Institute for Computer Science, Friedrich Schiller University Jena **fmi.uni-jena.de**
H. Martin Bücker, Ralf Seidler **{martin.buecker,ralf.seidler}@uni-jena.de**

Parallel Computing 2, Summer 2015

Exercise 3

Problem 3.1 (Kung 1974)**.** Let $x \in \mathbb{R}$ and $n = 2^\nu$ for some $\nu \in \mathbb{N}$. Formulate a parallel algorithm for the computation of x^n. How many processors does your algorithm use? What is its time complexity?

Hint: Do not use the identity $x^n = \exp(n \ln x)$, but the approach based on

$$\frac{n}{x^n - 1} = \sum_{k=1}^{n} \frac{\omega^k}{x - \omega^k} \ ,$$

where $\omega = \exp(-2\pi i/n)$ and $i = \sqrt{-1}$. Suppose that the values for the powers of unity ω^k for $k = 1, 2, \ldots, n$ are given in some pre-calculated form.

Conclusion: *To derive a parallel algorithm, it is not mandatory to "follow the traditional serial path."*

Problem 3.2 (Cyclic reduction)**.** Assuming that n is an exact power of two, cyclic reduction is applicable to add n numbers in $\log n$ steps using $n/2$ processors. How can the principle of cyclic reduction be adapted to the situation where only $n/\log n$ processors are available? How does the speedup and efficiency change in this situation?

Hint: Assume that $\log n$ is an integer and that $n/\log n$ is an exact power of two. Consider exclusively the cost of arithmetic operations.

Conclusion: *Minor changes may lead to a dramatically changing efficiency.*