Parallel Programming

Sheet No. 1 — PS

Benedikt Weihs

1. **Exercise 1**

   (a) $S(p) = \frac{t_S}{t_p}$ = Execution time with one processor / Execution time with p processors.

   (b) Amdahl's law states that the maximum speedup is given by $S(p) = \frac{t_S}{ft_S+(1-f)t_S/p} = \frac{p}{1+(p-1)f}$ where f is the fraction of the total Execution time which has to be performed serially.

   (c) Speedup:

   - $S(6) = \frac{6}{1+(6-1)\cdot 0.1} = \frac{6}{1.5} = 4$
   - $S(p) = \frac{p}{1+(p-1)f} = \frac{1}{1/p+f(p-1)/p} \to \frac{1}{f}$ hence $S = 10$

   (d) Speedup:

   - $S(6) = \frac{6}{1+(6-1)\cdot 0.2} = \frac{6}{2} = 3$
   - $S(p) = \frac{p}{1+(p-1)f} = \frac{1}{1/p+f(p-1)/p} \to \frac{1}{f}$ hence $S = 5$

   (e) $f(p) = \frac{p/S(p)-1}{p-1}$ and hence $f(64) = \frac{64/10-1}{63} = 8.6\%$

2. **Exercise 2**

   The graph located in task2/results.pdf shows average execution times where the programs slow medium and fast correspond to the first, second and third column respectively. The first line shows data points from the lcc2 cluster whereas the second third and fourth lines show the results for my laptop and various optimization flags. The errorbars show the computed standard error which is only an indirect measure for stability of the times (standard error=standard deviation / $\sqrt{}$ number of iterations). The number of iterations is shown in the table below. Optimization flags improve times significantly for the program compiled from fast.c. All execution times of the other implementations remain roughly the same.

number of iterations for	slow	medium	fast
lcc2	4	20	50
laptop	10	30	100