

実践編 (後)

陰山

はじめに

2 次元並列

ハイブリッド
並列化

計算科学演習

実践編 (後): MPI+OpenMP によるハイブリッド並列化

陰山 聡

神戸大学大学院 システム情報学研究科 計算科学専攻

2012.07.19

3 週分の内容

実践編（後）

陰山

はじめに

2 次元並列

ハイブリッド
並列化

1 実践編（前半）

シミュレーションプログラミング（陰山）

2 可視化

gnuplot による 1 次元可視化と 2 次元可視化（政田）

3 実践編（後半）

OpenMP + MPI によるハイブリッド並列化（陰山）

今日の演習で使うコード

実践編（後）

陰山

はじめに

2 次元並列

ハイブリッド
並列化

- 1 マシン「scalar」
- 2 `cp -r /tmp/120719` 自分のディレクトリ

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

2次元並列

1次元並列

実践編（後）

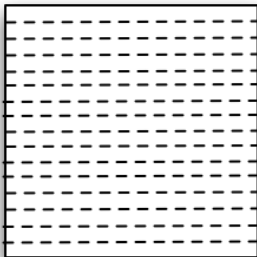
陰山

はじめに

2次元並列

ハイブリッド
並列化

引き続き、正方形領域の熱伝導問題（平衡温度分布）を考える。これまでの並列化：1次元領域分割による並列化：16並列



2次元並列

実践編（後）

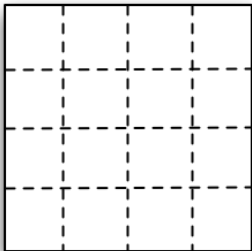
陰山

はじめに

2次元並列

ハイブリッド
並列化

これも 16 並列。どちらが速いか？ 1次元領域分割と 2次元領域分割どちらを採用すべきか？



そもそも1次元分割ができない場合

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

- 格子数 NGRID 61
- 並列プロセス数 100
 - 1次元分割不可能
 - 2次元分割なら可能（総格子点数 3721）

1次元分割と2次元分割のちがい

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

格子点数 NGRID が十分大きければ 1次元分割と2次元分割は同じか？

- プロセスあたりの計算量は同じ
- 通信量が違う

計算と通信

実践編（後）

陰山

はじめに

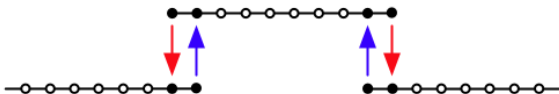
2次元並列

ハイブリッド
並列化

1次元空間を格子点で離散化した上で、MPIでプロセス間通信を行う場合を考える。

計算用格子点（白丸）：6個

通信用格子点（黒丸）：4個



計算と通信

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

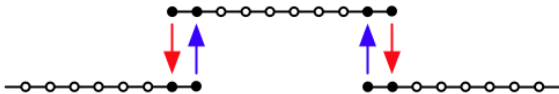
計算格子には2種類ある。

- 1 その上で計算だけを行う格子。
- 2 MPI通信のデータを送ったり、受けたりする格子である。

（一番外側から2番目の格子は計算も通信も行う。）

計算用格子点（白丸）：6個

通信用格子点（黒丸）：4個



通信は時間がかかるので、通信を行う格子点は少ないほうが望ましい。

2次元領域分割

実践編（後）

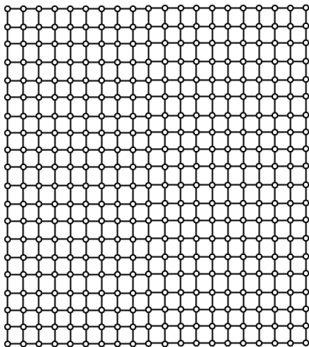
陰山

はじめに

2次元並列

ハイブリッド
並列化

正方形領域を 400 個の格子点で離散化した場合



2次元領域分割

実践編（後）

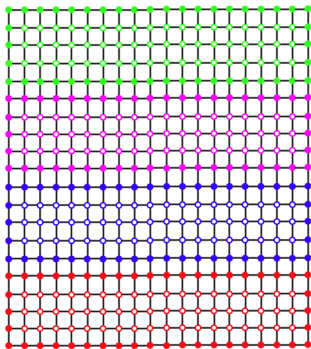
陰山

はじめに

2次元並列

ハイブリッド
並列化

4つのMPIプロセスで並列化。1次元領域分割。



赤のプロセスの通信担当格子点の数：46

2次元領域分割

実践編（後）

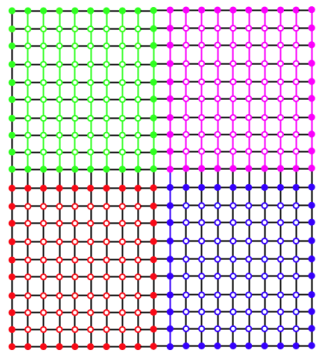
陰山

はじめに

2次元並列

ハイブリッド
並列化

2次元領域分割の場合、同じく4つのMPIプロセスで並列化。



赤のプロセスの通信担当格子点の数：38

2次元領域分割の方法

実践編（後）

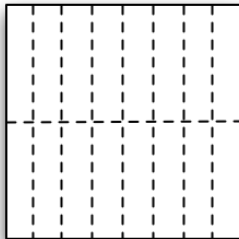
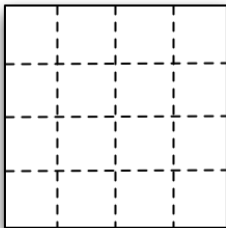
陰山

はじめに

2次元並列

ハイブリッド
並列化

どちらがよいか？



2次元領域分割の方法

実践編（後）

陰山

はじめに

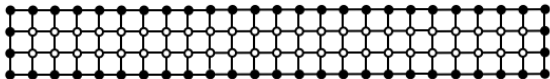
2次元並列

ハイブリッド
並列化

計算格子（白丸）：46 個

通信格子（黒丸）：54 個

合計：100 個



2次元領域分割の方法

実践編（後）

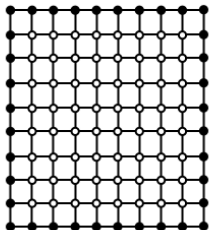
陰山

はじめに

2次元並列

ハイブリッド
並列化

計算格子（白丸）：64 個
通信格子（黒丸）：36 個
合計：100 個



面積の等しい長方形の中で、4辺の長さの合計が最も小さいものは正方形。

2次元領域分割による並列コード

実践編（後）

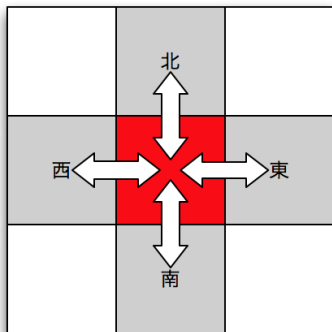
陰山

はじめに

2次元並列

ハイブリッド
並列化

領域分割による並列化を行うときに注意すべき点は、MPI プロセスの配置方法である。



隣同士の通信がもっとも通信速度的に「近い」位置にプロセスを配置することが望ましい。

2次元領域分割による並列コード

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

4番のプロセスはランク番号1,3,5,7のプロセスと頻繁に通信する

2	5	8
1	4	7
0	3	6

2次元領域分割による並列コード

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

もしも使用している並列計算機のネットワークの構成上、4番のプロセスはむしろランク番号0,2,6,8のプロセスと通信した方が速い場合には、以下のようにプロセスを配置する方が望ましい。

5	6	7
0	4	8
1	2	3

MPI_CART_CREATE

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

MPI関数の一つ MPI_CART_CREATE を使うと（使用する計算機がネットワークの通信性能に関する情報を提供している場合には）通信効率の点で最適な配置でプロセスを自動的に分配してくれる。

演習

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

コード

/tmp/120719/heat8

heat8.js

heat8_flatmpi.js

single MPIでのプロセス数と計算速度

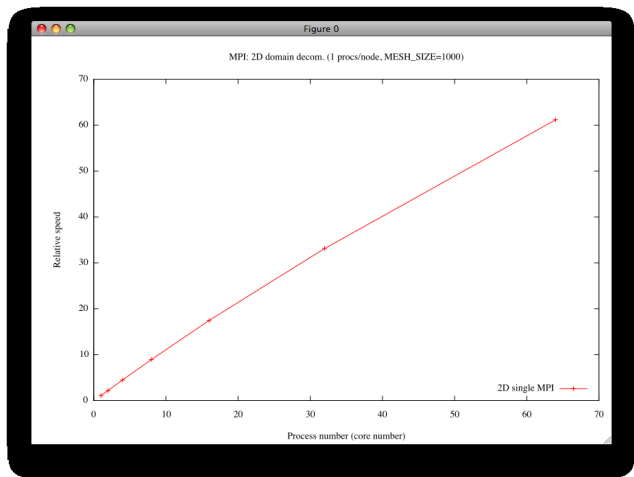
実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化



1次元領域分割と2次元領域分割の比較

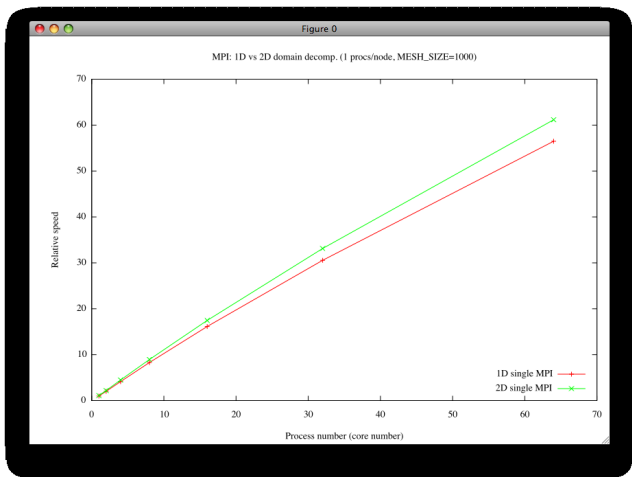
実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化



実践編（後）

陰山

はじめに

2 次元並列

ハイブリッド
並列化

ハイブリッド並列化

MPI + OpenMP = ハイブリッド並列化

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

簡単。

一つのノードに一つの MPI プロセスを走らせ、計算時間の最もかかる do-loop 部分 いまの場合はヤコビ法の部分 だけを「4. OpenMP を使った並列計算」で学んだ方法で二つのスレッドに fork させて計算すればよい。（scalar はデュアルコアなので2スレッド。）

演習

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

heat8 をハイブリッド並列化せよ。
解答は heat9 にあるが、できるだけ自分でやってみよ。

flat MPI と hybrid による 128 並列計算の比較

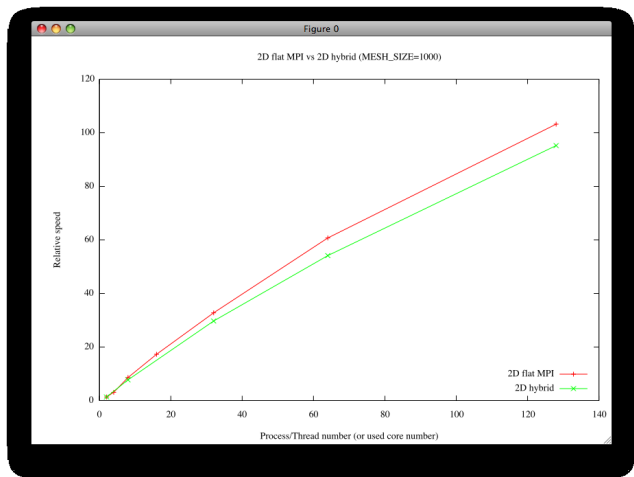
実践編（後）

陰山

はじめに

2 次元並列

ハイブリッド
並列化



レポート課題

実践編（後）

陰山

はじめに

2次元並列

ハイブリッド
並列化

この演習で考えた熱伝導問題では、一辺の長さが1の正方形領域を $NGRID \times NGRID$ 個の等間隔の格子で離散化し、計算した。格子間隔は横方向、縦方向ともに $h = 1/(NGRID+1)$ 。では、縦の辺の長さが3、横の辺の長さが5、の長方形を、

- 縦の方向に 200 点（格子間隔 $h_1=3/201$ ）
- 横の方向に 400 点（格子間隔 $h_2=5/401$ ）

で離散化したとき、flat MPI で8プロセスを使って並列化して解く場合、領域をどのように分割するのが良いか述べてよ。（理論的な考察のみで十分。実際に計算しなくてもよい。）

提出先：kageyama.lecture@gmail.com

メールタイトル：120719 学籍番号 名字

例：120719 120x227x Yamada

締めきり：2012年7月26日 24時