Übungen zur Vorlesung
Parallele numerische Algorithmen
Blatt 4

Aufgabe 13

Beschreiben Sie einen Algorithmus zur Berechnung von $(x_1 + x_2)(x_2 + x_3)(x_3 + x_4)$ mit Hilfe eines Präzedenzgraphen. Wie lauten T_1, T_∞ und p^*? Beschreiben Sie einen optimalen Schedule für $p = 2$ Prozessoren.

Aufgabe 14

Zeigen Sie, dass ein Polynom der Form $p(x) = a_n x^n + \ldots + a_1 x + a_0$ in $\Theta(\log n)$ Schritten parallel ausgewertet werden kann. Die Eingaben sind der Wert von x und die Koeffizienten $a_0, \ldots, a_n$. Bestimmen Sie Speedup und Effizienz des zugehörigen Algorithmus ohne Berücksichtigung der Kommunikationszeiten.

Aufgabe 15

Entwerfen Sie einen Algorithmus, um möglichst schnell $\sum_{i=1}^{k} A^i$ zu berechnen, wobei A eine $n \times n$ Matrix ist. Beschreiben Sie die Ablauf des Algorithmus mit Hilfe eines Präzedenzgraphen.

Aufgabe 16

Entwerfen Sie Algorithmen für die folgenden Kommunikationsprobleme und zeigen Sie, dass die Algorithmen eine optimale Laufzeit haben. Für das Kommunikationsmodell gelten die in der Vorlesung genannten Annahmen.

Kommunikationsprobleme

i Einzel-Broadcast

ii Allgemeiner Broadcast

iii Kompletter Austausch

Topologien

a (ausgeglichener) Binärbaum mit $p = 2^{k-1}$ Prozessoren

b d-dimensionales Gitter mit $p = n^d$ Prozessoren

c Hyperwürfel mit $p = 2^k$ Prozessoren

Aufgabe 17

Die Faltung (convolution) zweier n-dimensionaler Vektoren a und b ist definiert als $2n-1$ dimensionaler Vektor y mit $y_k = \sum_{i+j=k+1} a_i \cdot b_j$, wobei $1 \leq k \leq 2n-1$ und $1 \leq i, j \leq n$. Entwerfen Sie einen möglichst schnellen Algorithmus zur Faltung zweier Vektoren auf einem linearen Array.