Speedup for Multi-Level Parallel Computing

Shanjiang Tang, Bu-Sung Lee, Bingsheng He

School of Computer Engineering

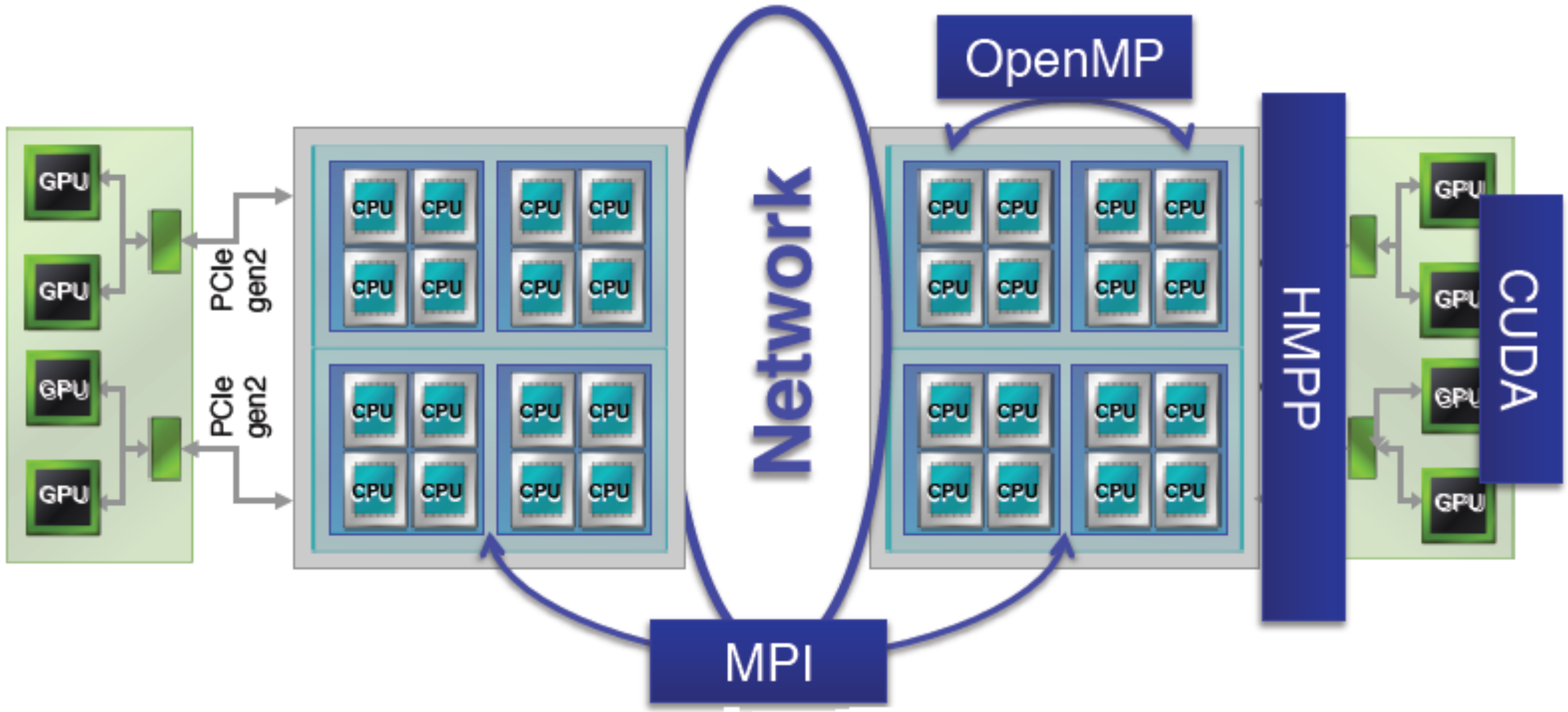
Nanyang Technological University

21st May 2012

OutLine

- Background & Motivation
- Multi-Level Parallel Speedup
- Evaluation
- Conclusion

Multi-Level Computing Architecture and Paradigm



NANYANG TECHNOLOGICAL UNIVERSITY

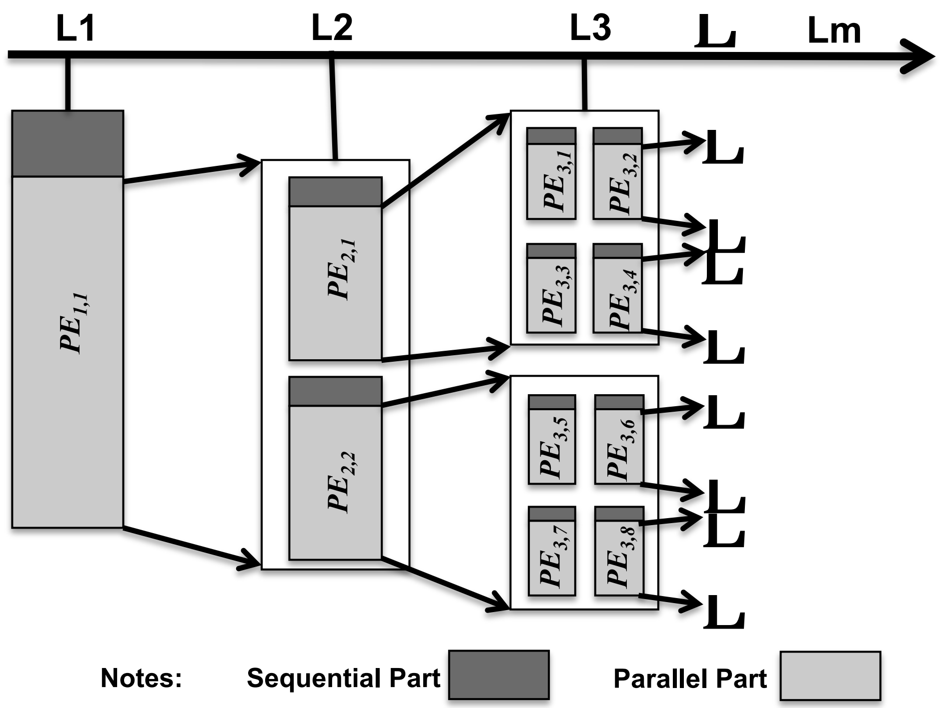
Multi-Level Computing Architecture and Paradigm

- MPI+OpenMP
- MPI+CUDA
- MPI+OpenMP+CUDA

  …..

NANYANG TECHNOLOGICAL UNIVERSITY

Multi-Level Parallel Computing Model



Parallel Speedup

- Definition

$$Speedup = \frac{SequentialExecutionTime}{ParallelExecutionTime}$$

- Classification
  - Absolute Speedup

$$Speedup = \frac{BestSequentialALGExecutionTime}{ParallelALGExecutionTime}$$

  - Relative Speedup

$$Speedup = \frac{ParallelALGSequentialExecutionTime}{ParallelALGExecutionTime}$$

NANYANG TECHNOLOGICAL UNIVERSITY

Relative Speedup Model

- Fixed-size Speedup

➢Amdahl's Law

$$Speedup = \frac{sequentialTime}{parallelTime} = \frac{1}{1-\alpha+\frac{\alpha}{p}}$$

where α is parallel fraction workload of the program, p is the number of processors.

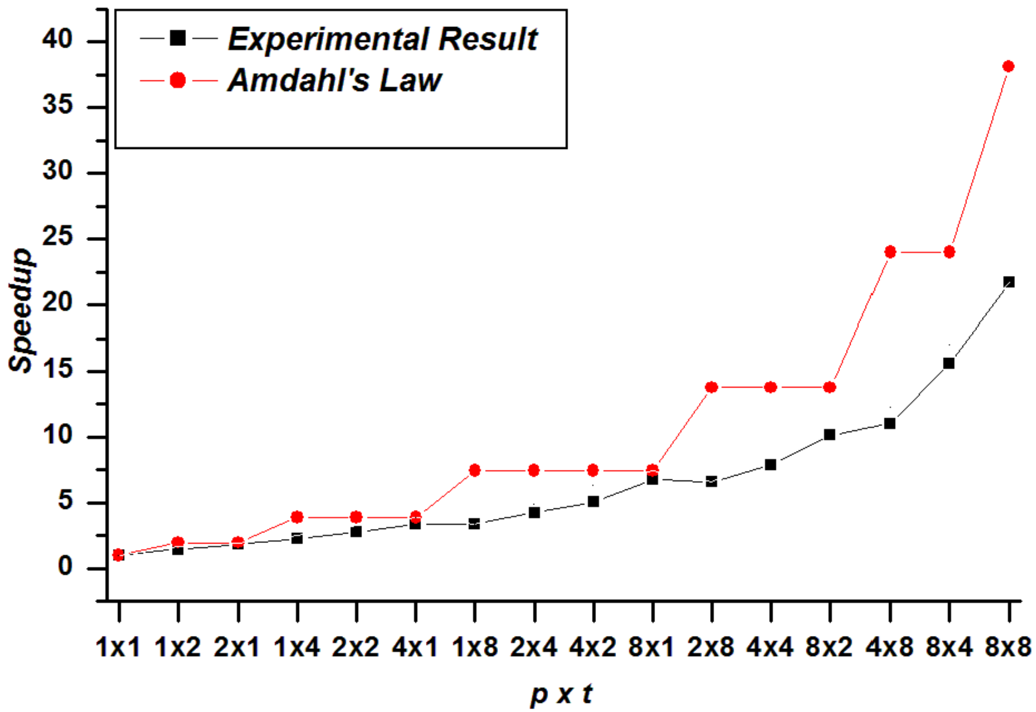
- Fixed-time Speedup

➢Gustafson's Law

$$Speedup = \frac{sequentialTime}{parallelTime} = \frac{1-\alpha+\alpha p}{1-\alpha+\alpha} = 1-\alpha+\alpha p$$

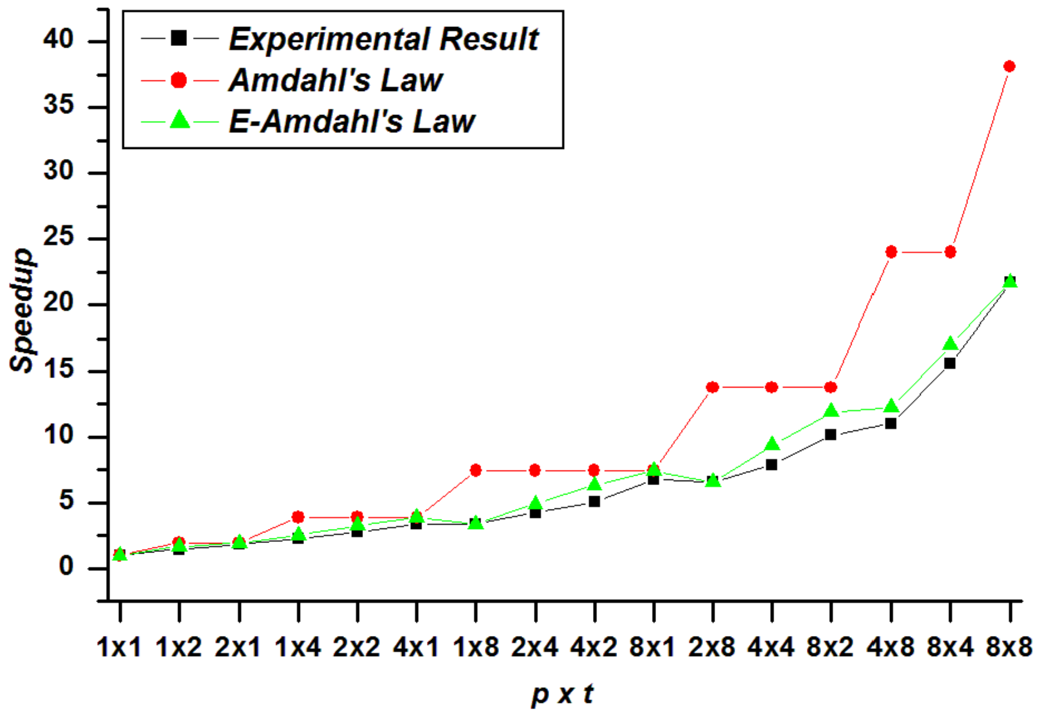
NANYANG TECHNOLOGICAL UNIVERSITY

Motivation Example—NAS Benchmark (MPI+OpenMP)



NANYANG TECHNOLOGICAL UNIVERSITY

Motivation Example—NAS Benchmark (MPI+OpenMP)



Amdahl's Law is UNSUITABLE for Multi-Level Parallel Computing

NANYANG TECHNOLOGICAL UNIVERSITY

OutLine

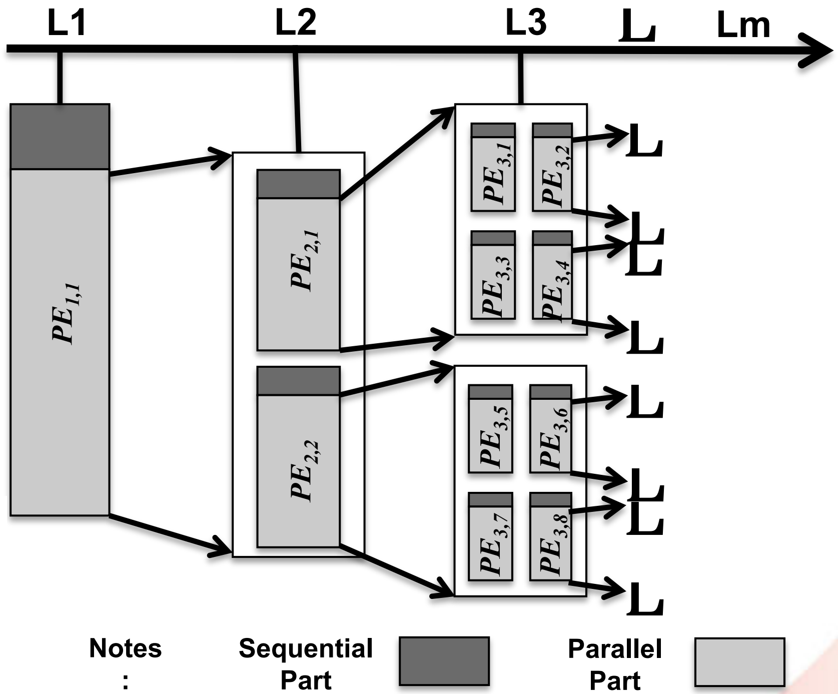
- Background & Motivation
- Multi-Level Parallel Speedup
- Evaluation
- Conclusion

E-Amdahl's Law

- Awareness of Different Grained-Level Parallelism

$$sp(i) = \begin{cases} \dfrac{1}{1 - f(m) + \dfrac{f(m)}{p(m)}} & (i = m) \\ \dfrac{1}{1 - f(i) + \dfrac{f(i)}{p(i)sp(i+1)}} & (1 \le i < m) \end{cases}$$

Symbol	Definition
m	The number of nested parallelism levels. $(m \geq 1)$
$p(i)$	The number of parallel processing elements in the i^{th}. level. $(p(i) \geq 1)$
$f(i)$	The portion of workload in the i^{th} level that can be parallelized. $(0 \leq f(i) \leq 1)$.
$sp(i)$	The multi-level speedup for the i^{th} level.



NANYANG TECHNOLOGICAL UNIVERSITY

E-Amdahl's Law

- Two-Level Parallelism Speedup Model (MPI+OpenMP)

$$sp(\alpha,\beta,p,t)=\frac{1}{1-\alpha+\dfrac{\alpha(1-\beta+\dfrac{\beta}{t})}{p}}$$

where

α is the parallel fraction of coarse-grained (MPI-level) parallelism.

β is the parallel fraction of fine-grained (OpenMP-level) parallelism.

p is the number of processes spawned.

t is the number of threads spawned per process.

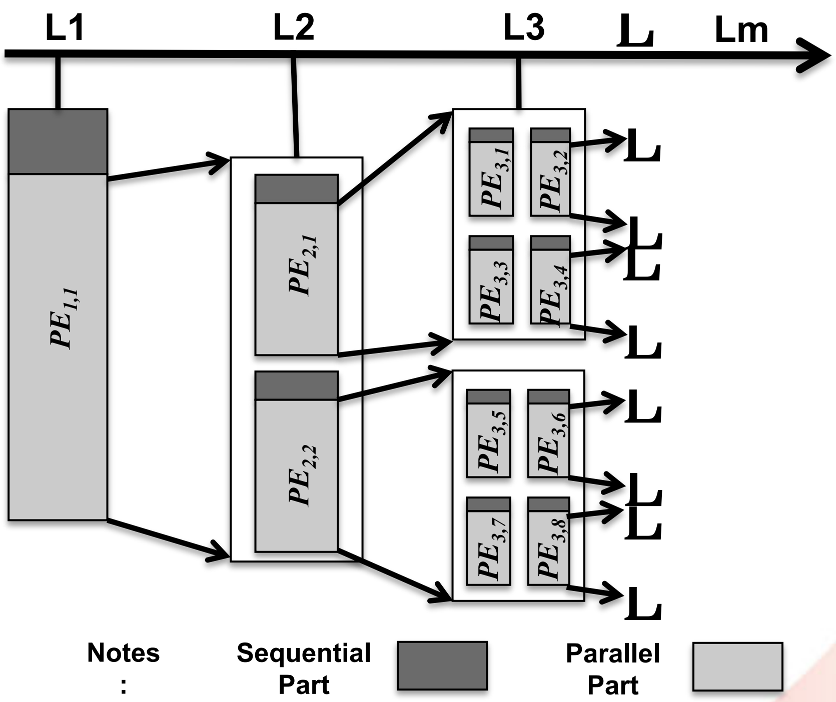
NANYANG TECHNOLOGICAL UNIVERSITY

E-Gustafson's Law

- Awareness of Different Grained-Level Parallelism

$$sp(i) = \begin{cases} 1 - f(m) + f(m)p(m) & (i = m) \\ 1 - f(i) + f(i)p(i)sp(i+1) & (1 \le i < m) \end{cases}$$

Symbol	Definition
m	The number of nested parallelism levels. $(m \geq 1)$
$p(i)$	The number of parallel processing elements in the i^{th}. level. $(p(i) \geq 1)$
$f(i)$	The portion of workload in the i^{th} level that can be parallelized. $(0 \leq f(i) \leq 1)$.
$sp(i)$	The multi-level speedup for the i^{th} level.



OutLine

- Background & Motivation
- Multi-Level Parallel Speedup
- Evaluation
- Conclusion

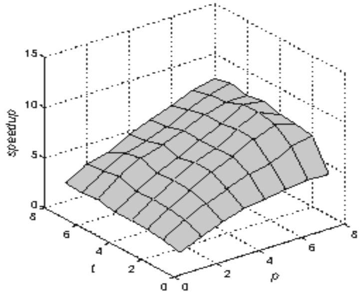
Experiment Setup

- Platform and Configuration
  - ➢ A linux cluster consisting of eight computing nodes each with two quad-core chips
  - ➢ Configuration: One thread per CPU core

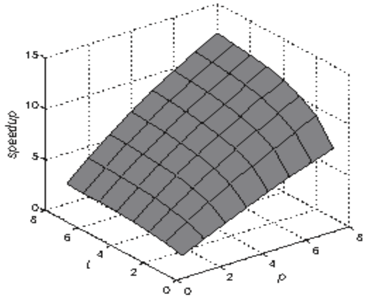
- Benchmarks

  NAS Parallel Benchmark (NPB) Multi-Zone (MZ) Version:
  - ➢ BT-MZ (Unbalanced Workload Partitioning)
  - ➢ SP-MZ (balanced Workload Partitioning)
  - ➢ LU-MZ (balanced Workload Partitioning)

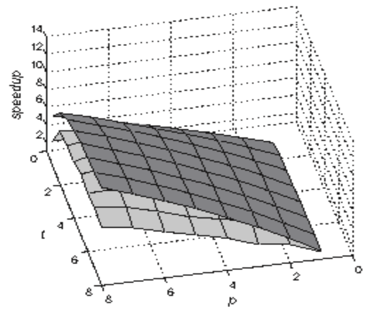
Performance Prediction



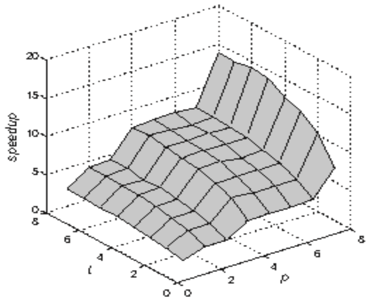
(a) Experimental result of BT-MZ (class W).



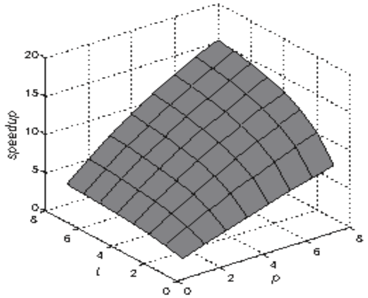
(b) Estimated result of BT-MZ. $\alpha = 0.9771, \beta = 0.5822$



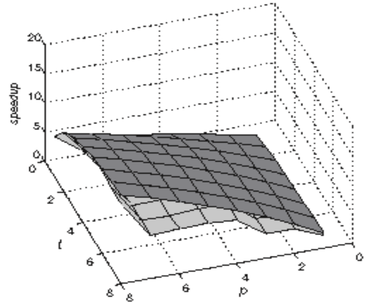
(c) Comparison result of (a) and (b).



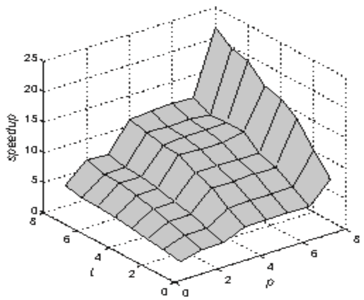
(d) Experimental result of SP-MZ (class A).



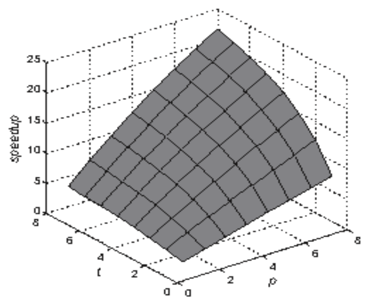
(e) Estimated result of SP-MZ. $\alpha = 0.9790, \beta = 0.7263$



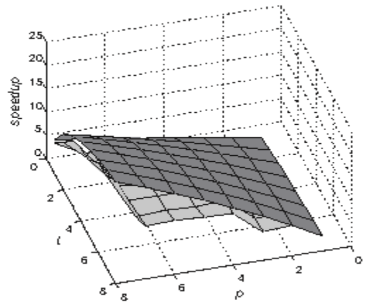
(f) Comparison result of (d) and (e).



(g) Experimental result of LU-MZ (class A).

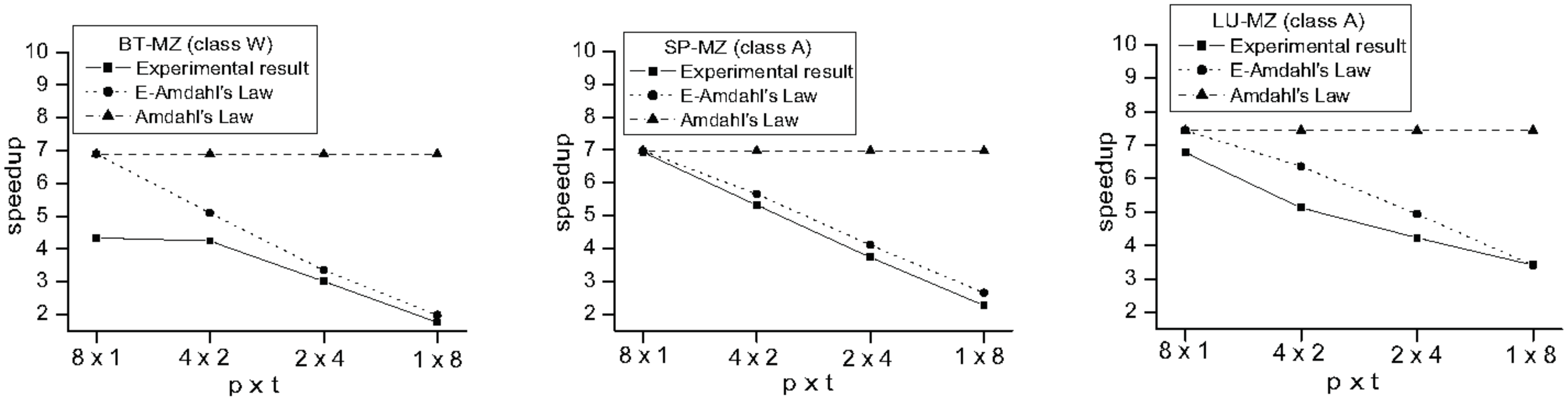


(h) Estimated result of LU-MZ. $\alpha = 0.9892, \beta = 0.8161$



(i) Comparison result of (g) and (h)

Prediction Result Comparison



(a) $\alpha = 0.9771, \beta = 0.5822$ (b) $\alpha = 0.9790, \beta = 0.7263$ (c) $\alpha = 0.9892, \beta = 0.8161$

Fig. 8: Experimental and estimated speedups of NPB-MZ for different combinations of $p \times t$ under a given total number of 8 processors. The speedup based on *Amdahl's Law* is estimated with the formula $\frac{1}{1-\alpha+\frac{\alpha}{p \times t}}$. The speedup based on *E-Amdahl's Law* is estimated by using Formula (17).

NANYANG TECHNOLOGICAL UNIVERSITY

OutLine

- Background & Motivation
- Multi-Level Parallel Speedup
- Evaluation
- Conclusion

Conclusion

- Traditional speedup models are unsuitable for multi-level parallelism
  - Unable to be awareness of different granularities of parallelism for multi-level parallel computing.
- Multi-level Parallelism Model
  - A guidance model for multi-level optimization.
  - A prediction model for multi-level parallelism.

NANYANG TECHNOLOGICAL UNIVERSITY

Thank You !

Question?

NANYANG
TECHNOLOGICAL
UNIVERSITY

Argument Estimation

Algorithm 1 *Argument Estimation for* α, β.

1: Execute the program k times in the multi-level parallel execution mode with the chosen parameters $(p_1, t_1), (p_2, t_2), ..., (p_k, t_k)$ respectively. Then it gets $(p_1, t_1, sp_1), (p_2, t_2, sp_2), ..., (p_k, t_k, sp_k)$.

2: Choose all possible combinations of two arrays (p_i, t_i, sp_i) and (p_j, t_j, sp_j) to figure out the value (α_s, β_s) based on Equation (17), where s denotes the s^{th} combination and $1 \le i, j \le k$ & $i \ne j$.

3: Check all possible pairs of value (α_s, β_s) to guarantee that the pair of estimated values is valid(i.e. $0 \le \alpha_s \le 1, 0 \le \beta_s \le 1$). Otherwise, discards it.

4: Collect all valid pairs of $(\alpha_i, \beta_i), (i = 1, 2, ..., k')$, where k' represents the number of valid pairs. Remove the noise pairs by clustering with the guard condition: $|\alpha_i - \alpha_j| < \varepsilon$ & $|\beta_i - \beta_j| < \varepsilon$.

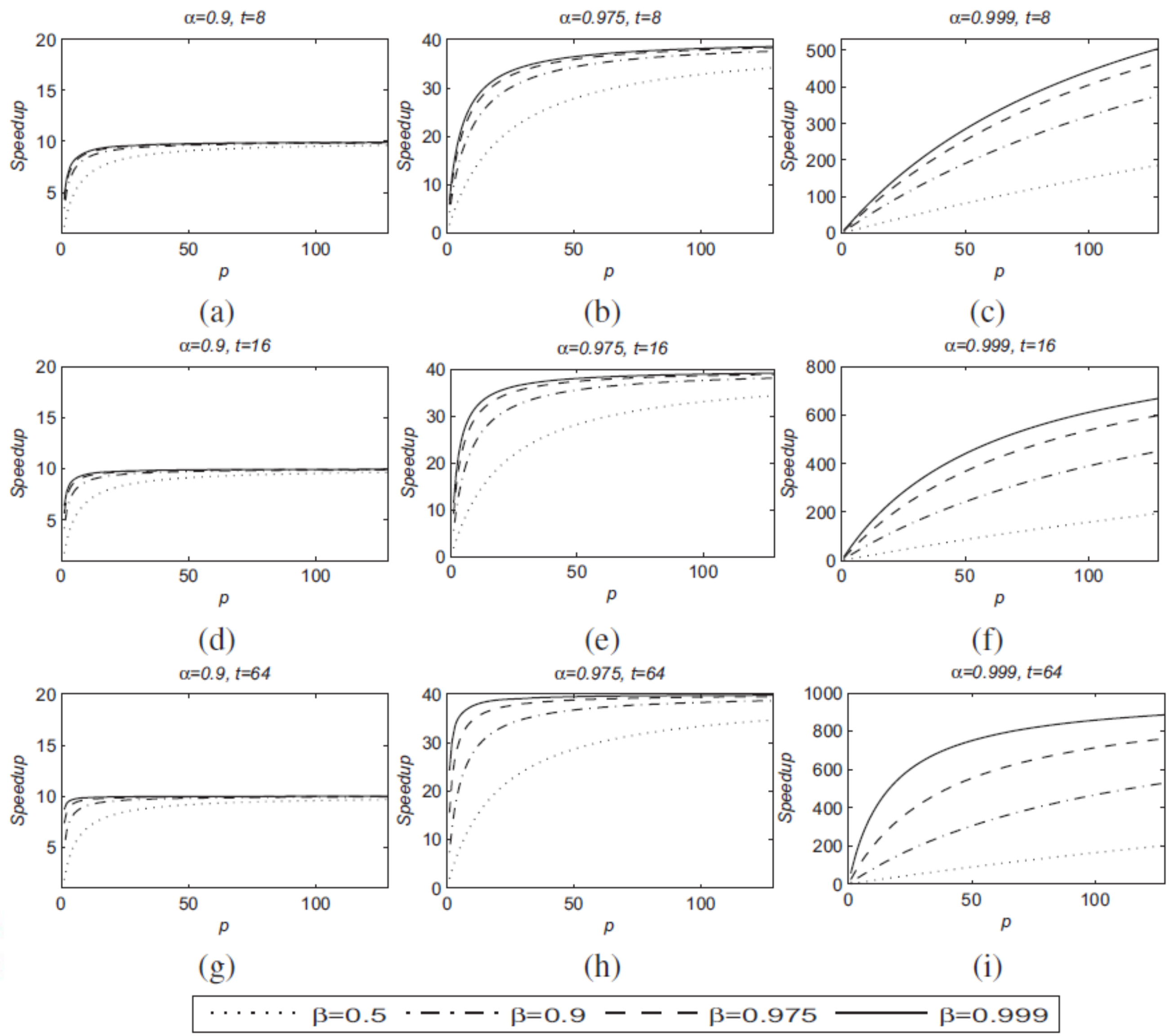
5: The exact value of α, β can thereby be estimated with the formula:

$$\begin{cases} \widehat{\alpha} = \frac{1}{\widehat{k}} \sum_{i=1}^{\widehat{k}} \alpha_i \\ \widehat{\beta} = \frac{1}{\widehat{k}} \sum_{i=1}^{\widehat{k}} \beta_i, \end{cases}$$

where $\widehat{k}$ is the number of clustered pairs.

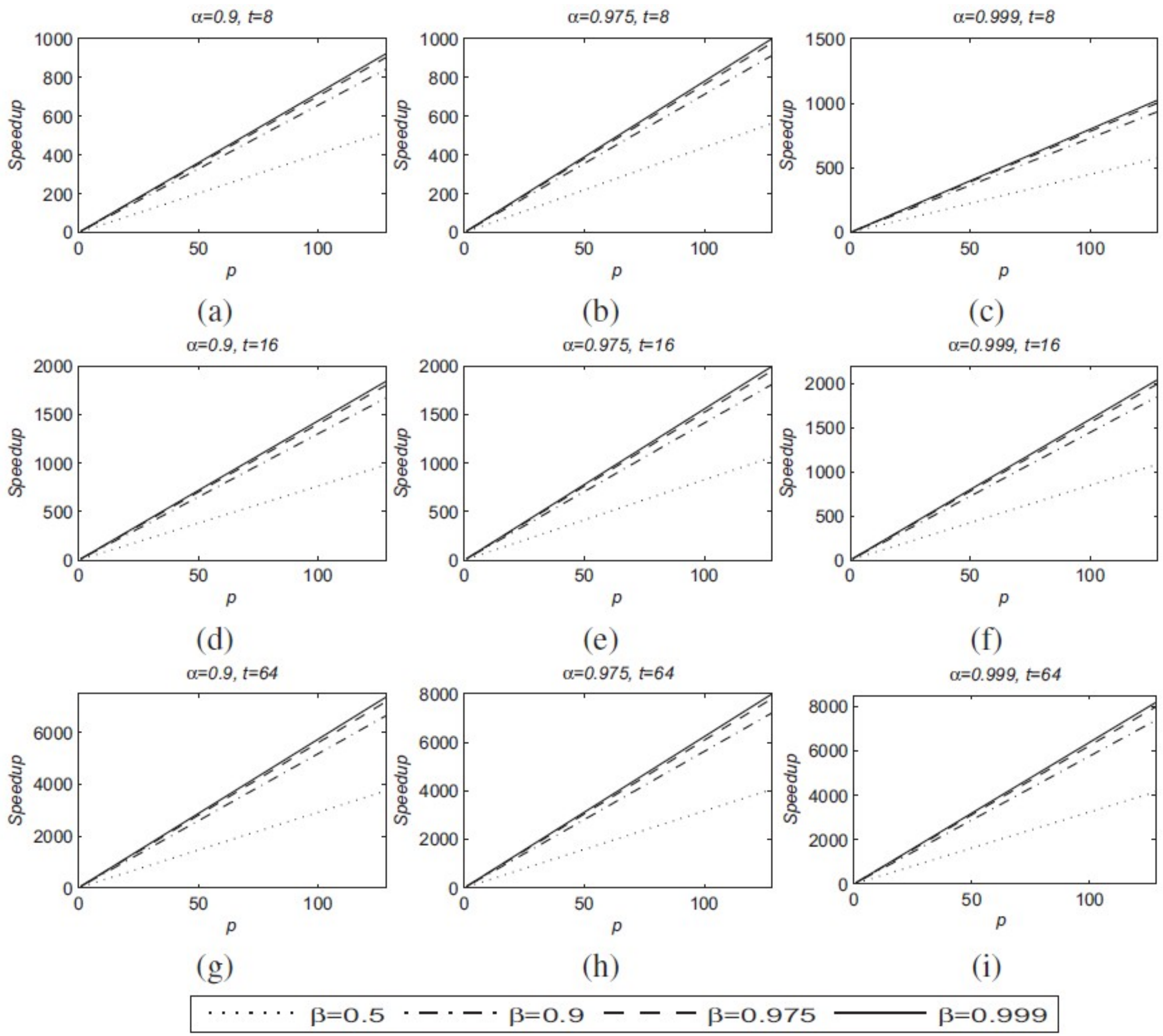
NANYANG TECHNOLOGICAL UNIVERSITY

Speedup Under E-Amdahl's Law



NANYANG TECHNOLOGICAL UNIVERSITY

Speedup Under E-Gustafson's Law



NANYANG TECHNOLOGICAL UNIVERSITY