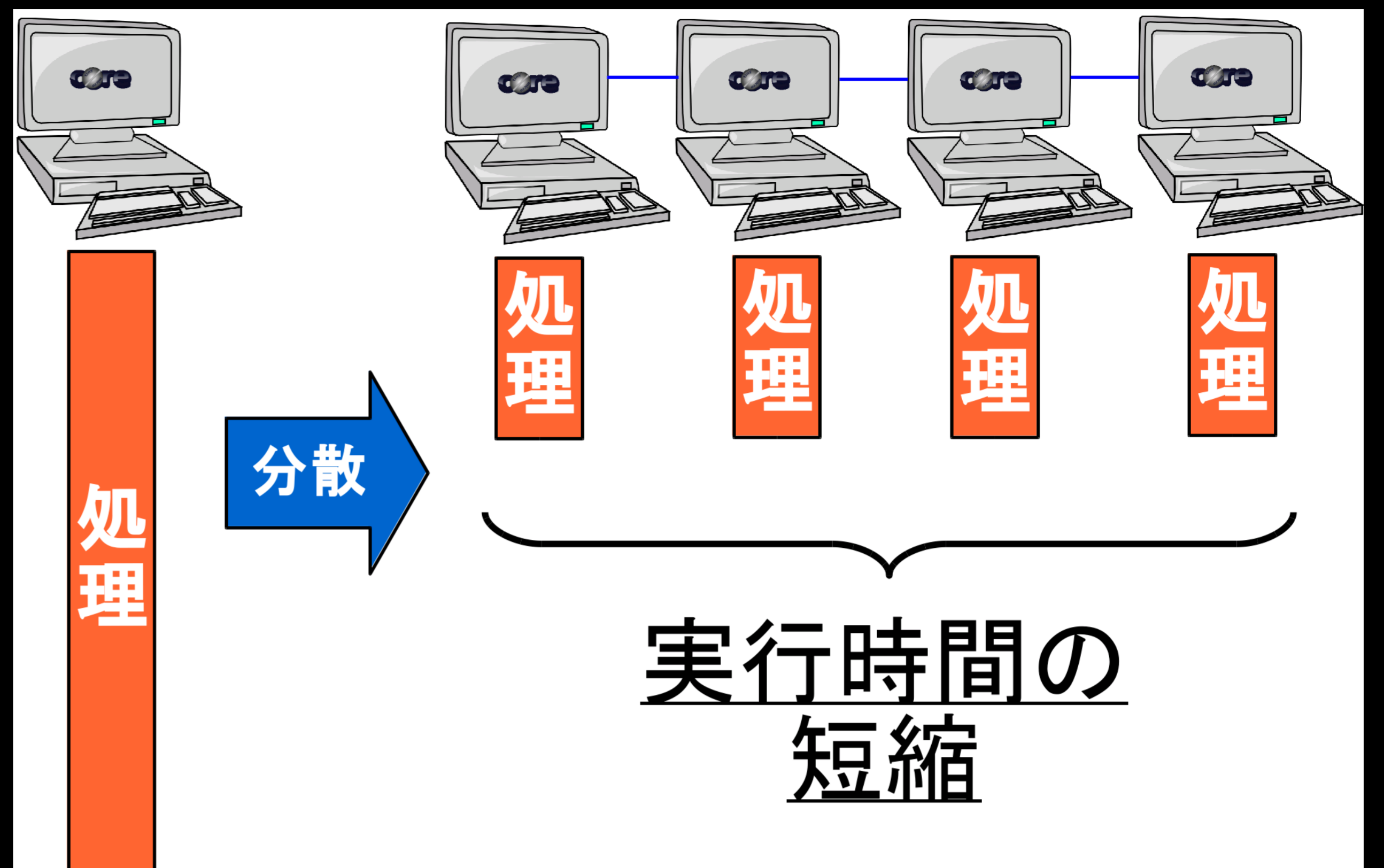


# 並列計算システムの構築

## Constructing a parallel processing cluster

### 1. 並列計算

- ・ ネットワークでPCを接続
- ・ 同期・データの送受信
- ・ **分散して処理**することにより実行速度を短縮
- ・ ワーカー ... 計算をする  
マスタ ... ワーカーを統括



### 2. 方法

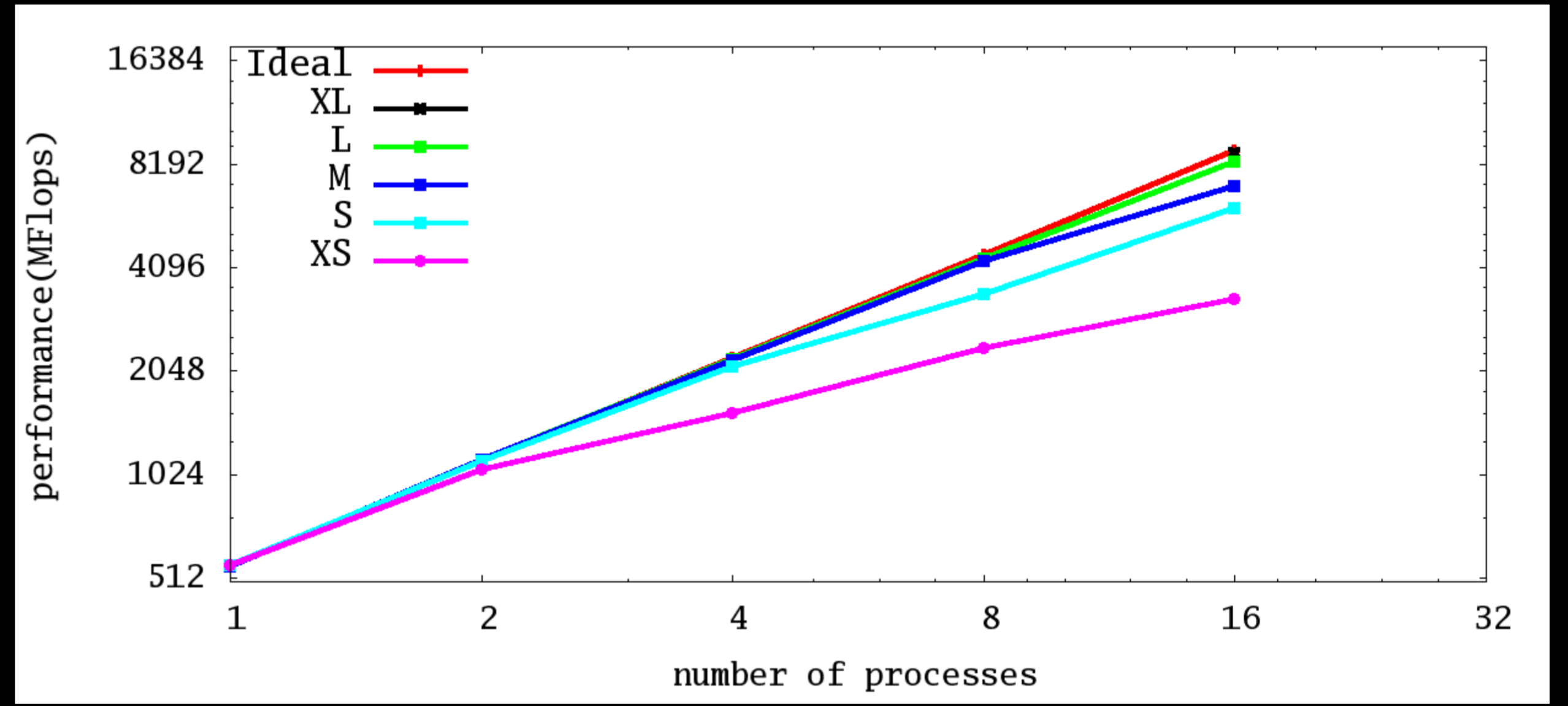
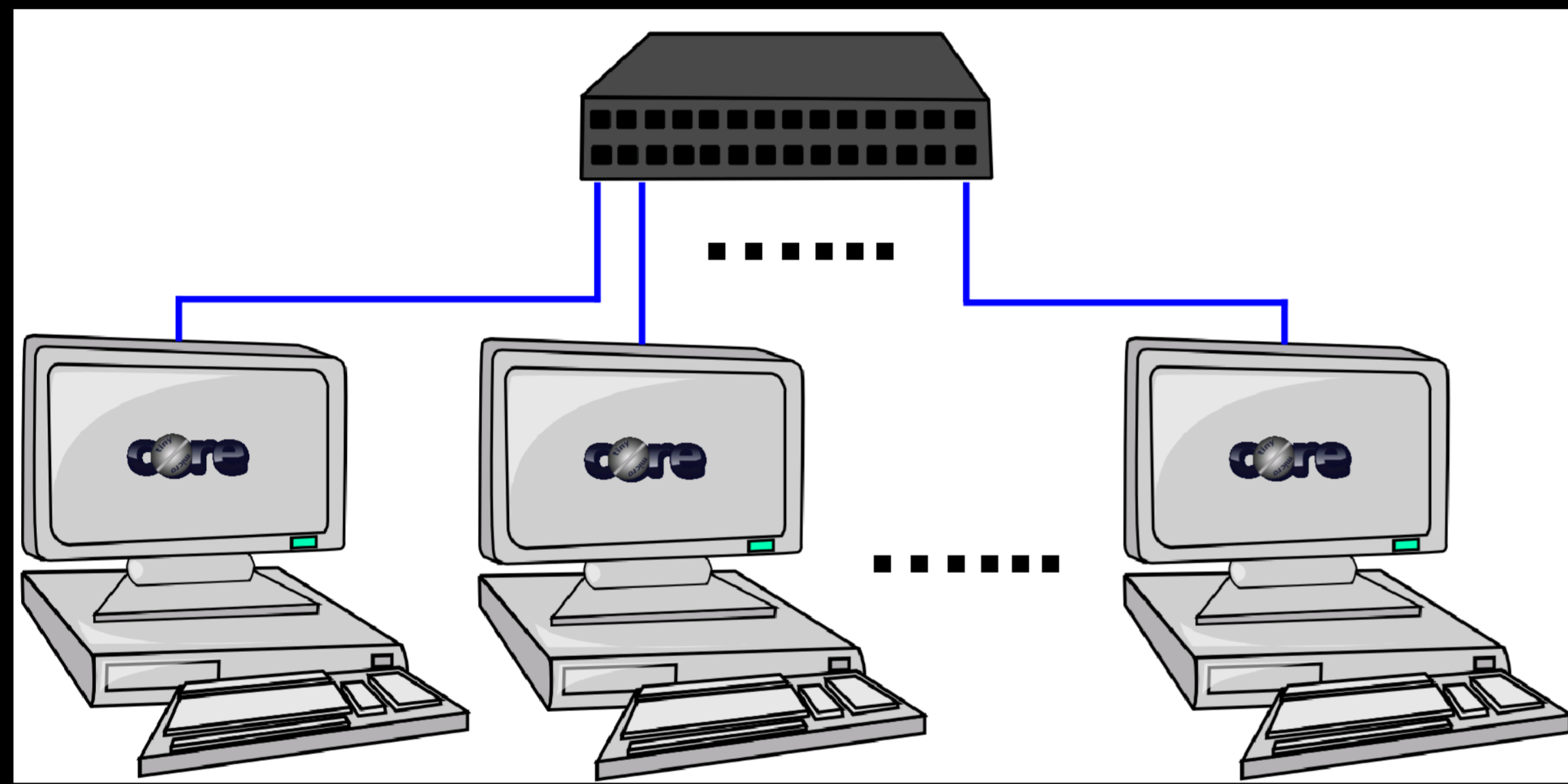
- ・ 学校のコンピュータ室の環境
- ・ OSはLinuxを**ネットワークブート**  
→インストールの必要なし
- ・ 並列計算ライブラリには **MPI**

### 3. 性能測定

- ・ 姫野ベンチマークを利用  
**たくさん通信**するプログラム
- ・ 計算するサイズを指定

### 4. 実験1

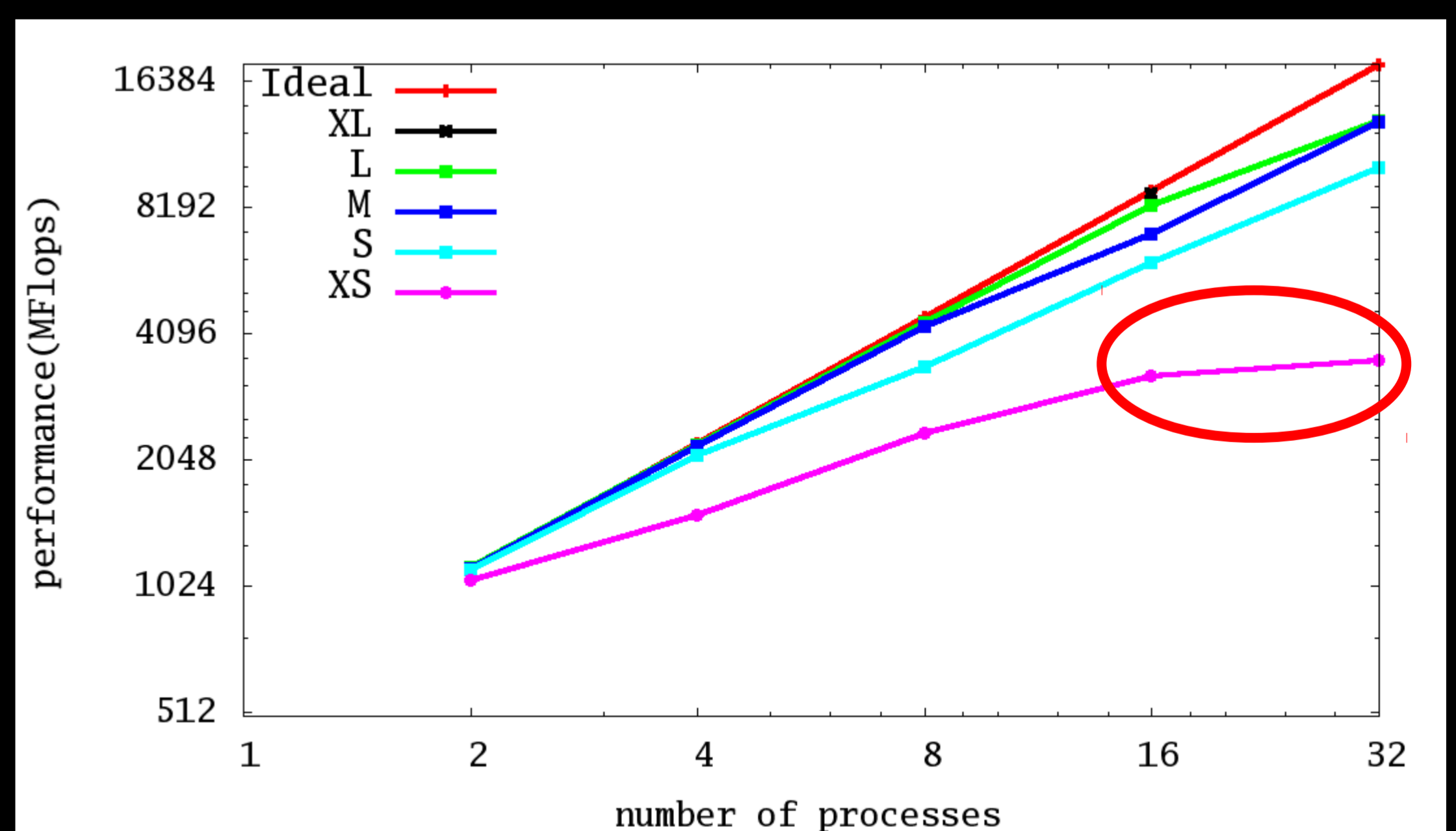
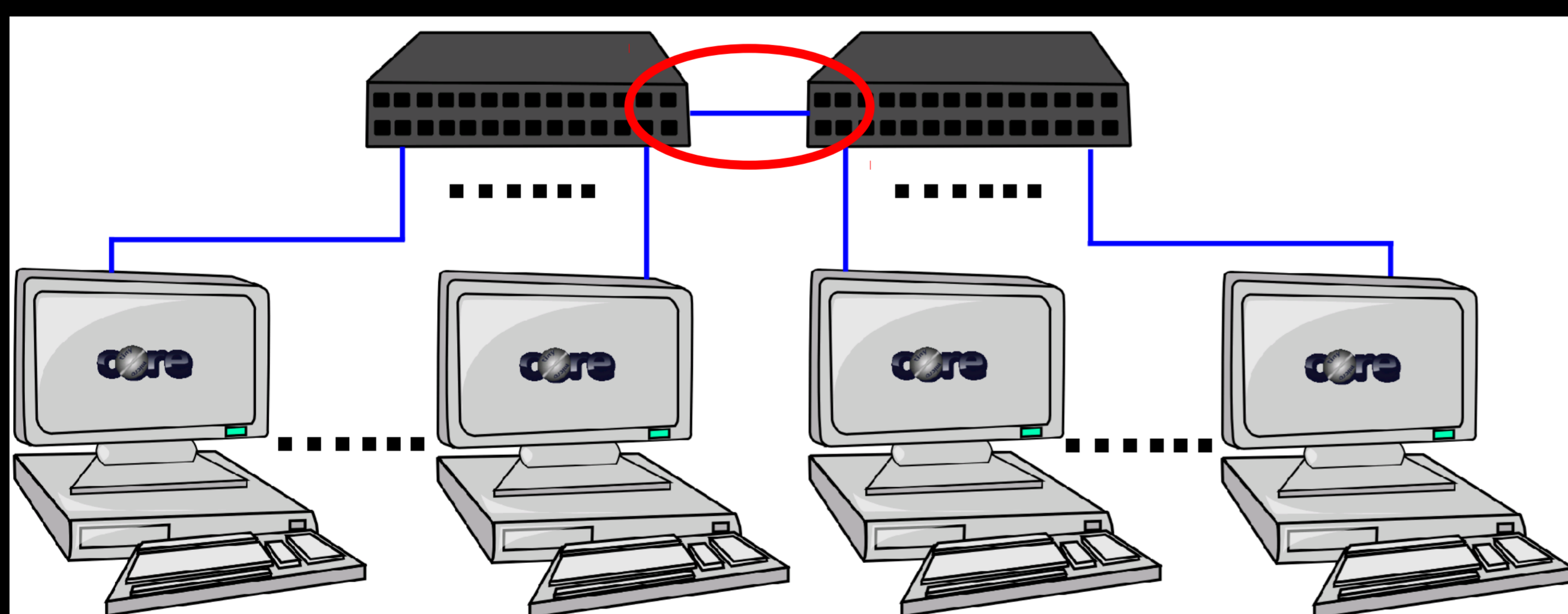
- ・ **1台のネットワークスイッチに1, 2, 4, 8, 16台のPCをつなぐ**



- ・ 計算サイズが**大きい** → 性能が台数に**比例**
- ・ 計算サイズの**小さい** → 性能が台数に**比例しない**

### 5. 実験2

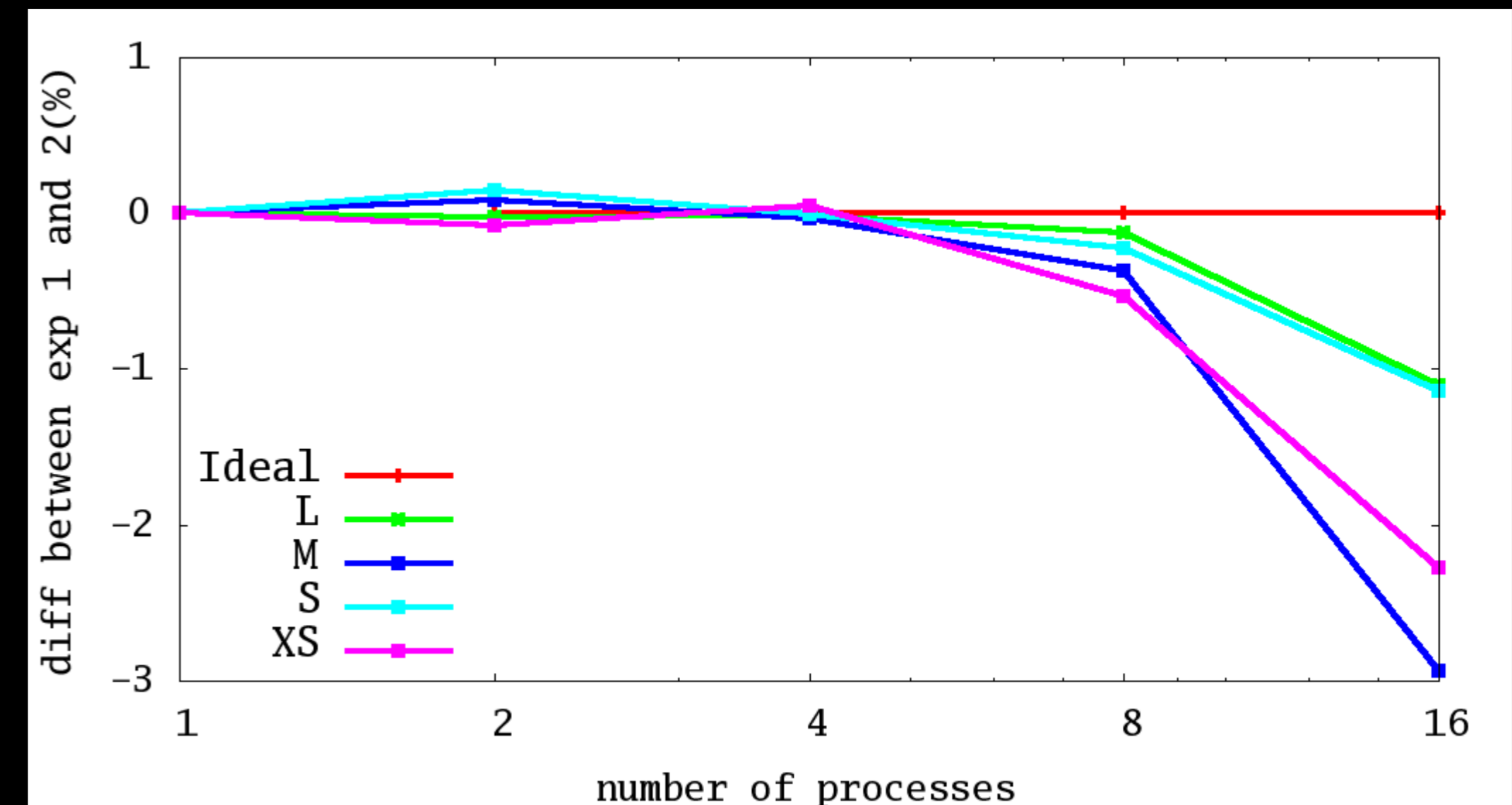
- ・ **2台のネットワークスイッチに左右均等に2, 4, 8, 16, 32台のPCをつなぐ**



- ・ XSの16台~32台では **ほぼ横ばい** (端的な**オーバーヘッド**)

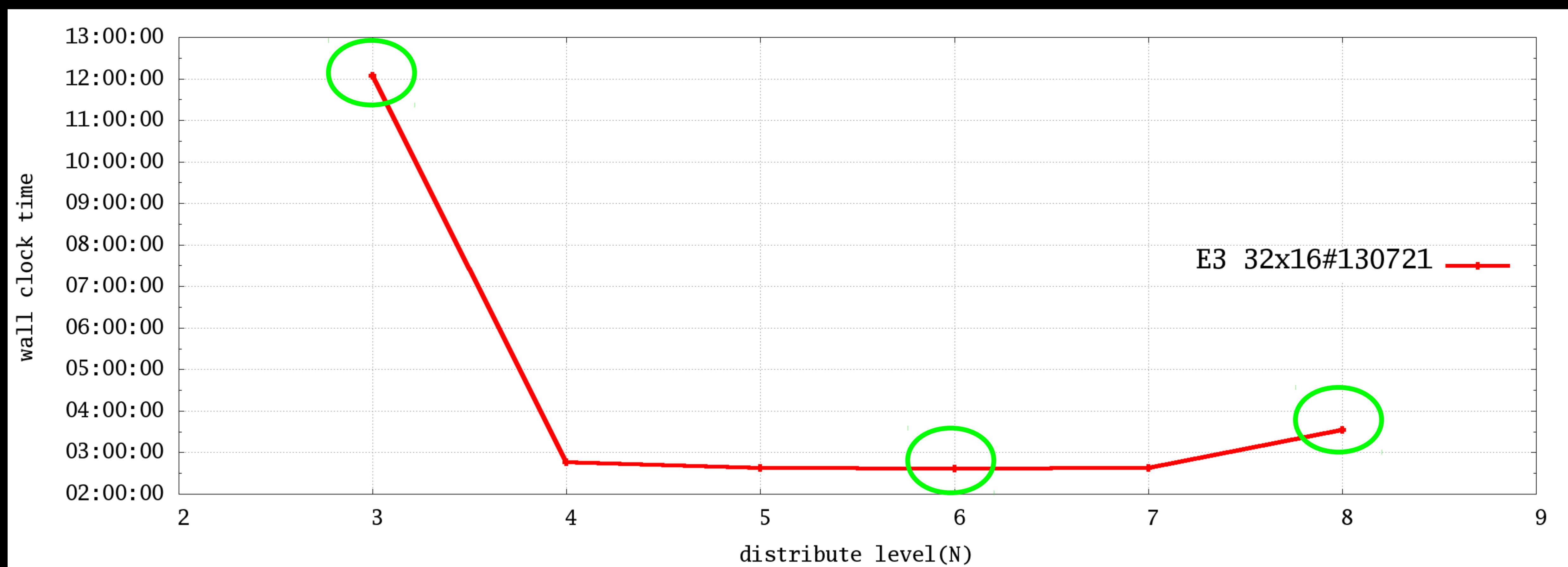
### 6. 実験1と実験2の比較

- ・ 台数が多い、計算サイズが**小さい**ほど**実験2の性能が低い**
- ・ ネットワークスイッチの間により大きな負荷がかかるため



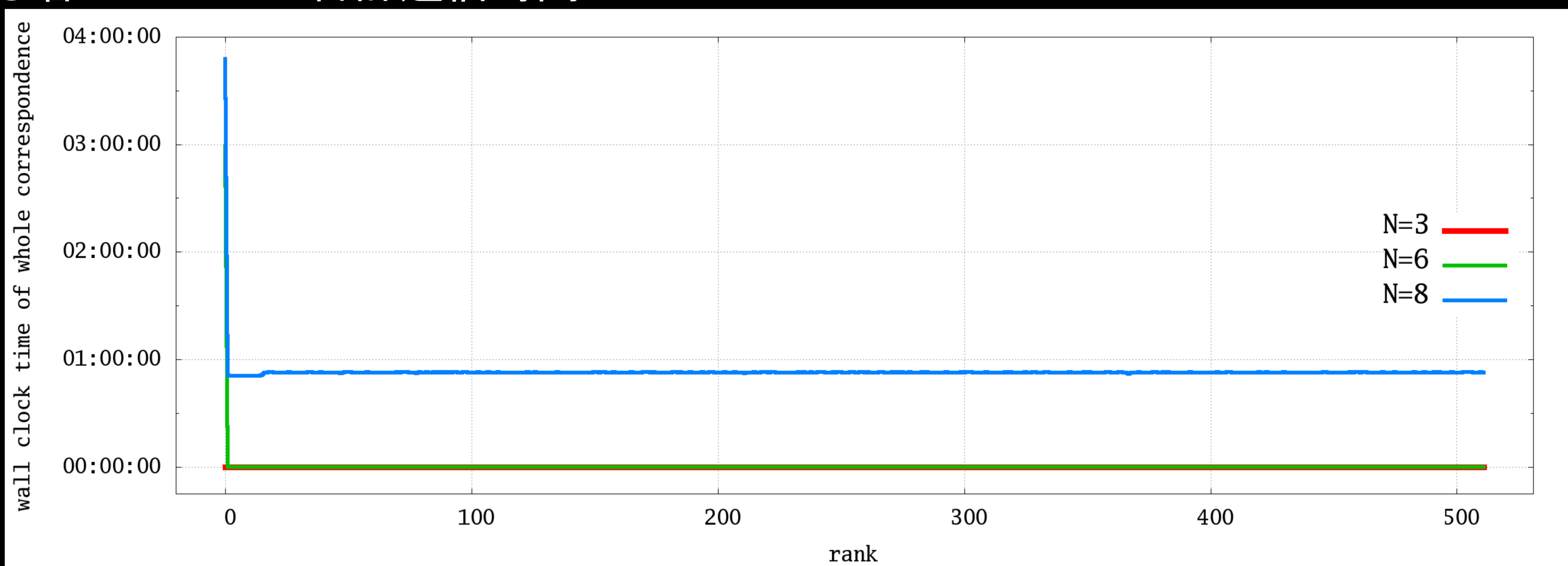
# 5. X=5での実行結果

○各配布レベル(N)での全体の実行時間



**N=3, 6, 8** の実行時間は極端  
→それらに注目

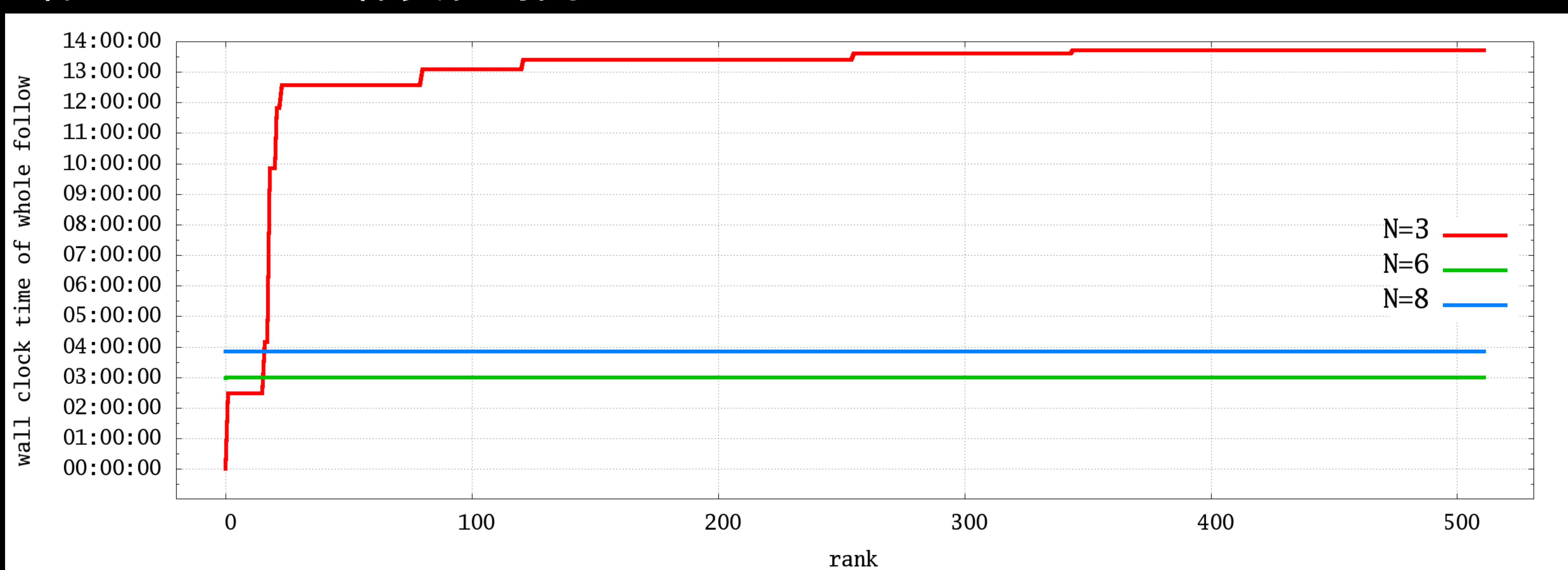
○各プロセスの合計通信時間



Nの値が大きい = マスタが作る組の量もワーカーに配る回数  
も多い(オーバーヘッドが発生)

N=8は**他より通信時間が長い**

○各プロセスの全体実行時間



Nの値が小さい = 配られた問題によって処理量が異なる

N=3は**負荷分散できていない**(粒度が荒い)

## ○謝辞

- 並木中等教育学校 齊藤 達也 先生
- 同 粉川 雄一郎 先生
- 筑波大学計算科学研究センター 朴 泰佑 教授
- 産業技術総合研究所 山本 直孝 様