

# 1. はじめに

本書は、Arm Forge (旧 Allinea Forge) を東京工業大学学術国際情報センターのTSUBAMEで利用する方法について説明しています。また、TSUBAME3を利用するにあたっては、[TSUBAME利用の手引き](#)もご覧下さい。サーバの利用環境や注意事項などが詳細に記述されていますので、よく読んでください。

<https://www.arm.com/>

## 1.1. 利用できるバージョン

TSUBAME3で利用可能な最新バージョンについてはTSUBAME計算サービスWebサイトの[アプリケーション](#)ページをご確認下さい。

研究に支障がない限り、バグ修正の入っている最新版をご利用下さい。

### 1.1.1. バージョンの切り替え

module コマンドでmodule ファイルを読み込むことでバージョンの切り替えが可能です。

[利用環境の切り換え方法](#)で切り替えが可能です。

読み込めるバージョンについては[サポートされているアプリケーション](#)をご確認下さい。

コマンド例

```
# Ver 18.0.1を使用する場合
$ module load forge/18.0.1

# Ver 7.0.5を使用する場合
$ module load forge/7.0.5
```

## 1.2. 概要

Arm Forgeは、MPI、UPC、CUDA、OpenMPなどの幅広い並列アーキテクチャとモデルをサポートする並列デバッガ (Arm DDT) とMPI, OpenMPおよびスカラープログラム用

の低オーバーヘッドのラインレベルプロファイラ(Arm MAP)を有するツールです。Arm DDT はパワフルなデバッグ機能を有します。特長は次の通りです。

- Single process and multithreaded software
- OpenMPI
- Parallel (MPI) software
- Heterogeneous software (GPUなど)
- Hybrid codes mixing paradigms ( MPI + OpenMP、MPI + CUDA)
- Multi-process software (client-server applications)

Arm MAP はパラレルプロファイラツールです。特長は次の通りです。

- MPI, OpenMP and single-threaded programsのサポート
- 実行の時間に関わらず、採取されるデータは小さい
- 数千のMPIプロセスでもわずか5%のアプリケーション速度低下
- バッチモードとインタラクティブモードの動作をサポート
- メモリ使用量、浮動小数点計算、およびMPI使用率の採取。また、AVX拡張を含むベクトル化されたSIMD命令のパーセンテージ コードの各部分で採取。

## 1.3. マニュアル

マニュアルは、インストールディレクトリ内に PDF ドキュメントとして格納されています。各バージョンごとのドキュメント格納ディレクトリは次の通りです。

Ver 18.0.1

```
/apps/t3/sles12sp2/isv/allinea/forge/18.0.1/doc/userguide-forge.pdf
```

Ver 7.0.5

```
/apps/t3/sles12sp2/isv/allinea/forge/7.0.5/doc/userguide-forge.pdf
```

## 2. 利用方法

### 2.1. Arm Forge の実行

#### 2.1.1. TSUBAME3にログイン

[ログイン方法](#)を参考にログインノードにログイン後、[インタラクティブノード](#)を利用したX転送を参考にノードをX転送付きで確保して下さい。以下以降の例では、全て計算ノードにログインした状態で行います。

#### 2.1.2. Arm DDTのGUIの起動

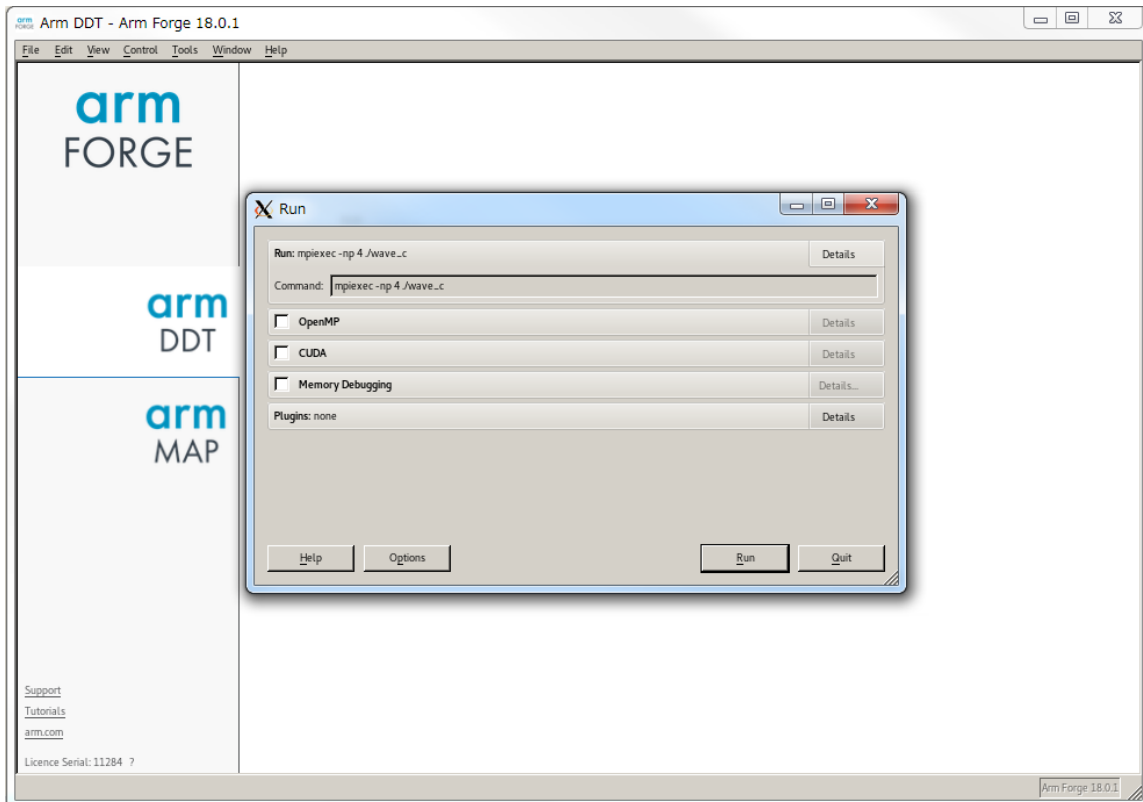
最初に、デバックする実行モジュールを作成します。以下、Ver 18.0.1におけるIntel Compiler、Intel MPIでMPI 4並列の実行例になります。

```
$ module load cuda
$ module load intel intel-mpi
$ cp /apps/t3/sles12sp2/isv/allinea/forge/18.0.1/examples/wave.c .
$ mpicc -g -fno-inline-functions -fno-omit-frame-pointer -O3
wave.c -o wave_c -lm -lrt
```

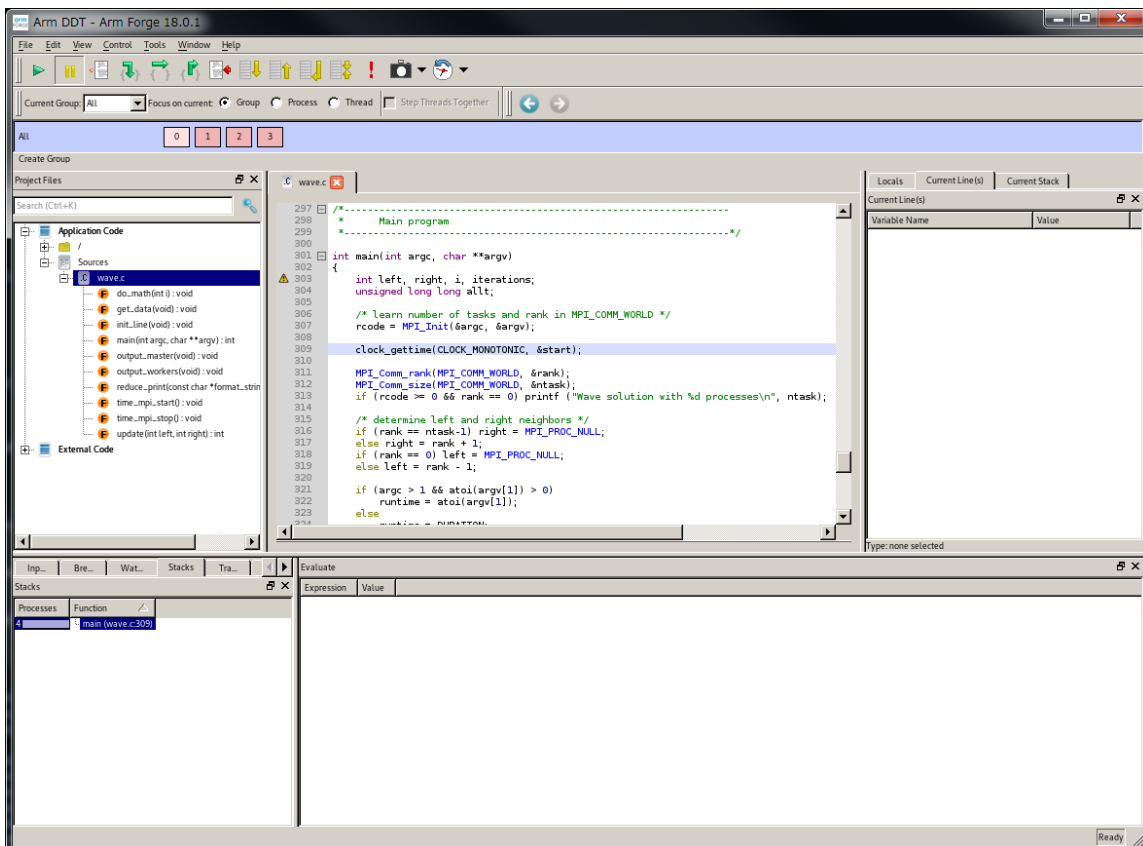
ここでは、-gのオプションを付加します。wave\_cという名前の実行モジュールが出来ているはずです。

```
$ module load forge/18.0.1
$ module load cuda
$ module load intel intel-mpi
$ ddt mpiexec -np 4 ./wave_c
```

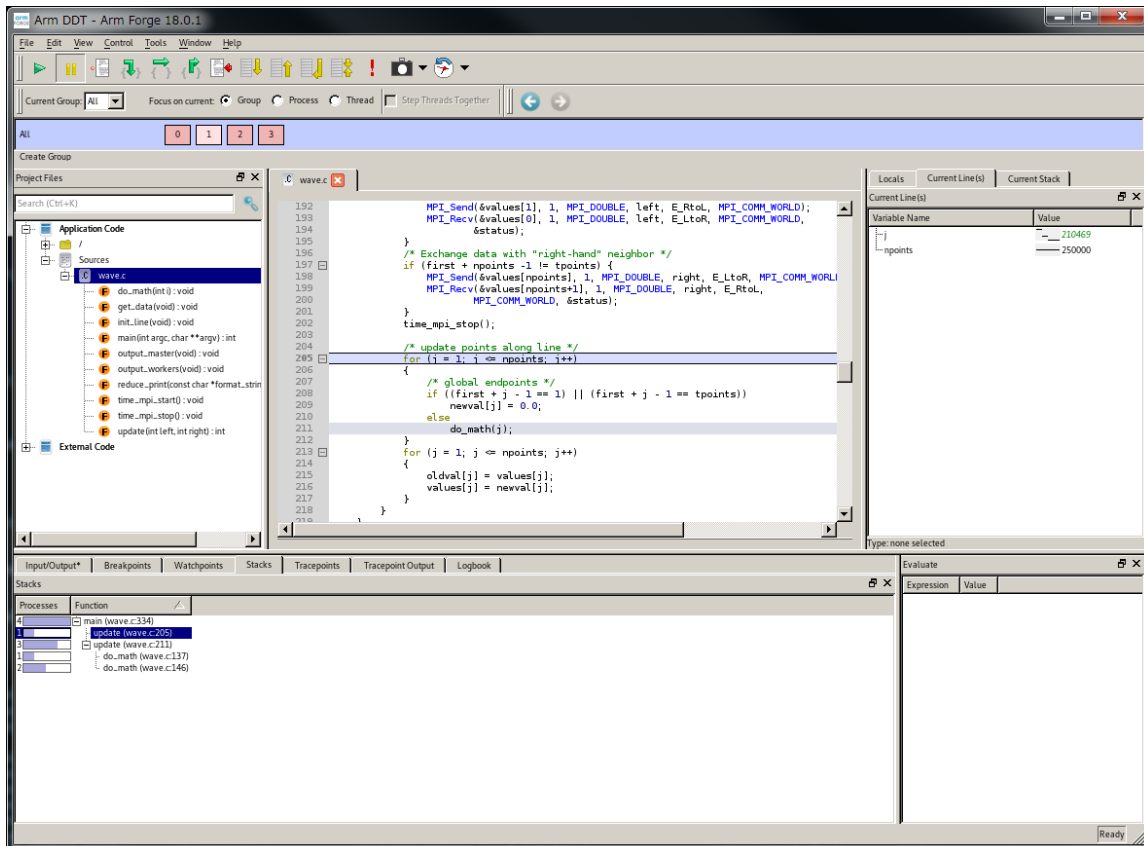
これにより、以下のような起動画面が表示されます。(プログラムを指定していないと選択画面が出ます)



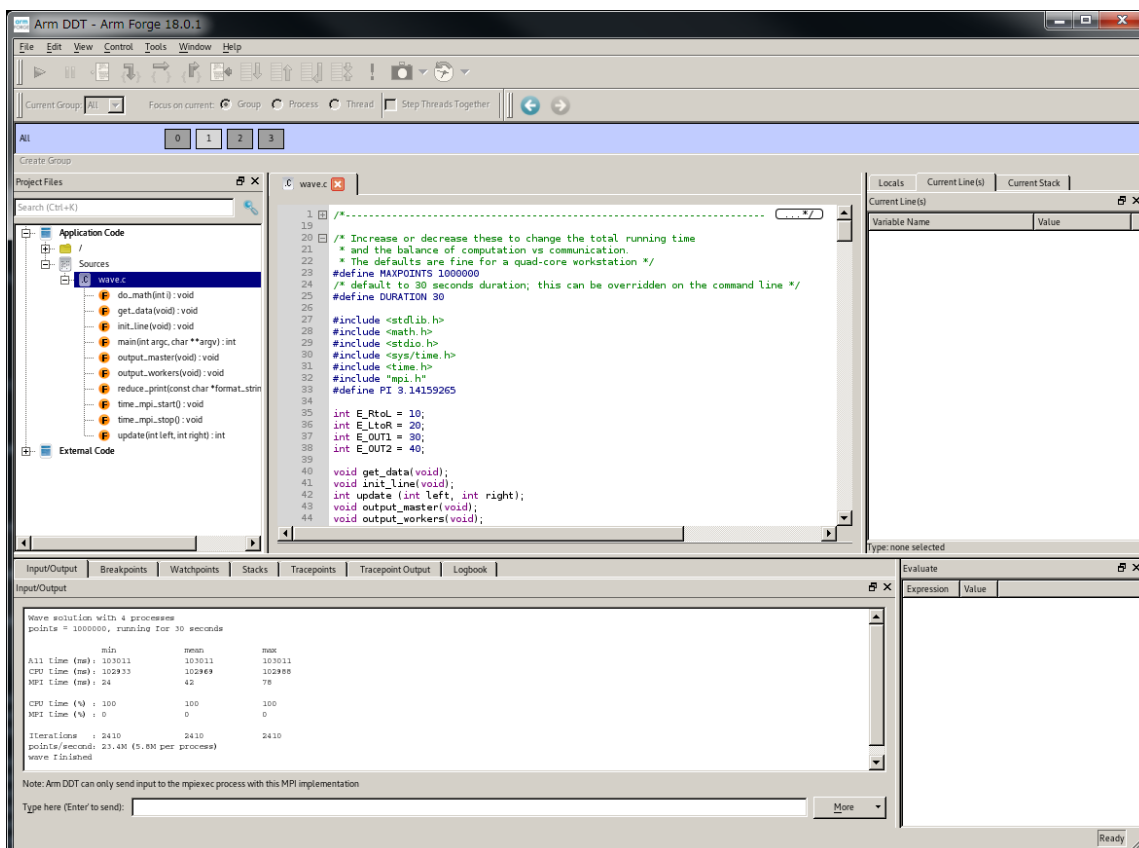
[Run]をクリックすると開始画面が表示されます。



ブレークポイントなどを設定し[Pause][Run]をクリックすると途中経過、値が参照可能です。



プログラムが完走するとInput/Outputに結果などが表示されます。



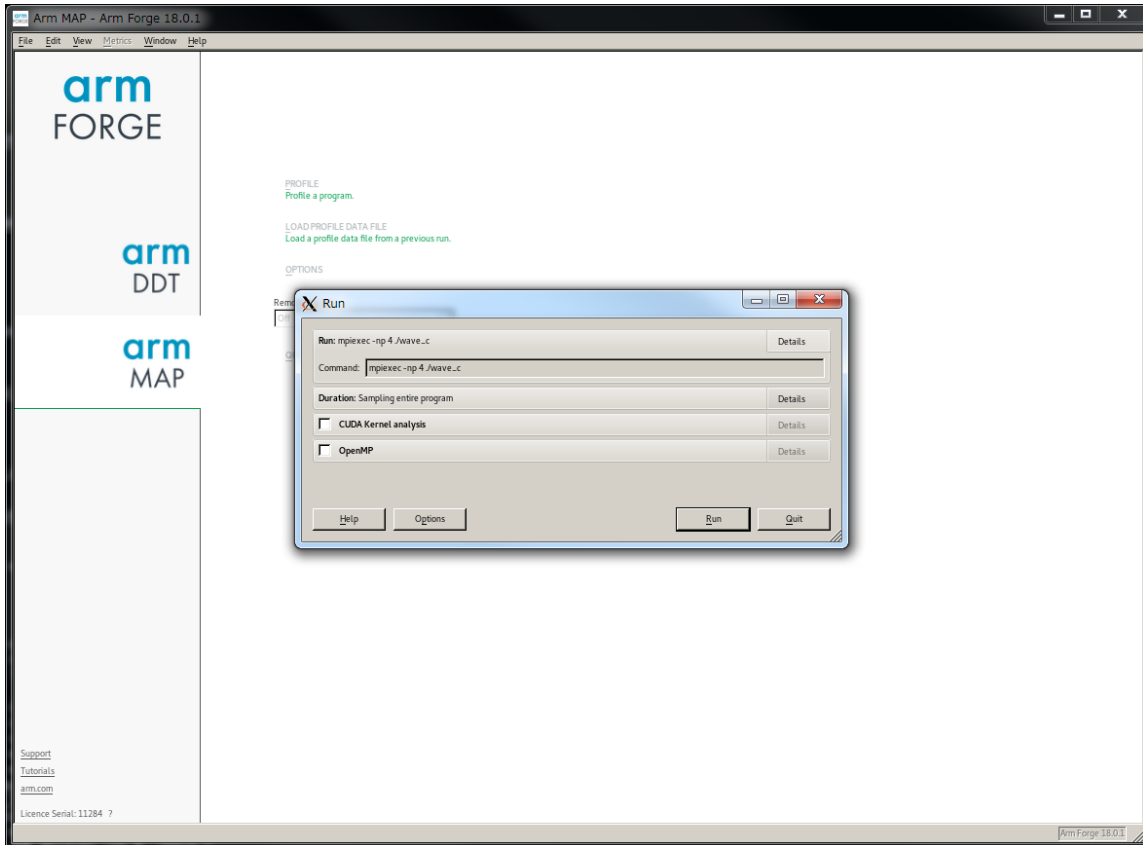
### 2.1.3. Arm MAPのGUIの起動

以下のようにArm MAPを起動します。

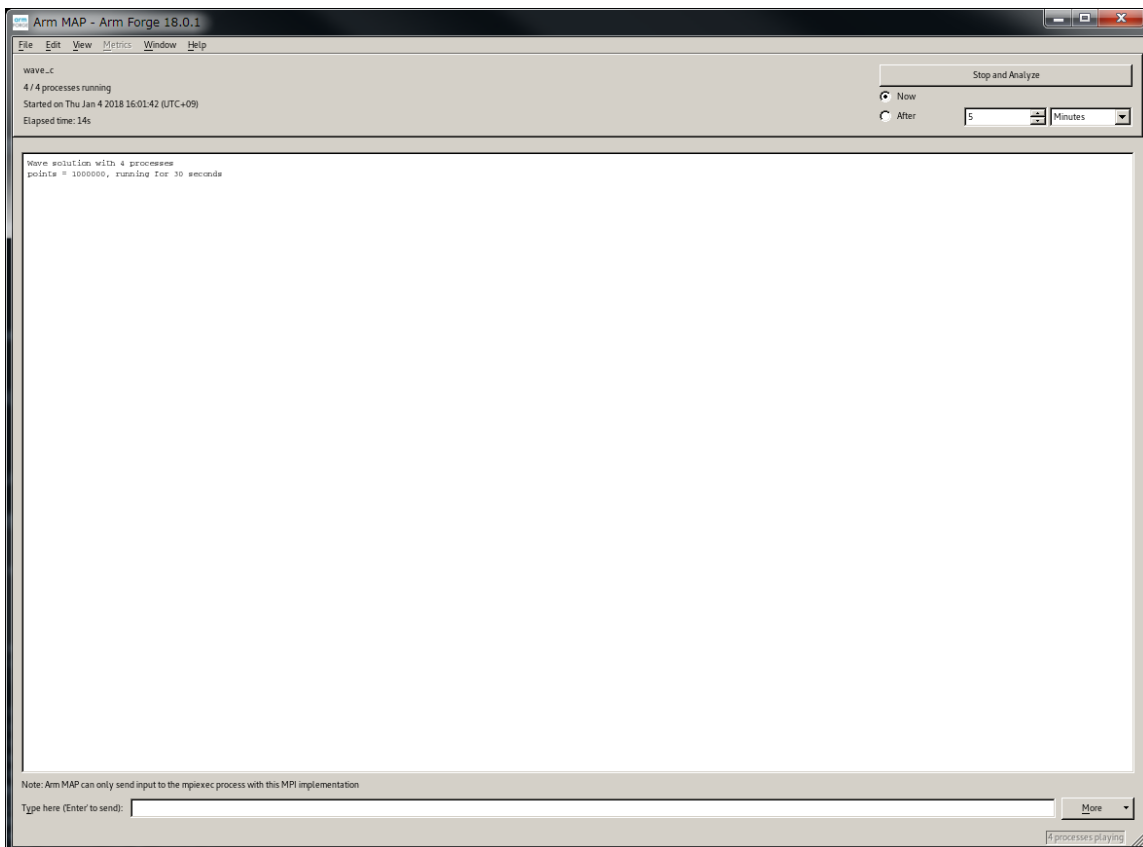
```

$ module load forge/18.0.1
$ module load cuda
$ module load intel intel-mpi
$ map mpiexec -np 4 ./wave_c
    
```

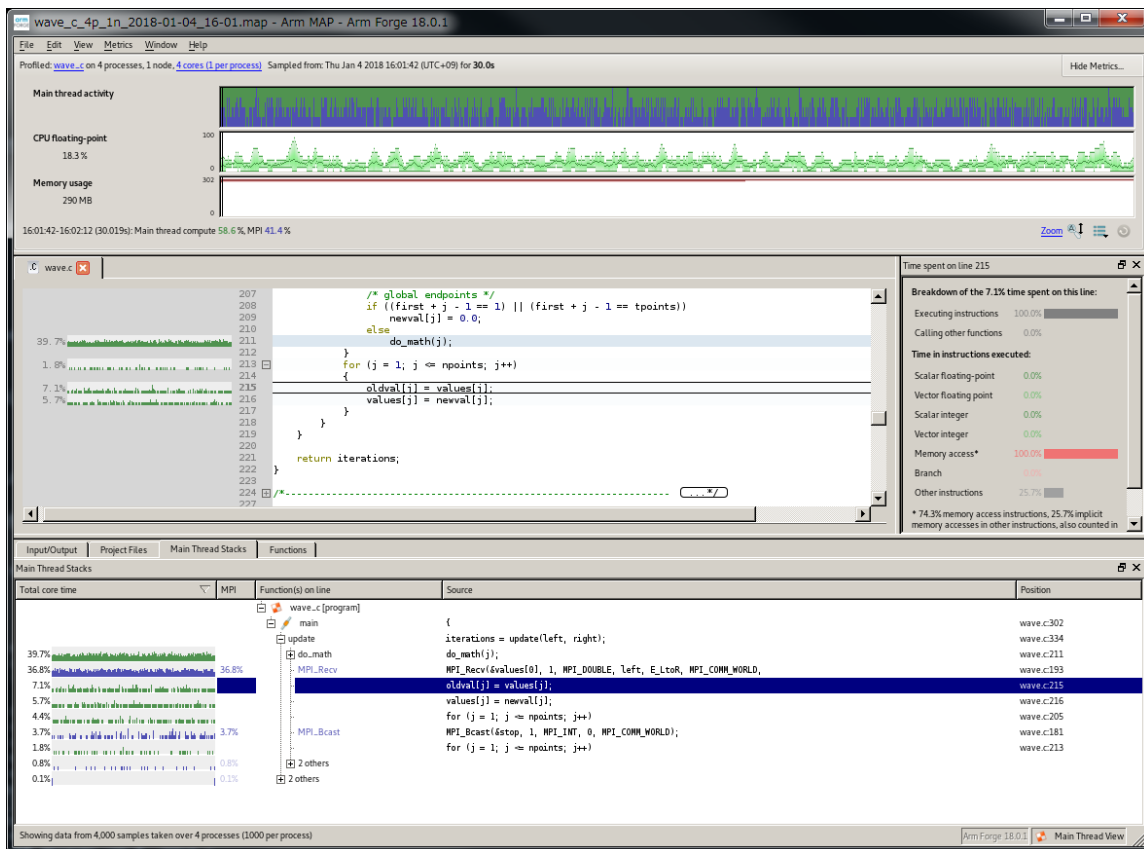
これにより、以下のような起動画面が表示されます。(プログラムを指定していないと選択画面が出ます)



[Run]をクリックすると実行開始されます。

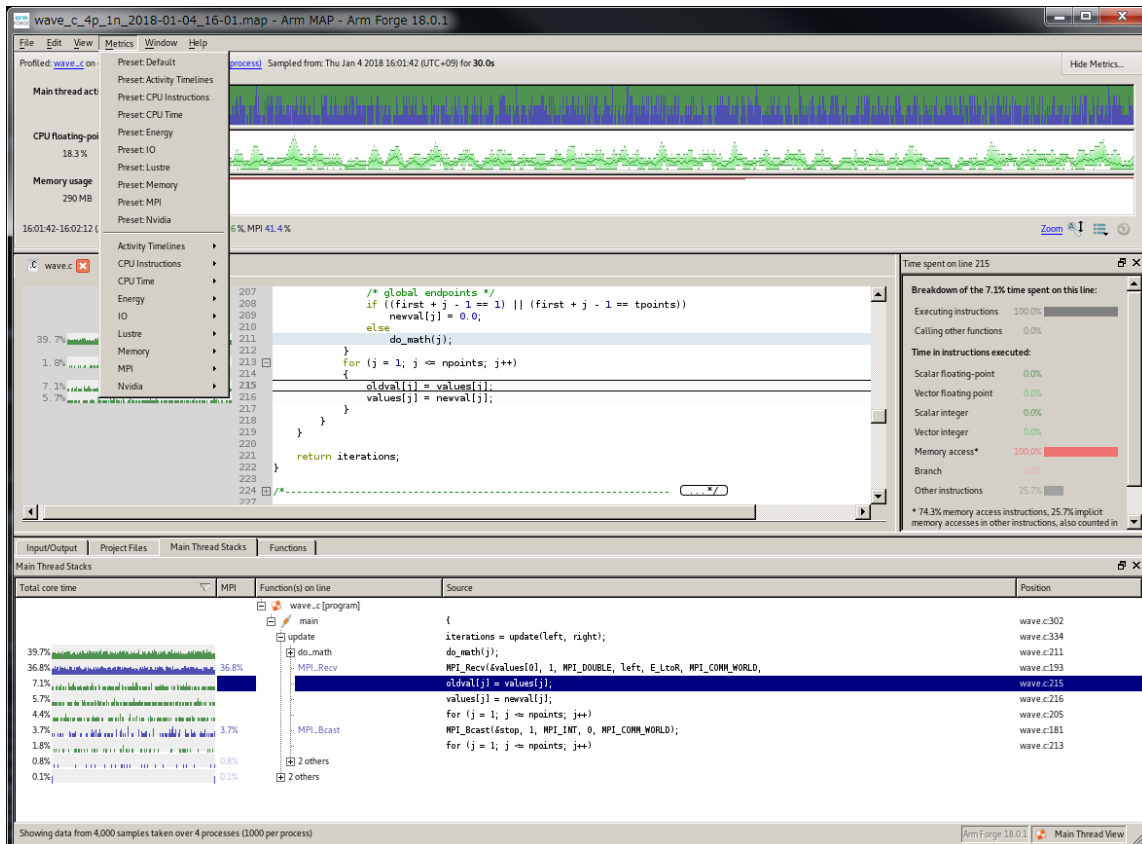


適当なタイミングなどで[Stop and Analysys]をクリックすると結果が表示されます。



[Metrics]をクリックすると各種データを参照可能です。





## 2.2. ライセンス使用状況の確認

webブラウザで次のURLにアクセスすることにより、ライセンス利用状況を確認できます。

<http://lice0.int.t3.gsic.titech.ac.jp:4241/status.html>

ログインノード上でもライセンス利用状況を確認できます。

```
w3m -dump http://lice0:4241/status.html
```

## 改訂履歴

改定日付	内容
2019/07/31	mkdocs版作成
2018/10/26	URLと文言を修正
2018/01/03	アプリケーションの呼称をAllinea ForgeからArm Forgeに変更 Arm Forge 18.0.1の使用方法を追記
2017/11/17	URLを修正 URLリンク切れのため削除
2017/09/15	初版作成