

Zentralübung Rechnerstrukturen: Leistungsbewertung und Fehlertoleranz Musterlösung zu Aufgabenblatt 2

1 Leistungsmessung I:

Die Zykluszeit hängt von der Organisation und der Technologie ab.

Die Anzahl der Instruktionen ist bedingt durch die Befehlssatzarchitektur und die Güte des Compilers.

Die Zyklen pro Instruktion werden durch die Organisation und die Befehlssatzarchitektur beeinflusst.

2 Leistungsmessung II:

$$f = \frac{i \cdot CPI}{t}, MIPS = \frac{f}{CPI \cdot 10^6}$$

$$f_A = \frac{3.5 \cdot 10^6 \cdot \frac{7}{5}}{2 \cdot 10^{-3}} = 2450 MHz$$

$$MIPS_A = \frac{f_A}{CPI_A \cdot 10^6} = \frac{2.45 \cdot 10^9}{5 \cdot 10^6} = 1750 MIPS$$

$$f_B = \frac{1.5 \cdot 10^6 \cdot \frac{3}{2}}{2 \cdot 10^{-3}} = 1125 MHz$$

$$MIPS_B = \frac{f_B}{CPI_B \cdot 10^6} = \frac{1.125 \cdot 10^9}{\frac{3}{2} \cdot 10^6} = 750 MIPS$$

Es ist Prozessor B zu wählen, weil

- ohne Berechnung: Gleich schnell in der Abarbeitung (2ms) bei wesentlich weniger ausgeführten Instruktionen (1.5 vs. 3.5 Mio Instruktionen)
- halbe Taktfrequenz ($P \sim U^2 \cdot f$, Fertigung)

3 Leistungsmessung III:

Benchmark-Berechnung

- Anzahl Instruktionen

$$i = \sum i_{typ} \\ = (300 + 75 + 150 + 25) * 10^3 = 550.000$$

- Taktzyklen

$$c = \sum i_{typ} * c_{typ} \\ = (300 * 1 + 75 * 2 + 150 * 3 + 25 * 4) * 10^3 = 1.000.000$$

- Zykluszeit bei 4GHz Taktfrequenz

$$t = \frac{1}{f} = \frac{1}{4GHz} = 0.25 * 10^{-9}s = 0.25ns$$

- Ausführungszeit

$$t_{exec} = c * t_{cyc} \\ = 1000 * 10^3 * 0.25 * 10^{-9} = 250 * 10^{-6}s = 250\mu s$$

- CPI

$$CPI = \frac{c}{i} = \frac{1000 * 10^3}{550 * 10^3} = \frac{100}{55} = \frac{20}{11} \approx 1.82$$

- MIPS

$$MIPS = \frac{i}{t * 10^6} = \frac{550.000}{250} = 2200$$

- MFLOPS: wie MIPS, wobei Anzahl der Befehle und Ausführungszeit nur für Fließkommaberechnung

$$MFLOPS = \frac{75.000}{(75.000 * 2) * (0.25 * 10^{-9}) * 10^6} = \frac{1}{0.5 * 10^{-3}} = 2000$$

4 Leistungsbewertung IV:

(vergl. Hennessy and Patterson, Computer Architecture A Quantitative Approach, 4.Auflage, S. 43-44.)

Es ändern sich nur die Zyklen pro Instruktion, Taktrate und Anzahl der Instruktionen bleiben gleich.

Der unoptimierte CPI-Wert errechnet sich nach:

$$CPI_{base} = \sum_{i=1}^n CPI_i * \frac{IC_i}{InstructionCount_{total}} = (4 * 0,25) + (1,33 * 0,75) = 2,0.$$

Die Zyklen pro Instruktion mit neuem FPSQR: $CPI_{newFPSQR}$ kann durch Abziehen der gesparten Zyklen erfolgen:

$$CPI_{newFPSQR} = CPI_{base} - 0,02 * (CPI_{oldFPSQR} - CPI_{newFPSQR})$$

$$= 2,0 - 0,02 * (20 - 2) = 1,64.$$

Die Alternative mit dem neuen CPI_{FP} -Wert errechnet sich analog zum CPI_{base} :

$$CPI_{newFP} = (0,75 * 1,33) + (0,25 * 2,5) = 1,62.$$

Aufgrund des geringeren CPI-Werts bietet sich die Alternative 2 mit den verbesserten Zyklen pro Gleitkommaoperation an.

Berechnung des Gewinns (Speedup) durch die Verwendung der Alternative (b) gegenüber dem vorherigen System (base):

$$\begin{aligned} Speedup_{(b)} &= \frac{CPU\ time_{base}}{CPU\ time_{(b)}} \\ &= \frac{i_{base} * Taktrate_{(b)} * CPI_{base}}{i_{(b)} * Taktrate_{base} * CPI_{(b)}} \\ &= \frac{CPI_{base}}{CPI_{(b)}} \end{aligned}$$

Eingesetzt ergibt sich:

$$Speedup_{(b)} = \frac{2,00}{1,62} \approx 1,23$$

→ Alternative (b) ist 1,23-mal schneller als das bisherige System. Die Berechnung des $Speedup_{(a)}$ verbleibt als kleine Übung.

5 Leistungsbewertung V:

- Der Laufzeitunterschied zwischen dem Base und Peak-Setup ist in den erlaubten Optimierungen zu suchen. Während Base nur konservative Standardoptimierungen erlaubt und gleiche Compileroptionen für alle Benchmarks vorschreibt, erlaubt Peak das aggressive Optimieren für die individuelle Architektur. Für 483.xalancbmk fällt auf, dass die Laufzeitunterschiede vernachlässigbar sind, dies lässt zwei Schlüsse zu:

- entweder waren die durchgeführten Optimierungen nicht wirkungsvoll,
- oder weitere Optimierungen wurden nicht angestrebt.

Ein Blick in die Sektion *Peak Optimization Flags* der Webseite verrät, dass außer `basepeak=yes` (welches nur die Anzahl der laufenden Kopien des Programms auf dem System beeinflusst) keine Compileroptimierungen angestrebt wurden. Dies steht im Gegensatz zu allen weiteren Programmen, die für den Peak-Lauf mit aggressiven Compileroptimierungen übersetzt wurden.

- Die gesuchte Formel lautet: $SPEC_{ratio} = \frac{Referenzzeit_x}{Laufzeit_x\ auf\ Testsystem}$ für einen Benchmark x . Umstellen und einsetzen der $SPEC_{ratio}$ und der Laufzeit für x aus der Tabelle liefert für $Referenzzeit_{462.libquantum} = 613 * 33,8\ s = 20719,4\ s$.

3. Es ist die Ultra Enterprise 2 von Sun Microsystems. Hierauf weist die errechnete Referenzlaufzeit des ausgewählten Benchmarks hin, welche recht gut mit dem Median der Laufzeiten überein stimmen:

Benchmark	Referenzzeit _{errechnet}	Laufzeit _{UltraEnterprise2}
462.libquantum	20719,4	20704

und auch der $SPEC_{int_base}^{2006} = 1.00$ spricht dafür.

6 Leistungsbewertung VI:

1. **Bedienzeiten:** $X_i = t_{\text{Zugriff}} + t_{\text{Übertragung}}$

$$X_1 = 12 \text{ ms} + \frac{100 \text{ kB}}{6000 \text{ kB/s}} = 28,67 \text{ ms}$$

$$X_2 = 10 \text{ ms} + \frac{100 \text{ kB}}{7500 \text{ kB/s}} = 23,33 \text{ ms}$$

2. **Maximaler Durchsatz:** $D_{\text{imax}} = \frac{1}{X_i}$

$$D_{1\text{max}} = \frac{1}{28,67 \text{ ms}} = 34,88 \frac{1}{\text{s}}$$

$$D_{2\text{max}} = \frac{1}{23,33 \text{ ms}} = 42,86 \frac{1}{\text{s}}$$

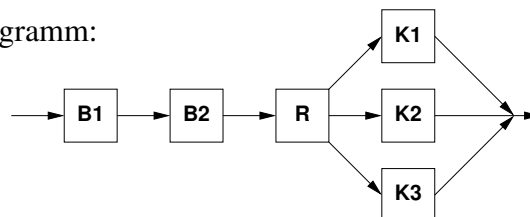
Nur Platten mit $D_{\text{max}} > A$ können eingesetzt werden, da sonst die Festplatte nicht genügend Zeit hat, um alle Aufträge rechtzeitig zu bedienen. Aufgrund von $A = 40/s$ ist somit nur die Platte 2 einsetzbar.

3. **Auslastung:** $U_i = D/D_{\text{imax}} = D * X_i$, wegen Systemsicht gilt hier $D = A$

$$U_2 = D * X_2 = 40 \frac{1}{\text{s}} * 23,33 \text{ ms} = 0,93, \text{ d.h. } 93\% \text{ Auslastung}$$

7 Fehlertoleranz: Blockdiagramm und Strukturformel

Zuverlässigkeitsblockdiagramm:



Strukturformel: $S = B_1 \wedge B_2 \wedge R \wedge (K_1 \vee K_2 \vee K_3)$
 (alternativ: $S = B_1 \text{ and } B_2 \text{ and } R \text{ and } (K_1 \text{ or } K_2 \text{ or } K_3)$)

Berechnung der Funktionswahrscheinlichkeit:

- Berechnung der Funktionswahrscheinlichkeit des parallelen Teilsystems K_1 bis K_3 durch Überführung in entsprechendes Seriensystem und Berechnung der Ausfallwahrscheinlichkeit:

$$(K_1 \vee K_2 \vee K_3) \rightarrow \neg(\neg K_1 \wedge \neg K_2 \wedge \neg K_3)$$

- $K \rightarrow \neg K$, entsprechend $\varphi(K) \rightarrow 1 - \varphi(K)$,
d.h. Ausfallwahrscheinlichkeit für K-System: $(1 - \varphi(K))^3$
- Transformierung in Funktionswahrscheinlichkeit: $\varphi(K) = 1 - ((1 - \varphi(K))^3)$
- Berechnung der Funktionswahrscheinlichkeit des kompletten Seriensystems:

$$\varphi = \underbrace{\varphi(B) * \varphi(B) * \varphi(R)}_{\text{Seriensystem}} * \underbrace{(1 - (1 - \varphi(K))^3)}_{\text{Parallelsystem}}$$

8 Fehlertoleranz: Maßzahlen und Berechnung

a) allgemein:

$$\varphi_m^n = \sum_{k=n}^m \binom{m}{k} * \varphi(K)^k * (1 - \varphi(K))^{(m-k)}$$

$n=8, m=10, \varphi(K) = \varphi(F) = 0,99$ – also:

$$\varphi_{10}^8 = \sum_{k=8}^{10} \binom{10}{k} * 0,99^k * 0,01^{(10-k)} = 0,999886$$

Chance auf Datenverlust somit: $1 - 0,999886 = 0,000114$.

b) $MTTF = 2a = (2 * 365 * 24 * 60)min = 1051200min$

$$MTTR = (2 + 10 + 2)min = 14min$$

$$V = \frac{MTTF}{MTTF + MTTR} = \frac{1051200}{1051214} = 0,999997$$

c) Bei der Berechnung der Punktverfügbarkeit wird typischerweise von einer konstanten Ausfallrate ausgegangen; dies trifft näherungsweise für die Betriebsphase der Badewannenkurve zu.

d) – allgemein: $z(t) = \frac{f_L(t)}{R(t)} = \frac{1}{R(t)} \cdot \frac{d}{dt}(F_L(t)) = \frac{1}{R(t)} \cdot \frac{d}{dt}(1 - R(t)) = \frac{-R'(t)}{R(t)}$

$$\rightarrow z(t) = \frac{\lambda e^{-\lambda t}}{e^{-\lambda t}} = \lambda.$$

– Gesucht: t mit $R(K, t) < R(S_{2v3}, t)$, d.h. Bestimmung der Schnittpunkte der beiden Überlebenswahrscheinlichkeiten.

2-von-3-System:

$$R(S_{2v3}, t) = \sum_{k=2}^3 \binom{3}{k} R(t)^k [1 - R(t)]^{3-k} = 3 * R(t)^2 - 2 * R(t)^3$$

Einzelkomponente: $R(K, t) = R(t)$

somit gilt: $R(K, t) = R(S_{2v3}, t) \Leftrightarrow R = 3 * R^2 - 2 * R^3$
 $\rightarrow R_1 = 0, R_2 = 1, R_3 = 0.5$

Wegen $R(K, t) = e^{-\lambda t}$ ergeben sich für R_2 und R_3 die dazugehörenden Werte $t_2 = 0$ und $t_3 = \frac{\ln(2)}{\lambda}$, d.h. das gesuchte Intervall ist $[t_2, t_3) = [0, \frac{\ln(2)}{\lambda})$.

– Es gilt:

$$MTTF = \int_0^{\infty} R(S, t) dt, \quad R(K, t) = e^{-\lambda t}$$

$$MTTF = \int_0^{\infty} R(S_{2v3}, t) dt = \frac{5}{6} \rightarrow \lambda = 1$$