

Parallelisierungsschema

Hochleistungsrechnen 2014/15 , Hu & Timm

Jacobi-Verfahren

Daten und Kommunikation

Die Daten der Matrix werden Blockweise nach Zeilen aufgeteilt. Jeder Prozess bekommt etwa gleich viele Zeilen. Dies lässt sich mit MPI_SCATTERV bewerkstelligen. Um diese Funktion nutzen zu können muss die gesamte Matrix aber auf dem Root-Prozess zunächst initialisiert werden. Sollte dafür nicht genügend Speicher zur Verfügung stehen gibt es ein Problem. Für die Zeit der Berechnung wird für jeden Prozess nur anteilig Speicher benötigt. Danach wird die Matrix im Root-Prozess wieder zusammengesetzt unter Benutzung von MPI_GATHERV.

Damit jeder Prozess rechnen kann werden sogenannte Halo-Zeilen eingefügt. Diese Halo-Zeilen enthalten den benötigten Teilabschnitt der Matrix des Nachbarn-Prozesses. Der Austausch der Halo-Zeilen lässt sich mit asynchroner Kommunikation effizient durchführen. Halo-Zeilen gibt es bei im allgemeinen Fall zwei, je eine unten und oben. Im Falle des letzten und ersten Prozesses gibt es jeweils nur eine Halo-Zeile.

Berechnung

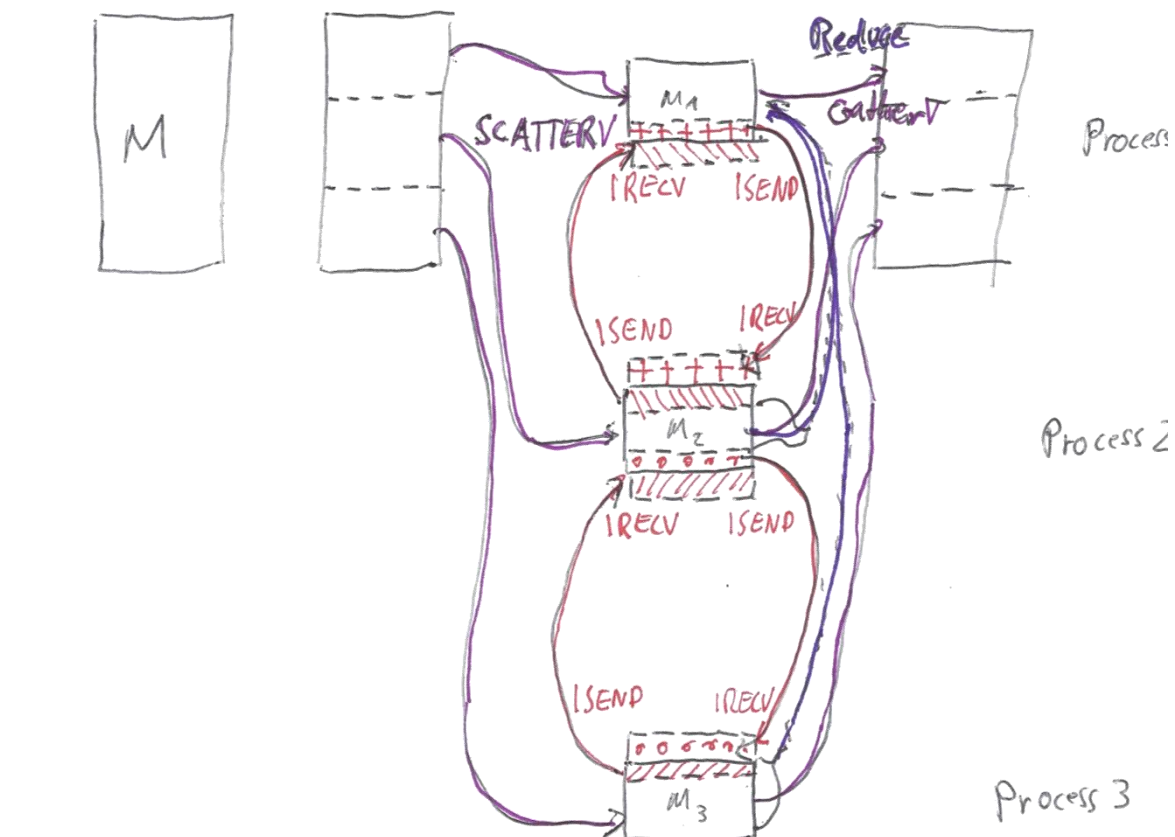
Alle Prozesse können sofort anfangen zu rechnen, da sie Alle für die erste Iteration notwendigen Daten haben. Die Randinformationen bekommen sie von Ihren Nachbarn, sobald diese berechnet wurden. Alle Prozesse befinden sich zu einem Zeitpunkt in der gleichen Iteration (MPI_Barrier vorausgesetzt)

Abbruch

Beim Abbruch nach Iterationen kann jeder Prozess aufhören, sobald der Iterationszähler mit dem Abbruchwert übereinstimmt. Die Matrix wird umgehend zusammengesetzt.

Der Abbruch nach Genauigkeit erfordert in jedem Prozess ein lokales Residuum zu berechnen. Dieses wird dann mittels MPI_Reduce(....,MPI_Max,....) zusammengeführt und vom Root-Prozess geprüft. Sollte das Abbruchkriterium erfüllt sein sendet der Root-Prozess einen Broadcast an alle übrigen. Die Matrix wird dann umgehend zusammengesetzt.

Jacobi



Gauss-Seidel

Daten und Kommunikation

Die Prinzipielle Aufteilung der Daten erfolgt wie beim Jacobi-Verfahren. Auch das Prinzip der Halo-Zeilen bleibt erhalten. Entfallen tut dafür die zweite lokale Matrix.

Berechnung

Bei der Berechnung der Lösung ergibt sich ein wichtiger Unterschied. Die Werte sind davon abhängig, was der vorhergehende Prozess berechnet hat, da nur eine Matrix benutzt wird. Dies bedeutet, dass ein Pipeline-Verfahren angewendet werden muss.

Füllen der Pipeline

Der Erste Prozess läuft los, alle anderen müssen warten, bis dieser den letzten Abschnitt seiner Matrix gerechnet hat und mit Prozess 2 austauscht. Dann kann auch Prozess 2 beginnen zu rechnen. Dieses Verfahren wiederholt sich so oft, bis alle Prozesse loslaufen konnten.

Bedeutung der Pipeline

Die Pipeline hat zur Folge, dass während der Berechnung keine (Gesamt)Matrix im Speicher liegt, die einen gleichen Iterationsstand hat. Die Matrix ist hingegen mosaikartig in Bereiche mit unterschiedlicher Iteration unterteilt. Dies hat zur Folge, dass die Berechnung nicht einfach angehalten werden kann.

Leeren der Pipeline

Wurde aufgrund des Abbruchkriteriums entschieden die Berechnung zu beenden, muss die Matrix auf einen einheitlichen Iterationsstand gebracht werden. Als einheitlicher Iterationsstand eignet sich nur der Maximale Iterationsstand der Matrix. D.h. Der erste Prozess muss seine Iteration zu Ende berechnen, dann der Zweite, (der Iterationszyklus ist dann bei beiden gleich) usw., bis der letzte Prozess genauso viele Iterationen durchlaufen hat wie der erste Prozess. Erst jetzt kann die Matrix im Hauptprozess zusammengesetzt werden.

Nachteile der Pipeline

Für das füllen und leeren der Pipeline wird Zeit verschwendet und durch das Warten der anderen Prozesse die Rechnerressourcen ineffizient genutzt. Dieser Nachteil relativiert sich bei einer lange dauernden Berechnung durch die schnellere Konvergenz des Verfahrens.

Abbruch

Beim Abbruch nach Iterationen kann jeder Prozess aufhören, sobald der eigne Iterationszähler mit dem Abbruchwert übereinstimmt. Die Pipeline leert sich so automatisch, da alle beim gleichen Iterationsstand beenden.

Für den Abbruch nach Genauigkeit ist es nur möglich ein leeren der Pipeline auf Befehl auszulösen. Zur Bestimmung dieses Zeitpunktes sendet jeder Prozess sein lokales Residuum zusammen mit den Residuen seiner Vorgänger an den nachfolgenden Prozess. Der letzte Prozess prüft, ob das Abbruchkriterium erfüllt ist. Falls ja wird die Iterationszahl bestimmt, mit der sich die Pipeline zum nächstmöglichen Zeitpunkt leeren lässt. Dies kann bedeuten, dass noch so viele Iterationen, nötig sind, wie Prozesse vorhanden.

Gauss-Seidel

