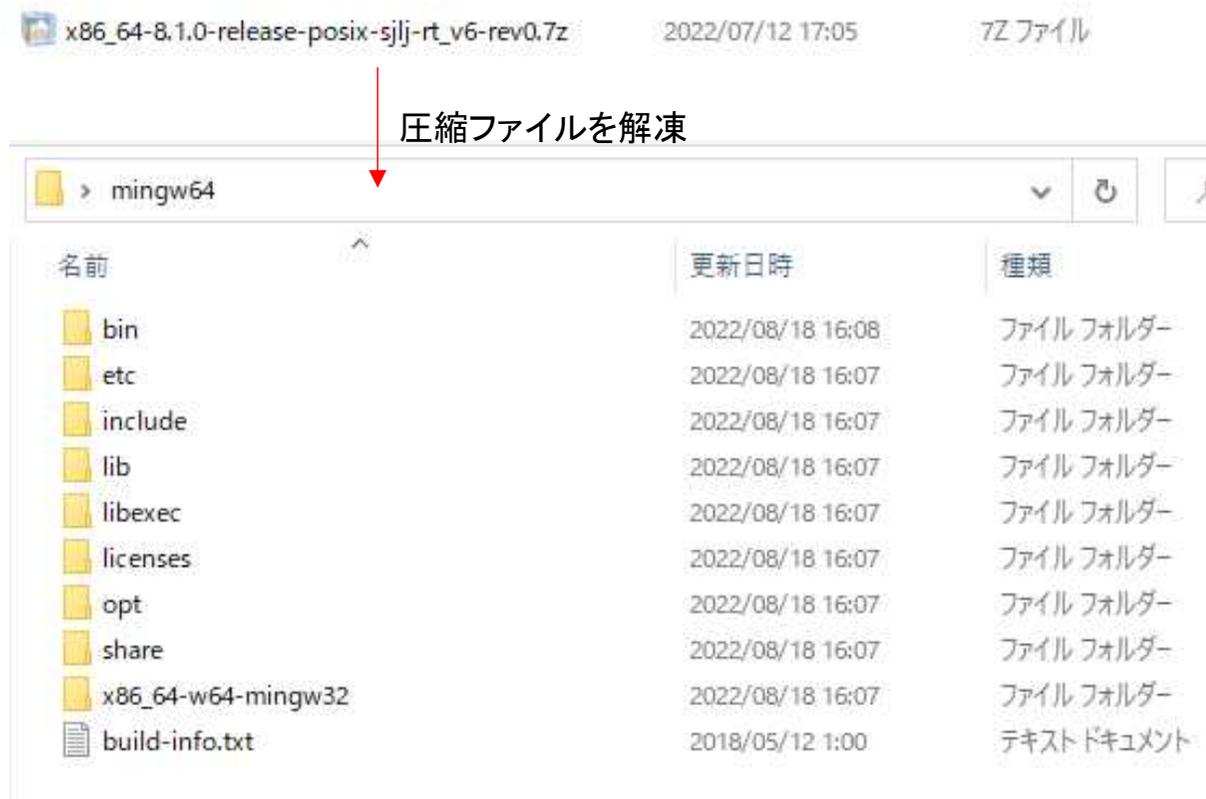
MinGW-W64 GCC-8.1.0

- x86_64-posix-sjlj**
- x86_64-posix-seh
- x86_64-win32-sjlj
- x86_64-win32-seh
- i686-posix-sjlj
- i686-posix-dwarf

64ビットの開発環境(x86_64-posix-sjlj)をダウンロードします。(当社実施環境)

開発環境の設定例 ② GFortran

2) ダウンロードした圧縮ファイル(.7z)を解凍すると, mingw64というフォルダが作成される.



開発環境の設定例 ② GFortran

- 3) mingw64 フォルダを任意の場所に移動し(注: 日本語を含むフォルダ名は不可), Windowsの環境変数の設定から, ¥mingw64¥bin フォルダへの Path を設定する.

環境変数

yorif のユーザー環境変数(U)

変数	値
LIB	C:\Program Files (x86)\Microsoft Visual Studio .NET 2003\SDK\v1.1...
NETGEN_USER_DIR	C:\Users\yorif\AppData\Roaming\Netgen-5.0occ
NETGENDIR	C:\NETGEN_OCC\Netgen-5.0_x64\bin
OneDrive	C:\Users\yorif\OneDrive
OneDriveConsumer	C:\Users\yorif\OneDrive
Path	C:\Users\yorif\AppData\Local\Microsoft\WindowsApps;C:\VI;C:\m...
SMS_LICENSE_FILE	C:\SMSMFLICENSE\MF\license.txt

新規(N)... 編集(E)... 削除(D)

システム環境変数(S)

変数	値
ANS_OLD_ATTACH	1
ComSpec	C:\WINDOWS\system32\cmd.exe
CUDA_PATH	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2
CUDA_PATH_V10_2	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2
DriverData	C:\Windows\System32\Drivers\DriverData
I_MPI_ONEAPI_ROOT	C:\Program Files (x86)\Intel\oneAPI\mpi\2021.6.0
IDB_PATH	C:\Program Files (x86)\Intel\

新規(W)... 編集(O)... 削除(L)

OK キャンセル

環境変数名の編集

%USERPROFILE%\AppData\Local\Microsoft\WindowsApps
C:\VI
C:\mingw64\bin
C:\mingw32\bin

(参考) 当社実施環境での設定例
Cドライブ直下にmingw64を移動

> PC > OS (C:) > mingw64

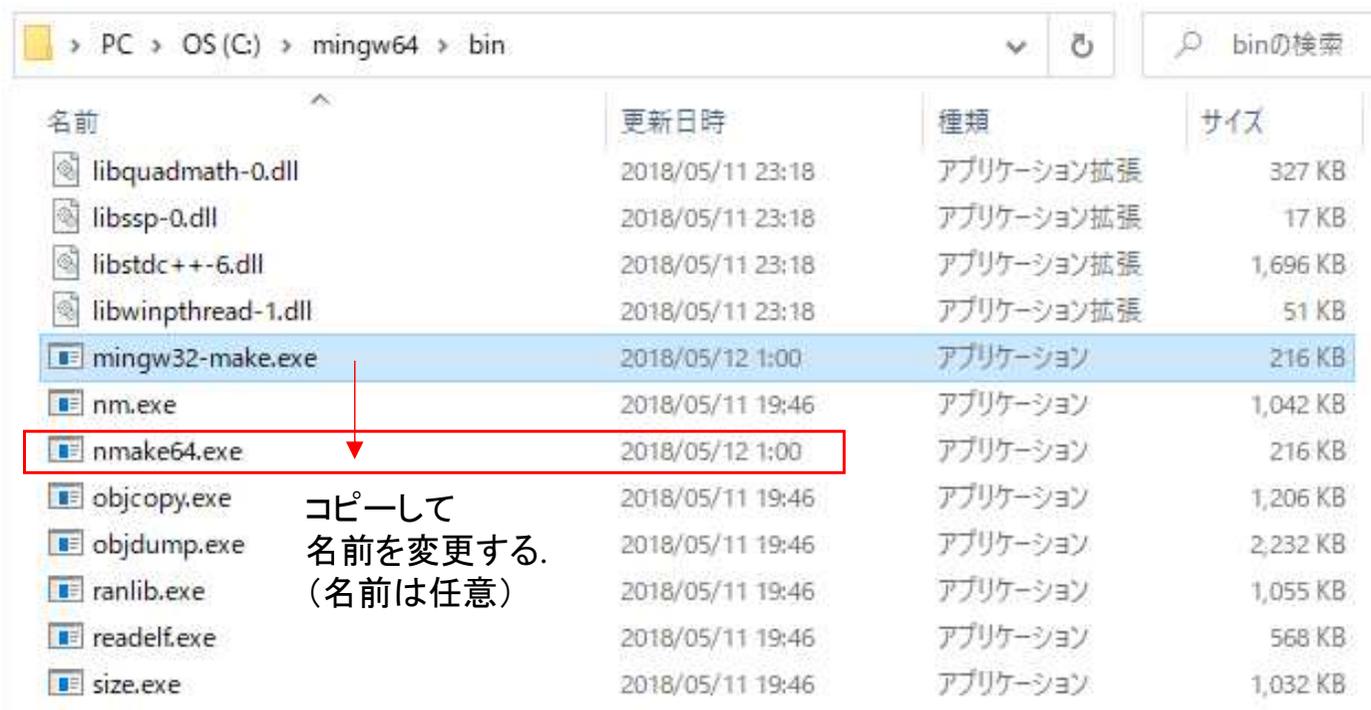
名前	更新日時
bin	2022/07/29 9:59
etc	2022/07/29 9:58
include	2022/07/29 9:58
lib	2022/07/29 9:58

新規(N) 編集(E) 参照(B)... 削除(D) 上へ(U) 下へ(O) テキストの編集(T)...

OK キャンセル

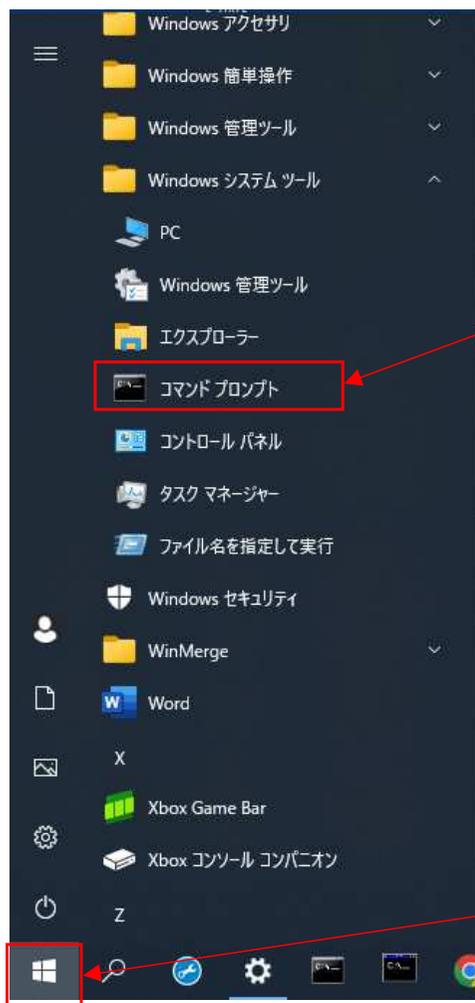
開発環境の設定例 ② GFortran

- 4) ¥mingw64¥bin フォルダ内の makefile を用いたコンパイル実行プログラム mingw32-make.exe をフォルダ内にコピーし, nmake64.exe に名前を変更する. ここまでがMinGwのインストール手順例です.



開発環境の設定例 ② GFortran

5) MinGw をインストール後, Windows 画面左下のスタートボタンから Windows システムツール内のコマンドプロンプトを起動します.



2. コマンドプロンプト
をクリックする.

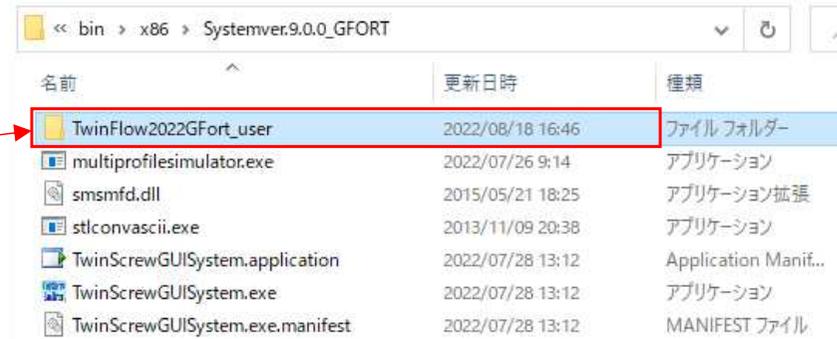
1. Windowsスタートボタン
をクリックする.
(当社実施環境 windows10)

3. コマンドプロンプトが起動する.



開発環境の設定例 ② GFortran

6) コマンドプロンプト上で, GFortran用ユーザーチンのソースコードが格納されているTSS内のフォルダ(¥bin¥x86¥Systemver.9.0.0_GFORT¥TwinFlow2022GFort_user)に移動します.



```
コマンドプロンプト
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.
C:\Users\yori>cd C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>
```

cd の後に, ¥TwinFlow2022GFort_user までのパス(C:ドライブからの絶対パス)を入力して, キーボードのEnterキーを押すと, カレントディレクトリが ¥TwinFlow2022GFort_userに変更されます.

開発環境の設定例 ② GFortran

7) ¥TwinFlow2022GFort_user 内に存在するソースコードを用途向きに書き直します(任意のエディタを使用)。

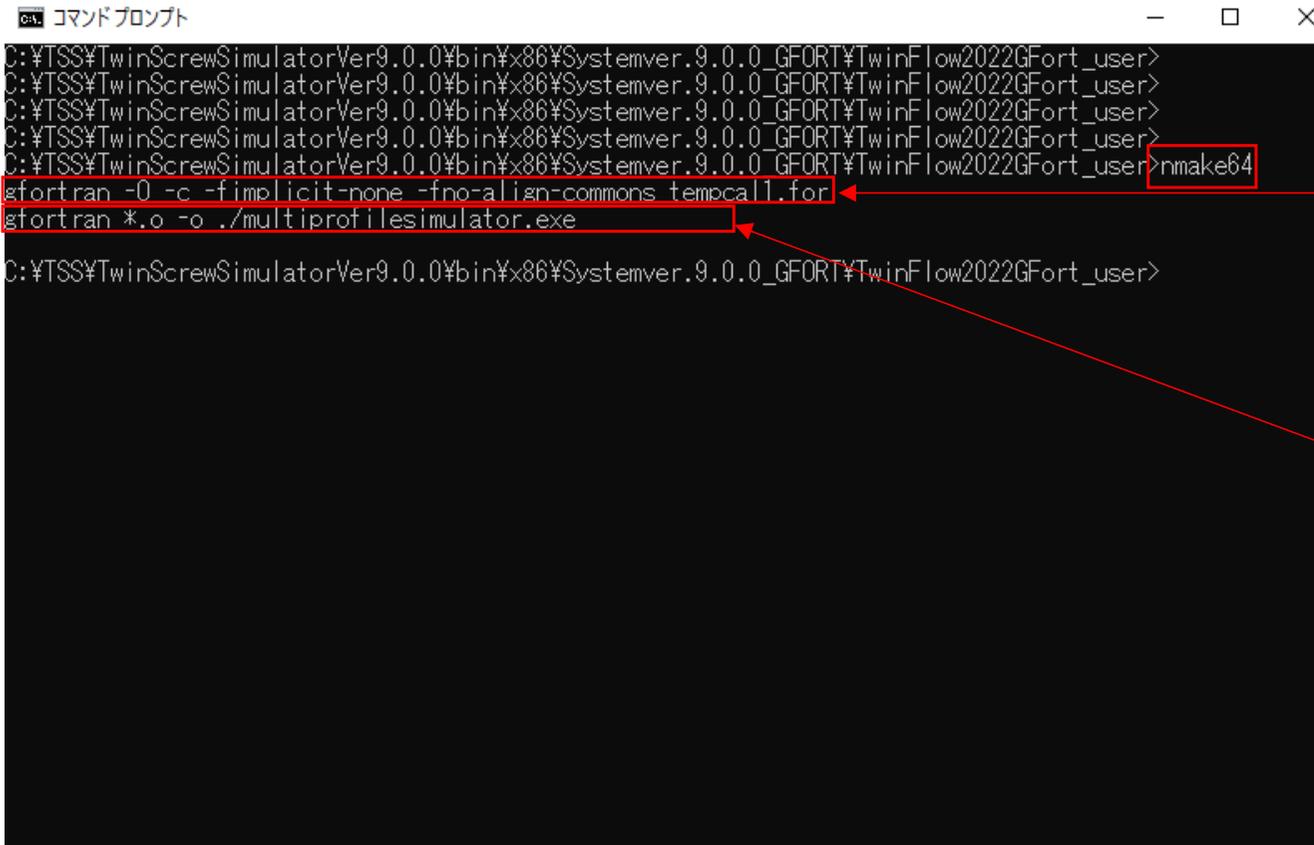
名前	更新日時	種類
chemfscal.for	2022/08/01 10:18	Fortran Source
chemvariable.for	2022/08/01 10:37	Fortran Source
initialsetforchem.for	2022/08/01 10:17	Fortran Source
tempcal1.for	2022/08/01 10:24	Fortran Source
viscal.for	2022/08/18 17:03	Fortran Source
INCFIL.INC	2022/08/01 16:59	Include File
dalloc.mod	2022/08/18 16:56	MOD ファイル
binarycal.o	2022/08/18 16:57	O ファイル
binarymatcal.o	2022/08/18 16:57	O ファイル
binarymultical.o	2022/08/18 16:57	O ファイル
blockd.o	2022/08/18 16:56	O ファイル
capcritical.o	2022/08/18 16:57	O ファイル
chemical.o	2022/08/18 16:57	O ファイル
chemfscal.o	2022/08/18 16:57	O ファイル
chemvariable.o	2022/08/18 16:57	O ファイル
coe2d3h.o	2022/08/18 16:56	O ファイル
coe2d4h.o	2022/08/18 16:56	O ファイル

ユーザがカスタマイズ可能な
サブルーチンのソースコード
(ファイル名.for)

(注)ソースコード以外に格納されている
ファイル(.o など)を変更したり
削除するとコンパイルができなく
なりますので注意して使用ください。

開発環境の設定例 ② GFortran

- 8) ソースコードを編集後, コマンドプロンプト上で nmake64 (p.63 の4) で変更した名前) と入力してキーボードの Enterキーを押すと, makefile を利用したコンパイルが実行されます.



```
コマンドプロンプト
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>nmake64
gfortran -O -c -fimplicit-none -fno-align-commons tempcall.for
gfortran *.o -o ./multiprofilesimulator.exe
C:\TSS\TwinScrewSimulatorVer9.0.0\bin\x86\Systemver.9.0.0_GFORT\TwinFlow2022GFort_user>
```

変更したソースコードのみ
コンパイルされます。

コンパイルに成功すると,
¥TwinFlow2022GFort_user
フォルダ内に実行プログラム
multiprofilesimulator.exe
が作成されます。

コンパイルに失敗した場合は
エラーメッセージが出力される
ので, ソースコードを修正
後に再度 nmake64 を実施して
ください。

開発環境の設定例 ② GFortran

- 9) ¥TwinFlow2022GFort_user フォルダ内の multiprofilesimulator.exe の更新日時がコンパイルした日時に変更されていることを確認後, Systemver.9.0.0_GFORT フォルダ内に存在する multiprofilesimulator.exe を上書き保存して更新すると, TSSのGUIから解析実行(Analysis/Execute)した際に更新された multiprofilesimulator.exe が実行されます。

名前	更新日時	種類
Makefile	2022/08/18 17:10	ファイル
smsmfd.dll	2015/05/21 18:25	アプリケーション拡張
multiprofilesimulator.exe	2022/08/18 17:11	アプリケーション
spooles.lib	2001/06/15 8:01	Object File Library
smsmfd.lib	2015/05/21 18:25	Object File Library
smsamgd.lib	2006/11/28 11:36	Object File Library
voftraceimplicit.o	2022/08/18 16:57	O ファイル
voftrace.o	2022/08/18 16:57	O ファイル
viscal.o	2022/08/18 17:03	O ファイル
velgra3d.o	2022/08/18 16:57	O ファイル
unfillcal.o	2022/08/18 16:57	O ファイル
twintosingle.o	2022/08/18 16:56	O ファイル

名前	更新日時	種類
TwinFlow2022GFort_user	2022/08/18 17:21	ファイル フォルダ
multiprofilesimulator.exe	2022/08/18 17:11	アプリケーション
multiprofilesimulator_buk.exe	2022/08/18 17:03	アプリケーション
smsmfd.dll	2015/05/21 18:25	アプリケーション拡張
stlconvascii.exe	2013/11/09 20:38	アプリケーション
TwinScrewGUISystem.application	2022/08/08 17:40	Application Manif...
TwinScrewGUISystem.exe	2022/08/08 17:40	アプリケーション
TwinScrewGUISystem.exe.manifest	2022/08/08 17:40	MANIFEST ファイル

上書き保存して更新

TSS(GUI)から解析実行すると, Systemver.9.0.0フォルダ内に存在するmultiprofilesimulator.exe が実行されます。

(参考) 更新前のmultiprofilesimulator.exe を別名でコピーしておく, または別の場所にコピーしておく, 更新前の実行プログラムを残しておくことができます。

サンプルプログラム

① 濃度計算

Initialsetforchem の内容

```
c+++++
c+  User define variable number
c+++++
c
c  Number of chemical species
c
c  chemnumber=1 ← 単一成分濃度
c
c  Number of chemical variable
c
c  chemvnumber=0
c
c
c  Number of common variable
c
c  commonvnumber=0
c
c+++++
```

```
c+++++
c  User define initial/boundary condition
c+++++
c  chemname(1)='test sample 1' ← ポストタイトル
c
c  do ie=1,nelem(1)
c  chemc(1,ie)=0.0 } ← 要素濃度の初期設定
c  end do
c
c  do in=1,nnode(1)
c  chemcn(1,in)=0.0 } ← 節点濃度の初期設定
c  end do
c
c  qsumtotal=0.0
c  do in=1,iinletnodenum
c  qsumtotal=qsumtotal+qflux(in,1) } ← 流入口に設定されている全流量の計算
c  end do
c
c  qsumhalf=0.0
c  do in=1,iinletnodenum/2
c  qsumhalf=qsumhalf+qflux(in,1) } ← 流入口境界の
c  end do                                     (in=1~iinletnodenum/2)
c                                             の半領域に設定されて
c                                             いる流量の計算
c
c  do ii=1,iinletnodenum/2
c  chemcn(1,ii)=0.5*qsumtotal/qsumhalf } ← 流入口境界の
c  end do                                     (in=1~iinletnodenum/2)の
c                                             半領域に濃度を設定, 残
c                                             りの半領域の濃度は0
c                                             (当設定によって濃度平均
c                                             値は0.5に漸近)
c+++++
c
```

chemfscal の内容

```
|  subroutine chemfscal
|  use dalloc
|  include 'incfil.inc'
|  c+++++
|  c+  User define left hand side coefficient & right hand source
|  c+++++
|  do ie=1,nelem(1)
|  c
|  chemf(1,ie)=0.0
|  chems(1,ie)=0.0
|  c
|  end do
|  c+++++
|  return
|  stop
|  end
```

$$\mathbf{u} \cdot \nabla f = 0$$



TSSが解析対象とする移流方程式:
 $(chemf(ic, ie) + \mathbf{u} \cdot \nabla) f(ic, ie) = chems(ic, ie)$

$$ic=1, f(1, ie)=chemc(1, ie)$$

$$chemf(1, ie)=0$$

$$chems(1, ie)=0$$

User define model タブメニューの設定

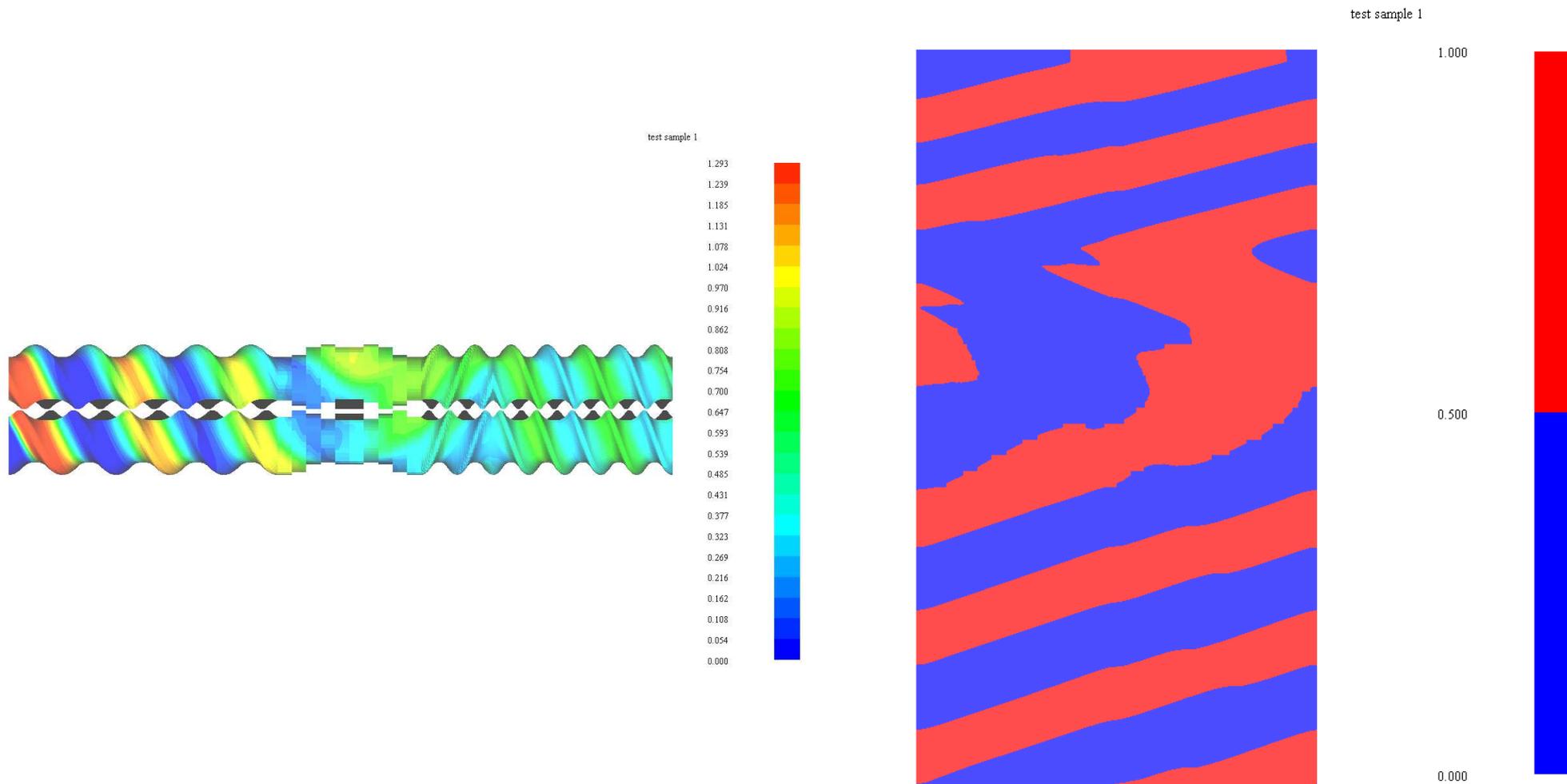
Twin Screw Modeling Thermal Boundary Condition set Analysis User define model Melting / Morphology model

Call user's routine ← ユーザ定義ルーチンのコール

Couple with thermal flow ← 当濃度計算は、熱流動解析に影響を及ぼさないため、非連成(チェックボックスを非チェック)

Calculation parameter

Iteration number	1	← 線形方程式の定常解析では、反復計算を要しないため 1 に設定
Relaxation parameter	1	
Matrix solver iteration number	50000	
Matrix solver convergence	1E-06	



濃度分布予測結果 (プロジェクト名 : testcon)

② 滞留時間分布(RTD)計算

Initialsetforchem の内容

```
c+++++
c User define variable number
c+++++
c
c Number of chemical species
c
chemcnumber=1 ← 単一成分濃度
c
c Number of chemical variable
c
chemvnumber=0
c
c Number of common variable ← 非定常計算刻みとシミュレ
c                               ション時間として使用する
c                               変数の数
c
commonvnumber=2 ←
c+++++
```

各種配列変数のダイナミックアロケーション (ユーザ定義不要の固定構文)

```
c
c Definition of common number variable
c
commonvpar(1)=0.2 ← 時間刻み:0.2 sec
commonvpar(2)=0.0 ← シミュレーション時刻
```

```
c+++++
c User define initial/boundary condition
c+++++
chemcname(1)='test sample 1' ← ポストタイトル
c
do ie=1,nelem(1)
chemc(1,ie)=0.0 } ← 要素濃度の初期設定
end do
c
do in=1,nnode(1)
chemcn(1,in)=0.0 } ← 節点濃度の初期設定
end do
c
do ii=1,iinletnodelist(1)
chemcn(1,ii)=1.0 } ← 流入口に配置された節
                    点に対する境界条件
                    (濃度:1)の設定
end do
c+++++
```

chemfsal の内容

```

subroutine chemfsal
  use dalloc
  include 'incfil.inc'
c+++++
c+ User define left hand side coefficient & right hand source
c+++++
  do ie=1,nelem(1)
c
  chemf(1,ie)=1.0/commonvpar(1)
  chems(1,ie)=chemc(1,ie)/commonvpar(1)
c
  end do
c+++++
  return
  stop
end

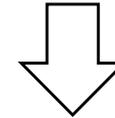
```

$ic=1, f(1,ie)=chemc(1,ie), \Delta t=commonvpar(1)$

$chemf(1,ie)=1/\Delta t=1/commonvpar(1)$

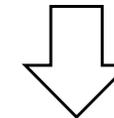
$chems(1,ie)=f^{n-1}/\Delta t=chemc(1,ie)/commonvpar(1)$

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla f = 0$$

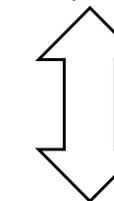


後退差分

$$\frac{f^n - f^{n-1}}{\Delta t} + \mathbf{u} \cdot \nabla f^n = 0$$



$$\left(\frac{1}{\Delta t} + \mathbf{u} \cdot \nabla \right) f^n = \frac{1}{\Delta t} f^{n-1}$$



← 比較

TSSが解析対象とする移流方程式:

$$(chemf(ic,ie) + \mathbf{u} \cdot \nabla) f(ic,ie) = chems(ic,ie)$$

chemvariable の内容

```

subroutine chemvariable
use dalloc
include 'incfil.inc'
volsum=0.0
csum=0.0
iz=axialdiv(1)-2*btlayer-1
ns=iinletnodenum*iz+1
ne=iinletnodenum*(iz+1)
do ie=ns,ne
  volsum=volsum+filleavb(ie)*vol(ie,1)
  csum=csum+filleavb(ie)*vol(ie,1)*chemc(1,ie)
end do
commonvpar(2)=commonvpar(2)+commonvpar(1)
write(lp,*) commonvpar(2),',',csum/volsum
write(*,*) commonvpar(2),',',csum/volsum
return
stop
end
    
```

実行ウィンドウへのシミュレーション時刻と流出口平均濃度のエコープリント

twininf ファイル (機番:lp)へのシミュレーション時刻と流出口平均濃度のエコープリント

流出口で計算される濃度の周方向体積重み付け平均値: $csum/volsum$ の計算

filleavb(ie):要素ieの充填率
 vol(ie,1):要素ieの体積
 chemc(1,ie):要素ieの濃度

$$volsum = \sum_{ie=ns}^{ne} filleavb(ie) * vol(ie,1)$$

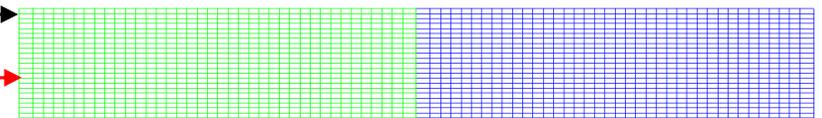
$$csum = \sum_{ie=ns}^{ne} filleavb(ie) * vol(ie,1) * chemc(1,ie)$$

iinletnodenum:周方向の要素/節点分割数

Outlet Boundary Condition	
<input type="radio"/> Pressure fixed	<input type="radio"/> Mass Flux fixed
<input checked="" type="radio"/> Un-fill analysis	<input type="checkbox"/> Old version algorithm
	<input checked="" type="checkbox"/> New version algorithm
Outlet Pressure(MPa)	<input type="text" value="20"/>
Output(kg/h)	<input type="text" value="20"/>
Boundary layer number	<input type="text" value="3"/>

btlayer : Boundary layer number (流出口境界条件の影響を緩和するための境界層数:3)

iz=axialdiv(1):軸方向要素分割数



iz=axialdiv(1)-2*btlayer-1 : 移流方程式の解析対象領域の最大izインデックス

iz=2
 iz=1

User define model タブメニューの設定

Call user's routine ← ユーザ定義ルーチンのコール

Couple with thermal flow ← 滞留時間計算は、熱流動解析に影響を及ぼさないため、非連成(チェックボックスを非チェック)

Calculation parameter

Iteration number	300	← 非定常計算サイクル数
Relaxation parameter	1	
Matrix solver iteration number	50000	
Matrix solver convergence	1E-06	

TSS出力ファイル

(xxx.twininf, xxx:プロジェクト名;kd90)

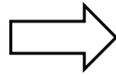
```

kd90.twininf - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
| 0.2000000 , 9.6845416E-25
| 0.4000000 , 7.1414051E-23
| 0.6000000 , 1.7506448E-21
| 0.8000000 , 2.4493692E-20
| 1.0000000 , 2.4700443E-19
| 1.2000000 , 1.9132015E-18
| 1.4000000 , 1.2136444E-17
| 1.6000000 , 6.5644498E-17
| 1.8000000 , 3.1143583E-16
| 2.0000000 , 1.3230550E-15
| 2.2000000 , 5.1119730E-15
| 2.4000000 , 1.8181945E-14
| 2.6000000 , 6.0102883E-14
| 2.8000000 , 1.8664805E-13
| 3.0000000 , 5.4538812E-13
| 3.2000001 , 1.5124221E-12
| 3.4000001 , 3.9931352E-12
| 3.6000001 , 1.0072932E-11
| 3.8000001 , 2.4383362E-11
| 4.0000000 , 5.8789656E-11
| 4.2000000 , 1.2757627E-10
| 4.4000000 , 2.7714095E-10
| 4.6000000 , 5.8332528E-10
| 4.8000000 , 1.1919331F-09
    
```

↑ シミュレーション時刻

↑ 流出口平均濃度

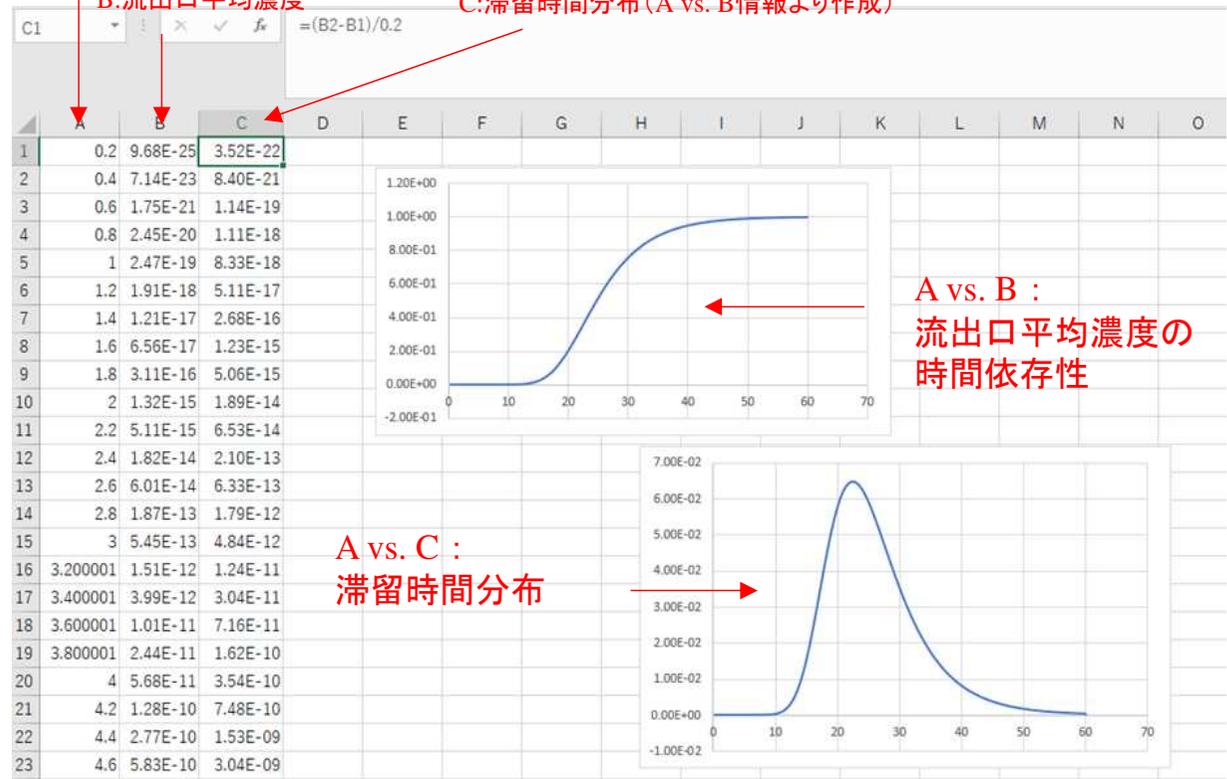
コンマ区切りでエ
クセルに読み込み



A:シミュレーション時刻

B:流出口平均濃度

C:滞留時間分布(A vs. B情報より作成)



A vs. B :
流出口平均濃度の
時間依存性

A vs. C :
滞留時間分布

滞留時間分布情報の作成方法

- 1) Cカラムの1行目に数式=(B2-B1)/0.02を入力してリターンキーを押す. ((B2-B1)/計算時間刻み:Δt=0.02の場合)

	A	B	C	D	E
1	0.2	9.68E-25	=(B2-B1)/0.02		
2	0.4	7.14E-23			
3	0.6	1.75E-21			
4	0.8	2.45E-20			
5	1	2.47E-19			

- 2) Cカラムの1行目をマウスクリックし, マウス右ボタンを押してコピーを選択.

	A	B	C
1	0.2	9.68E-25	3.52E-21
2	0.4	7.14E-23	
3	0.6	1.75E-21	
4	0.8	2.45E-20	
5	1	2.47E-19	
6	1.2	1.91E-18	
7	1.4	1.21E-17	
8	1.6	6.56E-17	

- 3) Cカラムの2行目から末尾行までをマウスドラッグ選択し, 再度マウス右ボタンを押して形式の貼り付けをマウスクリック選択.

	A	B	C	D	E	F
1	0.2	9.68E-25	3.52E-21			
2	0.4	7.14E-23	8.40E-20			
3	0.6	1.75E-21	1.14E-18			
4	0.8	2.45E-20	1.11E-17			
5	1	2.47E-19	8.33E-17			
6	1.2	1.91E-18	5.11E-16			
7	1.4	1.21E-17	2.68E-15			
8	1.6	6.56E-17	1.23E-14			
9	1.8	3.11E-16	5.06E-14			
10	2	1.32E-15	1.89E-13			
11	2.2	5.11E-15	6.53E-13			
12	2.4	1.82E-14	2.10E-12			

③ PP分解反応計算

Initialsetforchemの内容

```

double precision dio,q0o,q1o,q2o,q3o,dm0
C
C+++++
C+  User define variable number
C+++++
C
C  Number of chemical species
C
chemnumber=4
C
C  Number of chemical variable
C
chemvnumber=6
C
C  Number of common variable
C
commonvnumber=2
C
C+++++

```

当ルーチン内で利用する変数の型宣言

```

C+++++
C  User define Initial/Boundary condition set
C+++++
dio=1.0d-03
q0o=1.0081d-02
q1o=1.7857d+04
q2o=1.5136d+11
q3o=2.0*q2o*(2.0*q2o*q0o-q1o**2.0)/q1o/q0o
dm0=4.2D-02

```

化学反応計算の初期値 (境界値) 設定

```

C
chemcname(1)='Peroxide (moles/kg)',
chemcname(2)='Q0 (moles/m3)',
chemcname(3)='Q1 (moles/m3)',
chemcname(4)='Q2 (moles/m3)',

```

解析対象とする移流方程式の数=4

代数的関係式の数=6

```

C
chemvname(1)='Q3 (moles/m3)',
chemvname(2)='Mn (kg/m3)',
chemvname(3)='Mw (kg/m3)',
chemvname(4)='Mz (kg/m3)',
chemvname(5)='Mw/Mn (-)',
chemvname(6)='Mz/Mw (-)',

```

ポスト項目の見出し

```

C
dmn=dm0*q1o/q0o
dmw=dm0*q2o/q1o
dmz=dm0*q3o/q2o

```

各種平均分子量の初期値設定

```

C
commonvpar(1)=dmw
commonvpar(2)=dmz/dmw

```

Viscalと共用する変数定義

Common variable number = 2

```

c
do ie=1,nelem(1)
c
chemc(1,ie)=dio
chemc(2,ie)=q0o
chemc(3,ie)=q1o
chemc(4,ie)=q2o

```

化学種要素変数の初期設定

```

c
chempar(1,ie)=q3o
chempar(2,ie)=dmn
chempar(3,ie)=dmw
chempar(4,ie)=dmz
chempar(5,ie)=chempar(3,ie)/chempar(2,ie)
chempar(6,ie)=chempar(4,ie)/chempar(3,ie)
end do

```

```

c
do in=1,nnode(1)
chemcn(1,in)=dio
chemcn(2,in)=q0o
chemcn(3,in)=q1o
chemcn(4,in)=q2o

```

化学種節点変数の初期設定

```

c
chemparn(1,in)=q3o
chemparn(2,in)=dm0*q1o/q0o
chemparn(3,in)=dm0*q2o/q1o
chemparn(4,in)=dm0*q3o/q2o
chemparn(5,in)=chemparn(3,in)/chemparn(2,in)
chemparn(6,in)=chemparn(4,in)/chemparn(3,in)
c
end do

```

```

c
do ii=1,iinletnodenum
chemcn(1,ii)=dio
chemcn(2,ii)=q0o
chemcn(3,ii)=q1o
chemcn(4,ii)=q2o
end do
c+++++

```

流入口(in=1~iinletnodenum)に
濃度境界値を設定

chemfscalの内容

```

subroutine chemfscal
use dalloc
include 'incfil.inc'
c
double precision dk0,actv,dkd,th
fc=1.0
corg te=200.0+273.15
dk0=1.98d+012
actv=1.4947d+04
corg dkd=dk0*dexp(-actv/te)
c
do ie=1,nelem(1)
c
te=tempe(ie,1)+273.15
dkd=dk0*dexp(-actv/te)
c
Peroxide implicit difference
c
chemf(1,ie)=dkd
chems(1,ie)=0.0
c
th=2.0*fc*dkd*chemc(1,ie)
c
Moment a0,q1,q2 explicit difference
c
q0
chemf(2,ie)=0.0
chems(2,ie)=th*(chemc(3,ie)-3.0*chemc(2,ie))
& / (chemc(3,ie)-chemc(2,ie))
c
q1
chemf(3,ie)=0.0
chems(3,ie)=-2.0*th*chemc(2,ie)
& / (chemc(3,ie)-chemc(2,ie))
c
q2
chemf(4,ie)=0.0
chems(4,ie)=th*(-chempar(1,ie)/3.0
& +chemc(3,ie)/3.0-2.0*chemc(2,ie))
& / (chemc(3,ie)-chemc(2,ie))
c
end do
c
return
stop
end

```

← 当ルーチン内で利用する変数の型宣言

← Initiator efficiency

ペルオキシド分解速度定数
パラメータ

ペルオキシド分解速度定数の計算
tempe(ie,1):要素ieの温度

← $\frac{D[I]}{Dt} = -\kappa_d [I] \Rightarrow \left(\kappa_d + \frac{D}{Dt} \right) [I] = 0$

TSSが解析対象とする移流方程式:

$$\left(chemf(ic,ie) + \frac{D}{Dt} \right) f(ic,ie) = chems(ic,ie)$$

$$\frac{D[I]}{Dt} = -\kappa_d [I] \Rightarrow \left(\kappa_d + \frac{D}{Dt} \right) [I] = 0$$

$$\frac{DQ_0}{Dt} = 2f\kappa_d [I] \left(\frac{Q_1 - 3Q_0}{Q_1 - Q_0} \right)$$

$$\frac{DQ_1}{Dt} = -2f\kappa_d [I] \left(\frac{Q_0}{Q_1 - Q_0} \right)$$

$$\frac{DQ_2}{Dt} = 2f\kappa_d [I] \left(\frac{-\frac{1}{3}Q_3 + \frac{1}{3}Q_1 - 2Q_0}{Q_1 - Q_0} \right)$$

変数対応関係

[I]: chemc(1,ie)
 Q₀: chemc(2,ie)
 Q₁: chemc(3,ie)
 Q₂: chemc(4,ie)

chemvariableの内容

```
subroutine chemvariable
  use dalloc
  include 'incfil.inc'
  double precision q0o,q1o,q2o,dm0 ← 当ルーチン内で利用する変数の型宣言
c
c  dm0=4.2D-02 ← プロピレンの分子量:4.2D-02 kg/m³
c
c  do ie=1,nelem(1)
c
c    q0o=chemc(2,ie)
c    q1o=chemc(3,ie)
c    q2o=chemc(4,ie)
c
c    Closure relation
c
c    q3o=2.0*q2o*(2.0*q2o*q0o-q1o**2.0)/q1o/q0o ← Closure relation
c
c    chempar(1,ie)=q3o
c    chempar(2,ie)=dm0*q1o/q0o
c    chempar(3,ie)=dm0*q2o/q1o
c    chempar(4,ie)=dm0*q3o/q2o
c    chempar(5,ie)=chempar(3,ie)/chempar(2,ie)
c    chempar(6,ie)=chempar(4,ie)/chempar(3,ie)
c
c  end do
c
c  return
c  stop
c  end
```

変数対応関係

Chempar(1,ie) : Q_3
Chempar(2,ie) : 数平均分子量 M_n
Chempar(3,ie) : 重量平均分子量 M_w
Chempar(4,ie) : Z平均分子量 M_z
Chempar(5,ie) : M_w/M_n
Chempar(6,ie) : M_z/M_w

viscalの内容

nonnew : 非ニュートン識別番号

1:Newtonian

2:Power law

3:Log polynomial

4:Carreau

5:Carreau Yasuda

6:Cross

```

c
c*** (6) Cross Model
c
c   if(nonnew.eq.6) then
c
c     eta0=bcoef*exp(tref/(temhi+273.15))
c
c     Molecular weight average eta0 shift proposed by Fukuoka & Min
c
c     eta0=(chempar(3,ie)/commonvpar(1))*3.7*eta0
c
c     MWD shift of shear thinning
c
c     dc=chempar(6,ie)/commonvpar(2)
corg eta=eta0/(1.0+(eta0*gamm/rcoef)**(1.0-dh))
c     eta=eta0/(1.0+dc*(eta0*gamm/rcoef)**(1.0-dh))
c
c     vish(i,ie,ib)=omgc*eta+(1.0-omgc)*vishold
c     end if
c
c     if(temh(i,ie,ib).le.templower) then
c       vish(i,ie,ib)=solidvis
c     end if
c

```

$$\eta_0 = a \exp\left(\frac{T_b}{T+273.15}\right)$$

$a \rightarrow$ bcoef (Material fit パラメータ),
 $T_b \rightarrow$ tref (Material fit パラメータ),
 $T \rightarrow$ temhi: 要素ieの層iの温度

$$\eta_0(t) = \left(\frac{M_w(t)}{M_w(0)}\right)^{3.7} \eta_0(0)$$

$$dc = \frac{M_z(t)/M_w(t)}{M_z(0)/M_w(0)}$$

$$\eta = \frac{\eta_0}{1 + dc * \exp\left(\frac{\eta_0 * gamm}{rcoef}\right)^{1-dh}}$$

chemwriteの内容

```

subroutine chemwrite
use dalloc
include 'incfil.inc'
dimension chemcon(4),chemparameter(6)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccc
open(110,File='CHEMCON',access='sequential',status='unknown',
& recl=220)
open(111,File='CHEMPAR',access='sequential',status='unknown',
& recl=220)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccc
if(chemcnumber.gt.0) then
do in=inletnodenum+1,nnode(1)
do ic=1,chemcnumber
volsum=0.0
csum=0.0
do ii=1,nren(in,1)
ine=nrelem(inrelem(in,1)+ii-1,1)
if(unfillmask.eq.1) then
volsum=volsum+vol(ine,1)*fille(ine,1)
csum=csum+vol(ine,1)*chemc(ic,ine)*fille(ine,1)
else
volsum=volsum+vol(ine,1)
csum=csum+vol(ine,1)*chemc(ic,ine)
end if
end do
chemcn(ic,in)=csum/volsum
end do
end do
end if

```

ローカル変数の定義: 化学種4成分, 代数関係式6成分

出力ファイルオープン

機番110: ファイル名CHEMCON, 化学種4成分出力用

機番111: ファイル名CHEMPAR, 代数関係式6成分

境界節点を除く, 節点番号ループ

化学種ループ

節点inを含む要素数ループ

ine: 節点inを含む要素番号

unfillmask:0 ; 未充填領域も計算量を表示

:1 ; 未充填領域の計算量を非表示

流入境界を除く節点の化学種濃度を体積重み付けの要素濃度より計算

unfillmask:1の場合は, 要素充填率を考慮した体積重み付け平均,

unfillmask:0の場合は, 要素充填率を無視した体積重み付け平均

nren(in,1): 節点inを含む要素数

nrelem(inrelem(in,1)+ii-1,1): 節点inを含むii番目の要素番号

fille(ine,1) : 要素ineの充填率(0 or 1)

fille(ine,1) : 要素ineの充填率(0 or 1)

Unfill Analysis Condition

SOR iteration number

Result display considered with unfill

非チェック: unfillmask=0

チェック: unfillmask=1

c

```
if(chemvnumber.gt.0) then
do in=inletnodenum+1,nnode(1)
do ic=1,chemvnumber
volsum=0.0
csum=0.0
do ii=1,nren(in,1)
ine=nrelem(inrelem(in,1)+ii-1,1)
if(unfillmask.eq.1) then
volsum=volsum+vol(ine,1)*fille(ine,1)
csum=csum+vol(ine,1)*chempar(ic,ine)*fille(ine,1)
else
volsum=volsum+vol(ine,1)
csum=csum+vol(ine,1)*chempar(ic,ine)
end if
end do
chemparn(ic,in)=csum/volsum
end do
end do
end if
```

境界節点を除く, 節点番号ループ

代数関係式数ループ

節点inを含む要素数ループ

ine:節点inを含む要素番号

unfillmask:0 ; 未充満領域も計算量を表示

:1 ; 未充満領域の計算量を非表示

流入境界を除く節点の代数関係式
を体積重み付けの要素濃度より計算

unfillmask:1の場合は, 要素充満率を考慮した体積重み付け平均(未充満領域を非表示),
unfillmask:0の場合は, 要素充満率を無視した体積重み付け平均(未充満領域も表示)


```

cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
  if(chemvnumber.ne.0) then
do iz=0,axialdiv(1)
ntops=iinletnodenum*iz+1
ntope=iinletnodenum*(iz+1)

do ic=1,chemvnumber
chemparameter(ic)=0.0
end do
do ic=1,chemvnumber
countn=0.0
do in=ntops,ntope
countn=countn+1.0
chemparameter(ic)=chemparameter(ic)+chemparn(ic,in)
end do
chemparameter(ic)=chemparameter(ic)/countn
end do
write(111,*) znode(ntops,1)*10.0,',',
&
&
&
&
&
&
&
&
end do
close(111)
end if
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
return
stop
end

```

軸方向分割数ループ
 軸方向分割数iz断面内の開始節点番号
 軸方向分割数iz断面内の終了節点番号

代数関係式 ic の数平均値の計算

軸垂直断面内の代数的関係式平均値の計算

軸方向分割数iz断面のz座標と
 各種数平均代数的関係式
 計算値のファイルCHEMPAR
 への出力

CHEMPARのファイルクローズ

数平均値と代数的関係式対応関係	
chemparameter(1)	: Q_3
chemparameter(2)	: 数平均分子量 M_n
chemparameter(3)	: 重量平均分子量 M_w
chemparameter(4)	: Z平均分子量 M_z
chemparameter(5)	: M_w/M_n
chemparameter(6)	: M_z/M_w

User define model タブメニューの設定

Twin Screw Modeling Thermal Boundary Condition set Analysis User define model Melting / Morophology mo

Call user's routine ← ユーザ定義ルーチンのコール

Couple with thermal flow ← 当化学反応計算は、熱流動解析に影響を及ぼすため、熱流動場との連成解析(チェックボックスをチェック)

Calculation parameter

Iteration number	10	← 非線形反復計算回数
Relaxation parameter	1	
Matrix solver iteration number	20000	
Matrix solver convergence	1E-06	

TSS出力ファイル (CHEMCON, プロジェクト名;testpp)

CHEMCON - 変換

ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

0.000000E+00	9.9999970E-04	1.0080995E-02	17857.00	1.5135993E+11
1.250000	9.9575822E-04	1.0089481E-02	17857.00	1.5064089E+11
2.500000	9.9407567E-04	1.0092848E-02	17857.00	1.5035746E+11
3.750000	9.9229102E-04	1.0096415E-02	17857.00	1.5005814E+11
5.000000	9.9044770E-04	1.0100105E-02	17857.00	1.4975025E+11
6.250000	9.8857167E-04	1.0103855E-02	17857.00	1.4943832E+11
7.500000	9.8667399E-04	1.0107651E-02	17857.00	1.4912424E+11
8.750000	9.8474917E-04	1.0111504E-02	17857.00	1.4880702E+11
10.000000	9.8279375E-04	1.0115415E-02	17857.00	1.4848623E+11
11.250000	9.8082947E-04	1.0119345E-02	17857.00	1.4816535E+11
12.500000	9.7885262E-04	1.0123299E-02	17857.01	1.4784404E+11
13.750000	9.7685563E-04	1.0127296E-02	17857.01	1.4752080E+11
15.000000	9.7483594E-04	1.0131334E-02	17857.01	1.4719548E+11
16.250000	9.7279192E-04	1.0135424E-02	17857.01	1.4686780E+11
17.500000	9.7072008E-04	1.0139567E-02	17857.01	1.4653727E+11
18.750000	9.6861791E-04	1.0143776E-02	17857.02	1.4620348E+11
20.000000	9.6647901E-04	1.0148061E-02	17857.03	1.4586556E+11
21.250000	9.6431089E-04	1.0152399E-02	17857.03	1.4552469E+11
22.500000	9.6212840E-04	1.0156768E-02	17857.03	1.4518326E+11
23.750000	9.5992582E-04	1.0161176E-02	17857.03	1.4484051E+11
25.000000	9.5770007E-04	1.0165630E-02	17857.04	1.4449577E+11
26.250000	9.5547375E-04	1.0170088E-02	17857.05	1.4415267E+11
27.500000	9.5324090E-04	1.0174558E-02	17857.06	1.4381022E+11
28.750000	9.5098739E-04	1.0179064E-02	17857.08	1.4346648E+11
30.000000	9.4871264E-04	1.0183619E-02	17857.08	1.4312117E+11
31.250000	9.4641402E-04	1.0188220E-02	17857.08	1.4277404E+11
32.500000	9.4409013E-04	1.0192869E-02	17857.08	1.4242487E+11

↑ z座標

↑ [I]

↑ Q_0

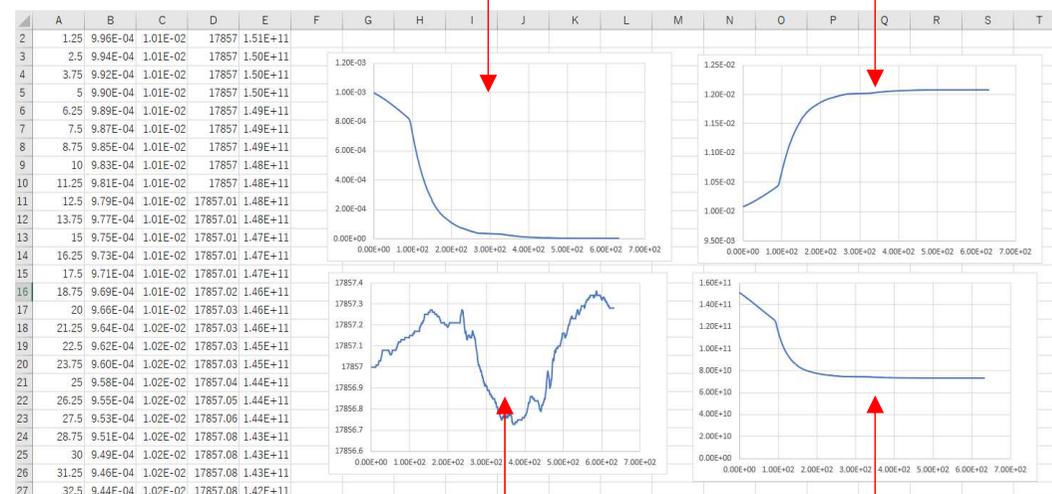
↑ Q_1

↑ Q_2

→ コマ区切りでエクスセルに読み込み

A vs. B :
z座標 vs. ペルオキシド濃度

A vs. C :
z座標 vs. Q_0



A vs. D :
z座標 vs. Q_1

A vs. E :
z座標 vs. Q_2

表1 ユーザが自由にカスタマイズ可能なサブルーチン

ユーザ定義ルーチン名	機能
1) initialsetforchem	化学反応種の初期設定及び解析条件の設定
2) viscal	粘度計算
3) tempcall	1D温度計算
4) chemfscal	化学反応式の左辺及び右辺荷重ベクトルの設定
5) chemvariable	化学反応成分の代数的関係式の計算
6) chemwrite	化学反応解析結果のファイル出力

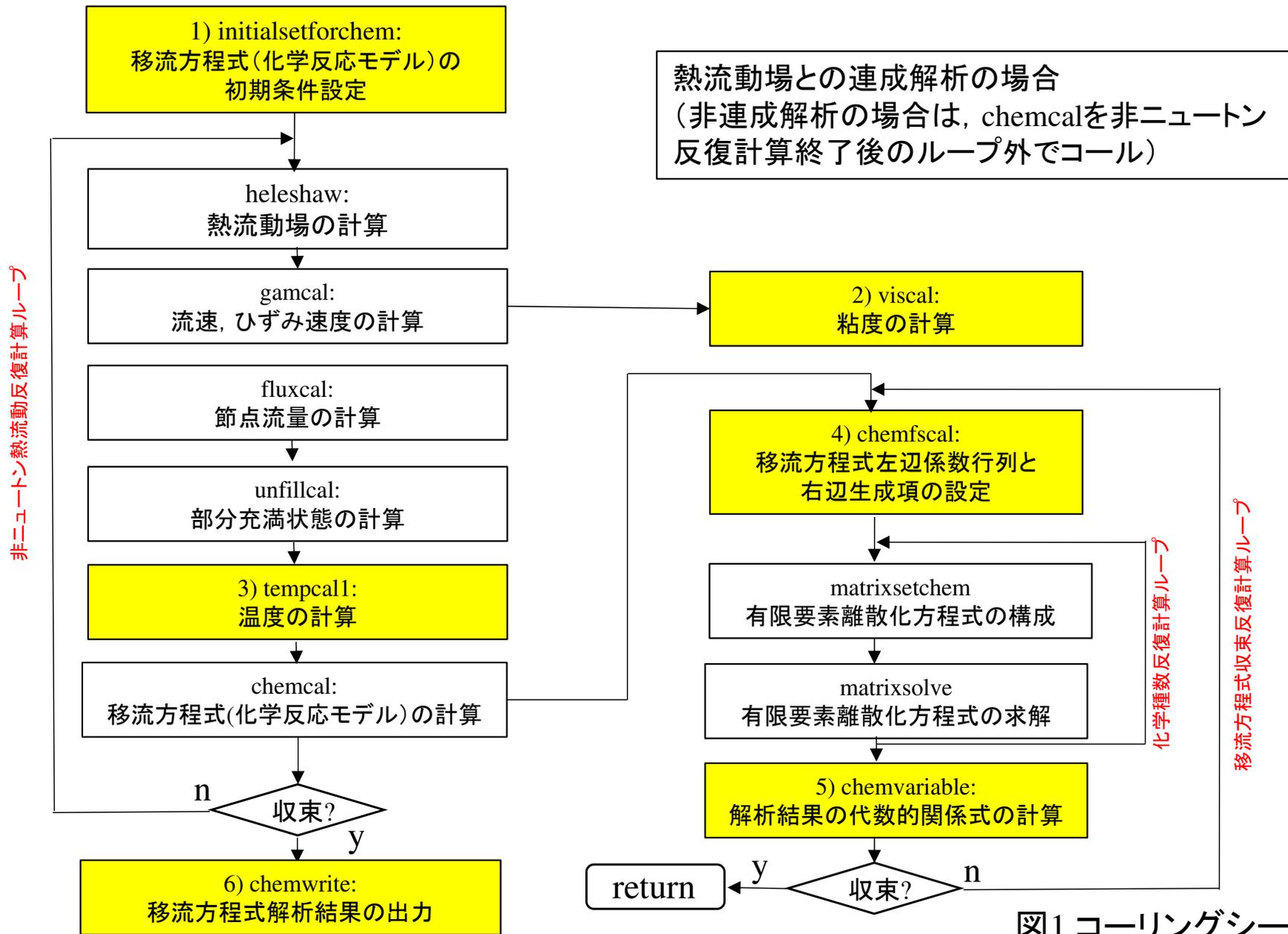


図1 コーリングシーケンス

表2 解析モデル要素情報

変数名	内容
nelem(1)	要素数
iinletnodenum	周方向要素分割数
axialdiv(1)	軸方向要素分割数
inqc(1)	要素構成節点数(inqc(1)=4)
ndiv	肉厚方向の要素分割数
entab(i,j,1)	要素jを構成するi番目の節点番号 (i=1~inqc(1),j=1~nelem(1))
height(i,1)	要素iの肉厚(i=1~nelem(1),単位cm)
vol(i,1)	要素iの体積(i=1~nelem(1),単位cm ³)

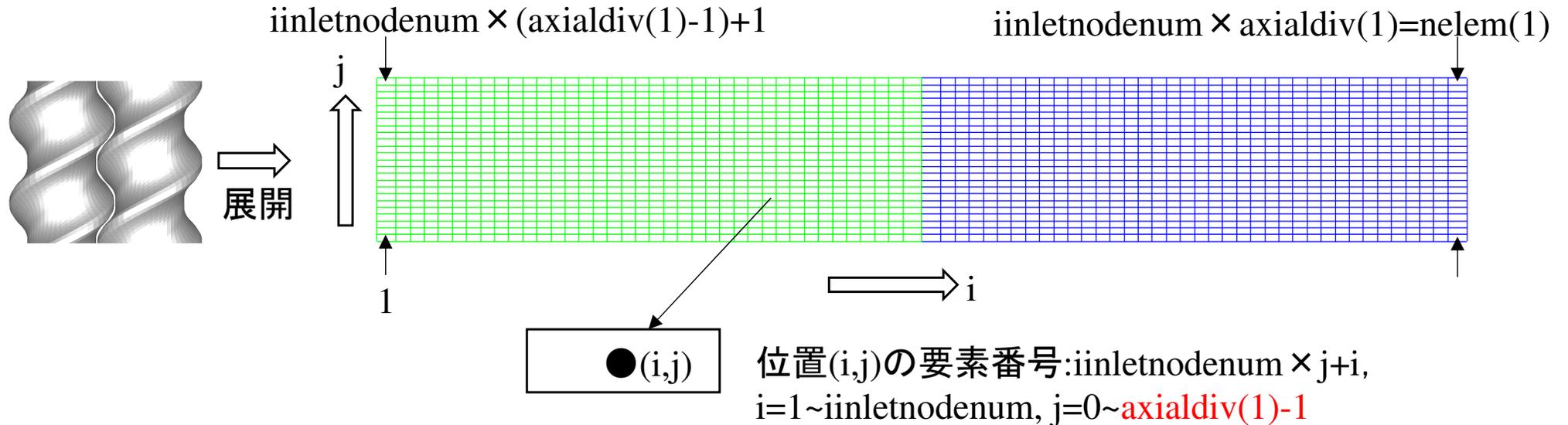


表3 解析モデル節点情報

変数名	内容
nnode(1)	節点数
xnode(i,1),ynode(i,1),znode(i,1)	節点iのx,y,z座標(i=1~nnode(1),単位cm)
nren(in,1)	節点inを含む(隣接する)要素数
nrelem(inrelem(in,1)+ii-1,1)	節点inを含む(隣接する)要素番号 (ii=1~nren(in,1))

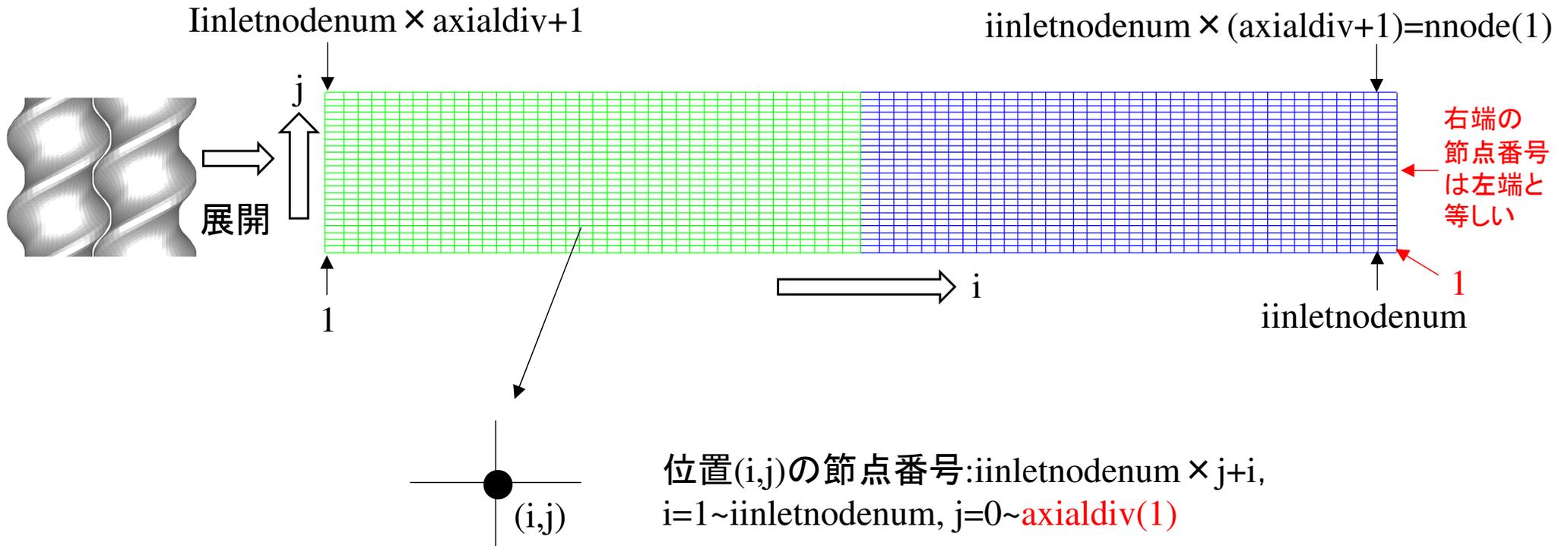


表4 境界条件,成形条件情報

変数名	内容
tempinlet	流入温度(°C)
pprescribedinlet	流入口設定圧力(MPa)
pprescribed	流出口圧力(MPa)
qprescribed	押出量(kg/h)
flowrateinlet	流入口設定流量(cc/s)
rpm	スクリュ回転数(rpm)
href(1,ie,1)	要素ieのスクリュ面熱伝達係数(W/cm ² /K)
tbound(1,ie,1)	要素ieのスクリュ面境界温度(°C)
href(2,ie,1)	要素ieのバレル面熱伝達係数(W/cm ² /K)
tbound(2,ie,1)	要素ieのバレル面境界温度(°C)
iboundt(in,1)	節点inの温度境界条件(0:拘束, 非0:自由)
ibound(in,1)	節点inの圧力境界条件(0:拘束, 非0:自由)

表5 物性情報

変数名	内容
rhoh(i,ie,1)	要素ieの層(i=1~ndiv+1)の密度(g/cm ³)
cph(i,ie,1)	要素ieの層(i=1~ndiv+1)の比熱(J/g/K)
vish(i,ie,1)	要素ieの層(i=1~ndiv+1)の粘度(Pa·s)

i=1 : スクリュ表面, i=ndiv+1 : バレル表面

表6 解析結果要素情報

変数名	内容
temh(i,ie,1)	要素ieの層(i=1~ndiv+1)の温度(°C)
gam(i,ie,1)	要素ieの層(i=1~ndiv+1)のひずみ速度(J/g/K)
uh(i,ie,1),vh(i,ie,1),wh(i,ie,1)	要素ieの層(i=1~ndiv+1)の流速ベクトル成分(cm/s)
tempe(ie,1)	要素ieの温度
fille(ie,1)	要素ieの充満率
filleavb(ie)	要素ieの平均充満率(濃度計算用)

表7 解析結果節点情報

変数名	内容
pres(in,1)	節点inの圧力(Pa)
temp(in,1)	節点inの温度(°C)
visn(in,1)	節点inの粘度(Pa·s)
gamn(in,1)	節点inのひずみ速度(s ⁻¹)

表8 ユーザ定義変数

変数名	内容
chemcnumber	解析対象とする移流方程式の本数(化学種数)
chemvnumber	解析で考慮する配列変数の数
commonvnumber	ルーチン間で共用するスカラー変数の数
chemcname(i)	化学種の名称 (i=1~chemcnumber)
chemvname(i)	配列変数の名称 (i=1~chemvnumber)
chemc(i,ie)	化学種の要素濃度 (i=1~chemcnumber, ie=1~nelem(1))
chemcn(i,in)	化学種の節点濃度 (i=1~chemcnumber, ie=1~nelem(1))
chempar(i,ie)	化学種依存の要素変数 (i=1~chemvnumber, ie=1~nelem(1))
chemparn(i,in)	化学種依存の節点変数 (i=1~chemvnumber, ie=1~nelem(1))
commonvpar(i)	ルーチン間で共用するスカラー変数 (i=1~commonvnumber)