

## Wissenschaftliches Rechnen

Sommersemester 2019

## Übungsblatt 3

### Aufgabe 9 (Programmaktualisierung)

Denken Sie daran, Ihr Programm mit `git pull origin praktikum` zu updaten und mit `cmake ..` und `make -j neu` zu compilieren. Falls Sie Probleme haben können Sie ihr Programm mit `git reset --hard origin/praktikum` auf den Stand der letzten Woche setzen und dann neu updaten. Alternativ können Sie auch mit `git checkout path` einzelne Dateien mit dem Dateipfad `path` zurücksetzen.

### Aufgabe 10 (Iterative Löser und Vorkonditionierer)

Betrachten Sie nochmals das Problem `Problem = Simple2D` mit `Mesh = UnitSquare` auf `level = 3, ..., 9`. Ziel dieser Aufgabe ist es, die Eigenschaften und Konvergenzraten des CG-Verfahrens (`LinearSolver = CG`) und des GMRES-Verfahrens (`LinearSolver = GMRES`) mit unterschiedlichen Vorkonditionierern zu untersuchen. Betrachten Sie hierfür die folgenden Konfigurationen:

```
Preconditioner = Jacobi (bei CG und GMRES)
Preconditioner = SGS (bei CG)
Preconditioner = GaussSeidel (bei GMRES)
Preconditioner = Multigrid (bei CG und GMRES)
```

Verwenden Sie für den Multigrid-Vorkonditionierer das Grobgitterlevel `plevel = 2` und den Grobgitterlöser `BasePreconditioner = LIB_PS`.

Setzen Sie weiterhin `LinearSteps = 800` als maximale Anzahl von Schritten.

Starten Sie das Programm immer mit einem Prozessor (also `mpirun -np 1 M++`).

Was fällt Ihnen bei Anwendung des Jacobi-Vorkonditionierers auf? Lassen Sie sich dabei in jedem Schritt den Fehler mit Hilfe von `LinearVerbose = 1` ausgeben (zumindest beim CG-Verfahren). Beachten Sie die extreme Verschlechterung der Konvergenzrate beim GMRES-Verfahren in höheren Leveln. Worauf ist dies zurückzuführen (Hinweis: Reduzierung des Speicherplatzbedarfs)?

Erstellen Sie zu den verschiedenen vorkonditionierten iterativen Lösern geeignete Tabellen. Diese sollen die Anzahl der Iterationsschritte des linearen Löser und die Konvergenzrate des iterativen Löser auf dem jeweiligen Verfeinerungslevel beinhalten.

### Aufgabe 11 (Iterative Löser und Vorkonditionierer - Bonusfrage)

Beobachten und erklären Sie das Verhalten, wenn Sie

- das Programm auf mehreren Prozessoren starten,
- anstatt `Problem = Simple2D` die Probleme `Problem = Discontinuous` und `Problem = Divergent` verwenden,
- anstatt der Krylow-UVR-Verfahren ein linearer Löser ohne Dämpfung `LinearSolver = LS` verwenden.

### Aufgabe 12 (Praktikumsbericht #1 - Optionaler Zusatz)

Erweitern Sie Ihren ersten Praktikumsbericht um einen kurzen Überblick über die benutzten iterativen Löser und Vorkonditionierer. Arbeiten Sie deren Unterschiede anhand Ihrer Experimente aus.

### Homepage:

Unter dem Link <http://www.math.kit.edu/ianm3/lehre/einfwissrech2019s/de> erreichen Sie die Homepage zur Vorlesung. Dort finden Sie alle Informationen zur Vorlesung.