

# Einführung in die Technische Informatik

Wintersemester 2022/23

Praktikum: Verteiltes Rechnen mittels MPI

Ziel des Praktikums ist das Verteilen der Matrix Multiplikation über mehrere Kerne und Prozessoren der Intel(R) Xeon(R) E5-2680 v3 CPUs des Taurus HPC-Systems der TU Dresden.

## Vorbereitung

### Aufgabe 1

Bestimmen Sie die maximale Double Precision Leistung des gegebenen Prozessors in GFLOPS.

### Aufgabe 2

Wiederholen Sie die Details zur Kommunikation mittels MPI aus der Vorlesung. Welche Funktionen könnten bei der Implementierung einer via MPI verteilten Matrix Multiplikation helfen?

### Aufgabe 3

Machen Sie sich mit dem Modul System des Supercomputers vertraut. Nutzen Sie dazu die folgende Webseite:

<https://doc.zih.tu-dresden.de/hpc-wiki/bin/view/Compendium/Applications>

Welche MPI Versionen sind in der „scs5“ Modulumgebung zu finden?

### Aufgabe 4

Wiederholen Sie das kompilieren von Anwendungen zur Nutzung mit MPI. Wie können Sie den verwendeten Compiler sowie die verwendete MPI Version ermitteln?

### Aufgabe 5

Machen Sie sich mit Slurm vertraut. Nutzen Sie dafür folgende Webseite:

<https://doc.zih.tu-dresden.de/hpc-wiki/bin/view/Compendium/Slurm>

Welche Konfiguration für srun müssen Sie nutzen, um einen Job mit insgesamt 12 Prozessen gleichmäßig verteilt auf 2 Knoten zu starten?

## Durchführung

Nutzen Sie folgendes Kommando um sich während der Praktikumszeit einen Test als Job zu starten. Bitte beachten Sie, dass `srun` `mpirun` ersetzt:

```
srun -n $NR_MPI_PROCESSES -t 00:10:00 --reservation=p_lv_eti2223_798 \  
-A p_lv_eti2223 --exclusive (weitere srun flags - siehe Aufgabe 5) ./my_command
```

Verzichten Sie außerhalb der Praktikumszeit auf das Flag  
`--reservation=p_lv_eti2223_798`

(`./my_command`, `$NR_MPI_PROCESS` und (weitere `srun` flags) müssen Sie natürlich ersetzen)

## Aufgabe 6

Nutzen Sie die Optimierungen aus dem ersten Praktikum zusammen mit „-march=native“ und „-O3“ und implementieren Sie eine verteilte Matrix Multiplikation auf  $n \times n$ -Matrizen mittels MPI. Dabei sollen die genutzten Daten von Rank 0 an die anderen Ranks verteilt werden. Die berechneten Daten sollen am Ende wieder bei Rank 0 gesammelt werden. Ermitteln Sie die Performance ihrer Implementierung für 1 bis 48 Prozesse und die Matrixgrößen  $n = 1024$ ,  $n = 2048$  und  $n = 4096$  auf einem bzw. zwei Knoten in GFLOPS. Wiederholen Sie jede Messung 10mal.

## Protokoll

Dokumentieren Sie ihre Recherchen und Antworten zu den Aufgaben 1 bis 6 in einem Protokoll. Stellen Sie die Ergebnisse aus den Aufgaben 6 geeignet grafisch dar. Geben Sie dabei den Mittelwert und jeweils das Minimum und das Maximum mit an. Wählen Sie einen geeigneten Mittelwert. Dokumentieren Sie auch Quellcode(-änderungen), genutzte Hardware genutzte Software (insb. Compiler und Flags) und Ihren Auswertungsprozess.

## Abgabe

Das Protokoll muss bis zum 21.12.2022 abgegeben werden. Auf dem Protokoll muss klar erkenntlich sein, wer an der Erstellung des Protokolls beteiligt war. Bitte reichen Sie pro Gruppe ein Protokoll über den Ordner Ihrer Gruppe in OPAL ein.

## Anmerkungen

- Sie müssen sich via VPN in die Universität verbinden, um auf das HPC-System gelangen zu können: [https://tu-dresden.de/zih/dienste/service-katalog/arbeitsumgebung/zugang\\_datennetz/vpn/openvpn](https://tu-dresden.de/zih/dienste/service-katalog/arbeitsumgebung/zugang_datennetz/vpn/openvpn)
- Anmeldung:  
`ssh <zih-login>@taurus.hrsk.tu-dresden.de`
- Ordner anlegen:  
`mkdir Uebung3`
- Dinge auf Taurus Kopieren:  
`scp ./matmul0.c <zih-login>@tauruslogin4.hrsk.tu-dresden.de:/home/Nutzername/Uebung1/`
- Ressourcen anfordern:  
`salloc --reservation=p_lv_eti_2021_189 --nodes=2 --tasks-per-node=24 --comment=no_monitoring -t hh:mm:ss -p haswell`
- Programme auf den angeforderten Ressourcen ausführen:  
`srun <programm>`  
Sie können die per `salloc` angefragten Ressourcen hier weiter einschränken.
- Ressourcen freigeben:  
`exit`
- SSH Verbindung beenden:  
`exit`
- Die Reservierung läuft nur (p\_lv\_eti\_2021\_190) über die Zeit des Praktikums
- Details zu `srun` und `salloc`:  
<https://doc.zih.tu-dresden.de/hpc-wiki/bin/view/Compendium/Slurm>  
oder:  
`man srun`