Reliability-Aware Scalability Models for High Performance Computing

Ziming Zheng, Zhiling Lan

Illinois Institute of Technology

Scalability model

- Systems are getting bigger and faster
  - 50.2% of high-end systems > 4096 processors *[http://www.top500.org]*
- Scalability is a key factor for evaluating, predicting and optimizing the performance
- Amdahl's law and Gustafson's law are well-known scalability models
- Both laws implicitly assume that the application can complete without experiencing any failure

Failure is commonplace!

ILLINOIS INSTITUTE OF TECHNOLOGY

Reliability issue

- Failure becomes a commonplace scenario instead of an exception *[B. Schroeder DSN 06]*
  - Failure rates are more than 1000 per year
  - Failure repair time is up to nearly 100 hours

- Scalability in the presence of failures ≠ scalability in the ideal failure-free environments

- Checkpointing (CKP) has been widely used for reliability

Scalabilty is impacted by failures and CKP

Outline

- Extend Amdahl’s law and Gustafson’s law
  - Considering failures
  - Considering checkpointing
- Assess the models via trace-based simulations
- Use the models to evaluate fast recovery and proactive failure prevention

Assumptions

- The time interval between failures on node i is exponentially distributed with an arrival rate of λ_i
- The failure arrival rate of *P* nodes is $\lambda_P = \sum_{i=1}^{P} \lambda_i$ (homogeneous systems $\lambda_P = P\lambda$)
- Repair time follows a general distribution with a mean of μ and is insensitive *to* P
- One unit of workload takes one unit of time per node

ILLINOIS INSTITUTE OF TECHNOLOGY

Nomenclature

P	Number of computing nodes or processes
λ_i	Failure arrival rate of node i
λ_P	Failure arrival rate of the P nodes allocated to the application(hour)
μ	Mean-Time-To-Recover(MTTR) (hour)
W	Application workload, the application failure-free operation count on a single node
W'	The scaled workload on a single node
W_p	The parallel workload
α	The fraction of the application that can be parallelized
O_{ckp}	Checkpoint overhead (hour)
T	Checkpoint interval (hour)
S^A	Amdahl's scalability model without checkpointing
S_f^A	Agumented Amdahl's scalability model without checkpointing
$S_{f,c}{}^A$	Agumented Amdahl's scalability model with checkpointing
S^G	Gustafson's scalability model without checkpointing
S_f^G	Agumented Gustafson's scalability model without checkpointing
$S_{f,c}{}^G$	Agumented Gustafson's scalability model with checkpointing

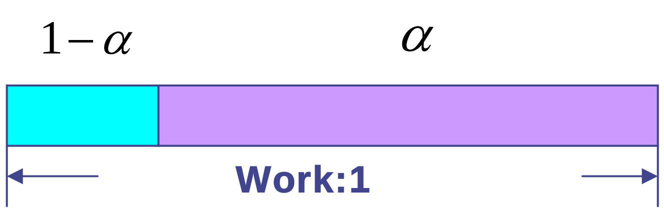
A: Amdahl's model; f: under failure; c: with checkpointing

Amdahl's law

- Gene M. Amdahl, "*Validity of the Single-Processor Approach to Achieving Large Scale Computing Capabilities*", 1967
- Amdahl's law (Amdahl's speedup model)

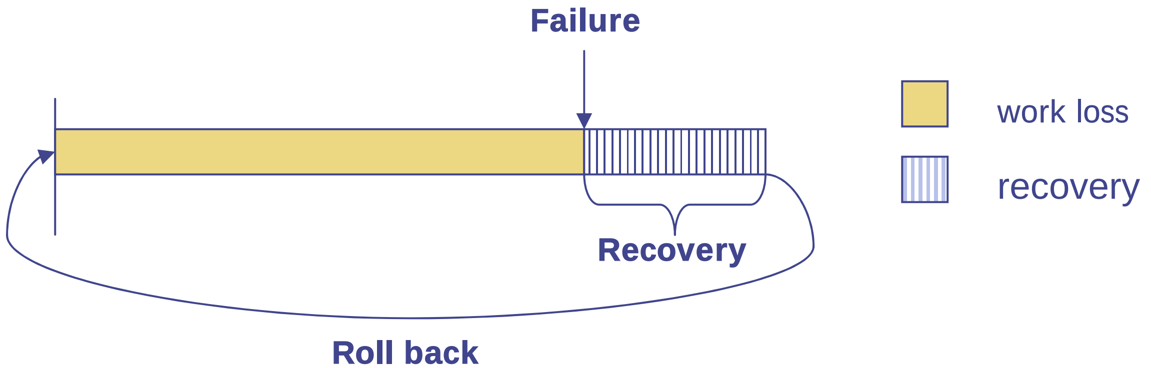
$$W_p = (1-\alpha)W + \alpha \frac{W}{P}$$

$$S^A = \frac{T_s}{T_p} = \frac{W}{W_p} = \frac{P}{P(1-\alpha)+\alpha}$$



Agumented Amdahl's model W/O checkpointing

- Without checkpointing, when a failure occurs the application will roll back to the beginning



- Expected time under failure $E(T_f(W_p)) = (\mu + \lambda_P^{-1})(e^{(1-\alpha+\frac{\alpha}{P})\lambda_P W} - 1)$

$$S_f^A = \frac{W}{E(T_f(W_p))} = \frac{W}{(\mu + \lambda_P^{-1})(e^{(1-\alpha+\frac{\alpha}{P})\lambda_P W} - 1)}$$

ILLINOIS INSTITUTE OF TECHNOLOGY

Augmented Amdahl's model W/O checkpointing

- S^A is a special case of S_f^A when $\mu = 0,\ \lambda_P = P\lambda,\ 1/P\lambda >> W_p$

$$S_f^A = \frac{W}{(P\lambda)^{-1}(e^{(1-\alpha)P\lambda W} e^{\alpha\lambda W} - 1)}$$

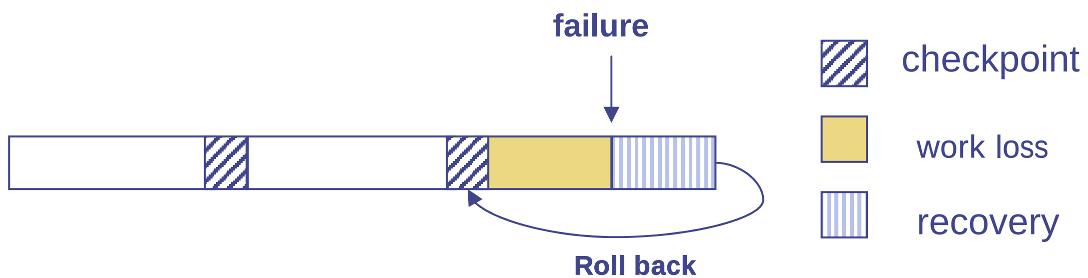
$$\approx \frac{W}{(P\lambda)^{-1}((1-\alpha)P\lambda W + \alpha\lambda W)}$$

$$= S^A$$

- In a failure-prone environment, S_f^A is different from S^A
  - S^A is independent of W
  - S_f^A exponentially decreases with the growth of W
  - Application with high workload is more vulnerable to failures

Augmented Amdahl's model W/ checkpointing

- Upon a failure the application will be restarted from the most recent checkpoint



- Daly's model is adopted to estimate the expected parallel execution time with checkpointing $E(T_{f,c}(W_p))$

$$E(T_{f,c}(W_p)) = \frac{e^{u\lambda_P}}{\lambda_P}\left(e^{(\tau+O_{ckp})\lambda_P} - 1\right)\frac{(1-\alpha+\frac{\alpha}{P})W}{\tau}$$

$$\tau = \begin{cases} \sqrt{\frac{2O_{ckp}}{\lambda_P}}\left[1+\frac{1}{3}\left(\frac{O_{ckp}\lambda_P}{2}\right)^{1/2}+\frac{1}{9}\left(\frac{O_{ckp}\lambda_P}{2}\right)\right] - O_{ckp} & O_{ckp} < \frac{2}{\lambda_P} \\ \frac{1}{\lambda_P} & O_{ckp} \ge \frac{2}{\lambda_P} \end{cases}$$

ILLINOIS INSTITUTE OF TECHNOLOGY

Augmented Amdahl's model W/ checkpointing

$$S_{f,c}^{A} = \frac{T_s}{E(T_{f,c}(W_p))} = \frac{\cancel{W}}{\frac{e^{u\lambda_P}}{\lambda_P}(e^{(\tau+O_{ckp})\lambda_P}-1)\frac{(1-\alpha+\frac{\alpha}{P})\cancel{W}}{\tau}}$$

$$= \frac{\lambda_P \tau}{e^{\mu\lambda_p}(e^{(\tau+O_{ckp})\lambda_P}-1)(1-\alpha+\frac{\alpha}{P})}$$

- ${S_{f,c}}^{A}$ is independent of W
- Checkpointing is helpful to maintain application scalability with high workload

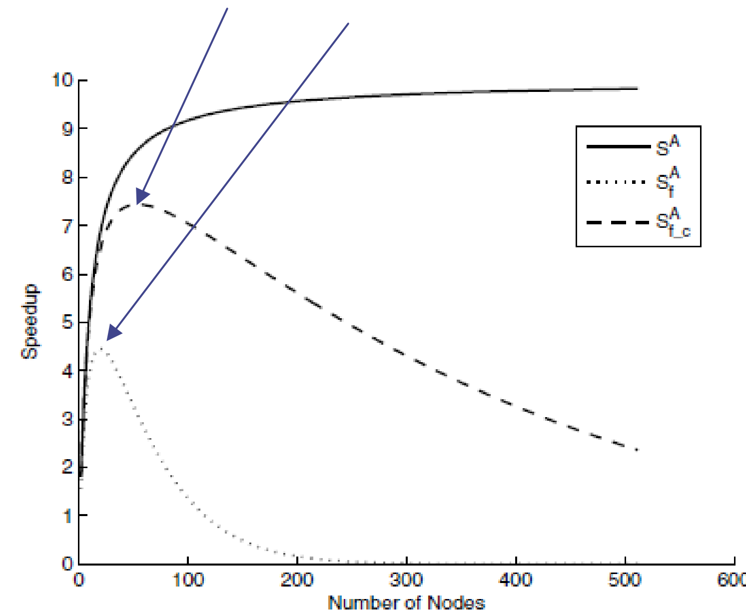
A Use Scenario

- S^A is monotonically increases with the growth of P, with an upper bound of $\frac{1}{1-\alpha}$
- S_f^A and $S_{f,c}{}^A$ may decrease with the growth of P

- Reliability-aware models can identify the optimal P

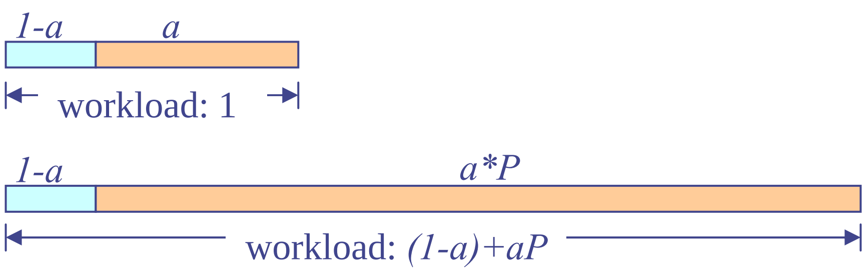
- Checkpointing increases the maximal achievable speedup

Maximal speedup



Gustafson's law

- J. Gustafson, "Reevaluating Amdahl's law", 1988
- Fix-time speedup
  - Emphasizes on the amount of workload that can be finished in a fixed time

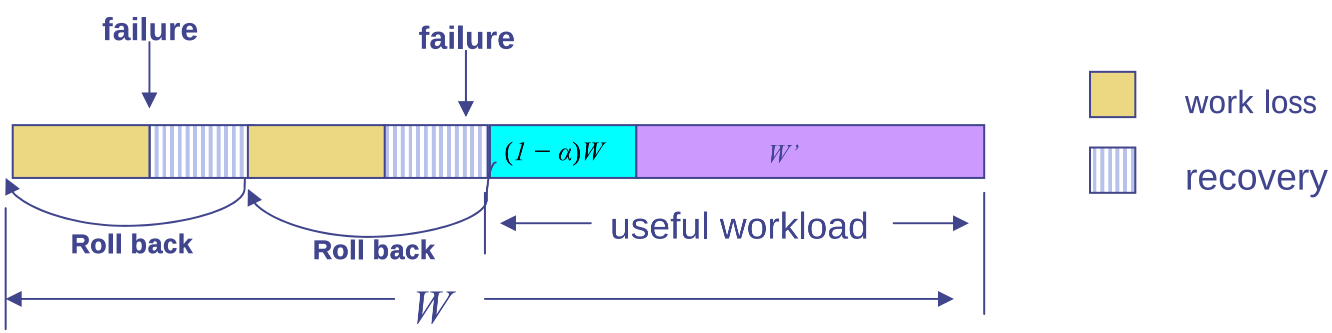


$$S^G = \frac{(1-\alpha)W + \alpha WP}{W} = 1 - \alpha + \alpha P$$

  - S^G is independent of W and linearly grows with P.

Augmented Gustafson's model W/O checkpointing

- As W increases, the application gets more vulnerable to failures
- *useful workload (per node)* = W − *work loss* − *recovery*
- scaled workload W'= *useful workload* − $(1 - \alpha)W$



$$S_f^G = \frac{(1-\alpha)W + W'P}{W} = 1-\alpha + \frac{P\ln\left(\frac{W}{\mu+(\lambda_p)^{-1}}+1\right)}{W\lambda_P} - P(1-\alpha)$$

Augmented Gustafson's model W/O checkpointing

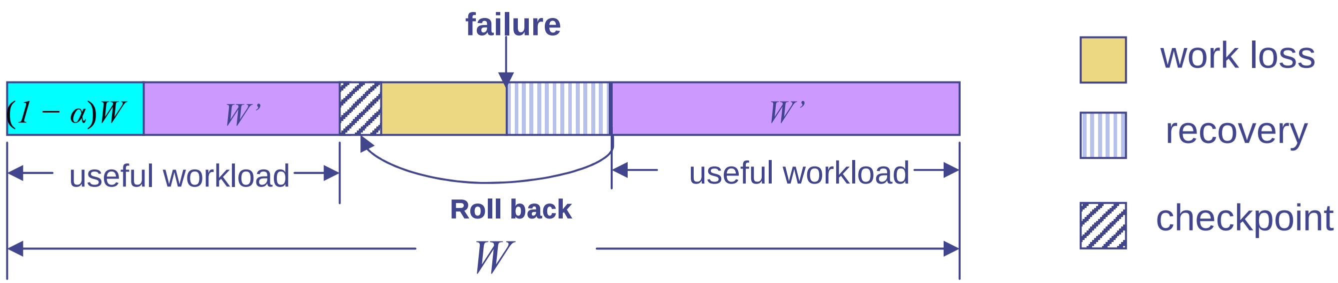
- S^G is a special case of S_f^G when $\mu = 0, \lambda_a = P\lambda, 1/P\lambda >> W$

$$\begin{aligned} S_f^G &= 1-\alpha+\frac{P\ln(WP\lambda+1)}{W\lambda}-P(1-\alpha) \\ &\approx 1-\alpha+\frac{WP\lambda}{W\lambda}-P(1-\alpha) \\ &= S^G \end{aligned}$$

- Without these conditions, S_f^G is different from S^G
  - S^G is independent of W
  - S_f^G decreases with the growth of W
  - work loss and recovery time significantly increase with the growth of W

Augmented Gustafson’s model W/ checkpointing

- useful workload=W − *work loss* − *recovery* − *overhead*



- scaled workload W'=*achievable workload* − $(1-\alpha)W$

$$S^{G}_{f,c} = \frac{(1-\alpha)W + W'P}{W} = \frac{(1-\alpha)\cancel{W} + \left(\frac{\tau P\lambda_P}{e^{u\lambda_P}\left(e^{(\tau+O_{ckp})\lambda_P}-1\right)} - (1-\alpha)\right)\cancel{W}P}{\cancel{W}}$$

$$= 1-\alpha + \frac{\tau P\lambda_P}{e^{u\lambda_P}\left(e^{(\tau+O_{ckp})\lambda_P}-1\right)} - (1-\alpha)P$$

- $S_{f,c}{}^{G}$ is independent of W

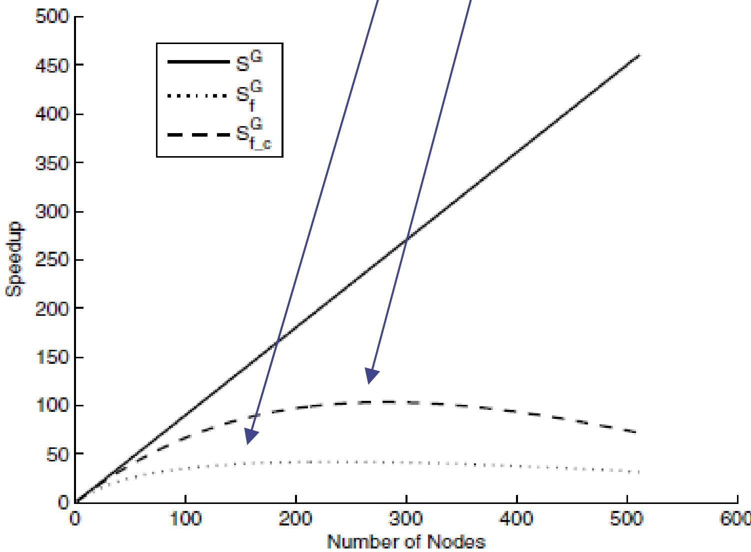
A Use Scenario

- S^G scales linearly with the number of nodes P
- S_f^G and $S_{f,c}^G$ are limited by failures and may decrease with the growth of P

Maximal speedup

- Reliability-aware models can identify the optimal P

- Checkpoint increases the maximal achievable speedup



Outline

- Extend Amdahl’s law and Gustafson’s law
  - Considering failures
  - Considering checkpointing
- Assess the models via trace-based simulations
- Use the models to evaluate fast recovery and proactive failure prevention

Evaluation

- Trace-based simulations to compare percentage prediction errors

$$\text{percentage prediction error} = \left|\frac{prediction - simulation}{simulation}\right|$$

- User provides the application-level parameters: workload W and the fraction α
- The system-level parameters are obtained from the trace fed into the simulator
  - The failure trace is from a production system (system #8) at Los Alamos National Lab (128 nodes with similar failure rates)

ILLINOIS INSTITUTE OF TECHNOLOGY

Augmented vs. Original Amdahl's Models

- Without checkpoint S^A(W/O CKP) vs. S_f^A
  - S_f^A is much more accurate than S^A
  - Further, as W increases the accuracy of S^A decreases dramatically
- With checkpoint S^A(W/ CKP) vs. $S_{f,c}{}^A$
  - The accuracy of $S_{f,c}{}^A$ outperforms S^A

W	S^A (W/O CKP)	S^A(W/ CKP)	S_f^A	$S_{f\_c}^A$
2000	1.99	0.14	0.27	0.09
5000	18.89	0.24	0.73	0.11
8000	20.55	0.16	0.87	0.04
10000	21.98	0.15	0.82	0.04

Percentage prediction errors

Augmented vs. Original Amdahl's Models

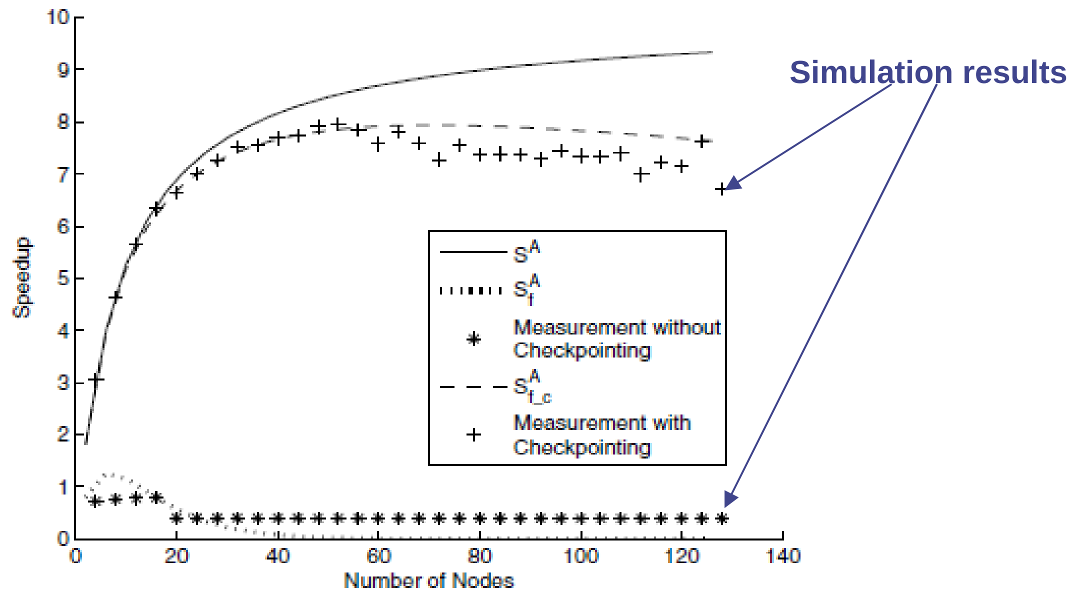
- S^A vs. S_f^A and $S_{f,c}{}^A$ with different α
  - The accuracy S^A is low when α is small
  - Even if α=0.999, application scalability under failures is still distinct from its scalability in the ideal failure-free environments

α	S^A (W/O CKP)	S^A (W/ CKP)	S_f^A	$S_{f\_c}^A$
0.7	15.66	0.52	0.95	0.34
0.75	8.73	0.34	0.95	0.19
0.8	10.67	0.28	0.27	0.14
0.9	21.98	0.15	0.82	0.04
0.95	23.83	0.24	0.79	0.10
0.999	3.32	0.07	0.93	0.06

Percentage prediction errors

Augmented vs. Original Amdahl's Models

- Compared to S^A, S_f^A and $S_{f,c}^A$ can better model application scalability in real environments
- The gap between S^A and the actual measurement becomes larger with the growth of P.



Augmented vs. Original Gustafson's Models

- Without checkpoint S^G(W/O CKP) vs. S_f^G
  - The useful workload in a failure-present environment is much less than that in an ideal failure-free environment
- With checkpoint S^G(W CKP) vs. $S_{f,c}^G$
  - The useful workload is not achievable as S^G due to the inevitable recovery process, work loss and checkpoint overhead

W	S^G (W/O CKP)	S^G (W/ CKP)	S_f^G	$S_{f\_c}^G$
1000	1.32	0.07	0.58	0.05
1200	15.7	0.13	2.3	0.01
1800	29.86	0.14	6.91	0.01
2000	12.71	0.14	2.33	0.01

Percentage prediction errors

Augmented vs. Original Gustafson's Models

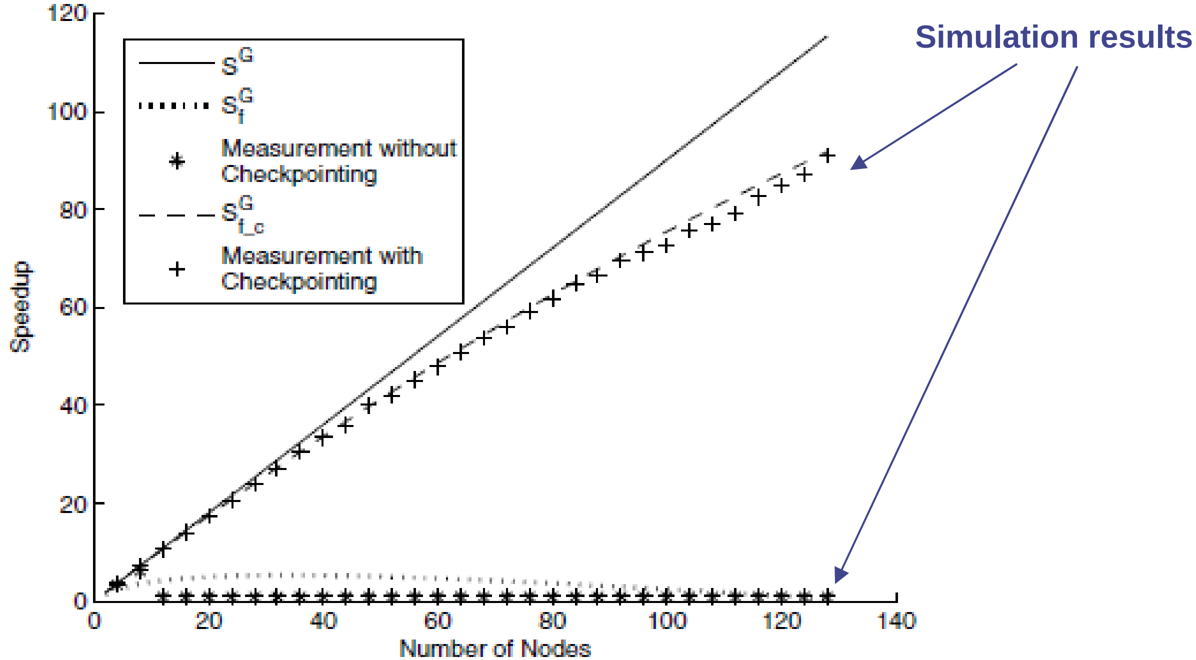
- S^G vs. S_f^G and $S_{f,c}{}^G$ with different α
  - S_f^G and $S_{f,c}{}^G$ outperform S^G
  - The error decreases with the growth of α

α	S^G(W/O CKP)	S^G (W/ CKP)	S_f^G	$S_{f\_c}^G$
0.7	3.70	0.12	1.43	0.05
0.8	1.9	0.08	0.80	0.06
0.9	1.32	0.07	0.58	0.05
0.95	1.15	0.06	0.51	0.05
0.999	1.01	0.06	0.46	0.05

Percentage prediction errors

Augmented vs. Original Gustafson's Models

- S_f^G and $S_{f,c}^G$ can better represent application scalability
- The gap between S^G and the actual measurement becomes larger with the growth of P

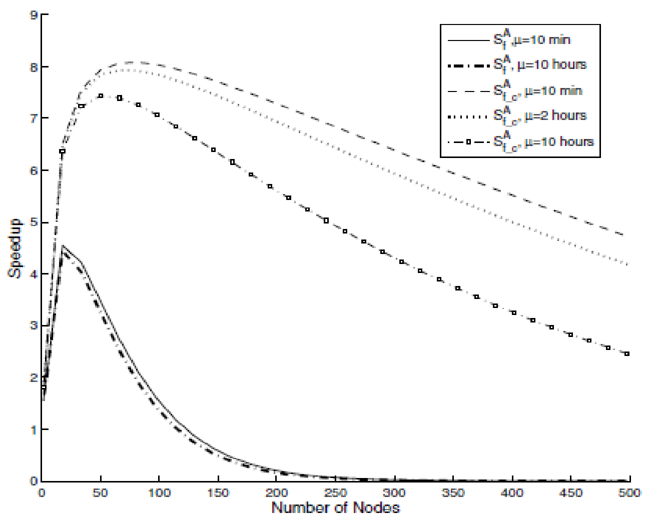


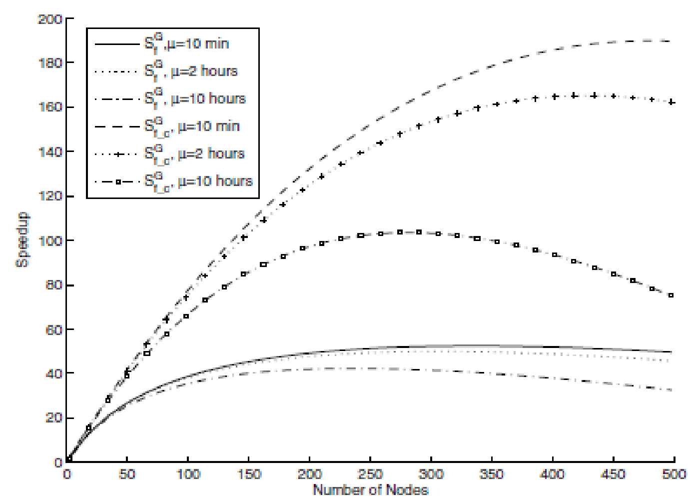
Outline

- Extend Amdahl’s law and Gustafson’s law
  - Considering failures
  - Considering checkpointing
- Assess the models via trace-based simulations
- Use the models to evaluate fast recovery and proactive failure prevention

Use of the models to assess fast recovery

- Fast recovery can reduce MTTR
  - Without checkpointing, fast recovery can not significantly improve application scalability
  - With checkpointing, fast recovery can significantly improve application scalability





Use of the models to assess failure prediction

- Based on failure prediction, proactive actions can prevent failure experiencing and avoid rollbacks
- Prediction accuracy

		Actual Data	
Predicted Result		**Fatal**	**Non-Fatal**
	Positive	**TP**	**FN**
	Negative	**FP**	**TN**

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

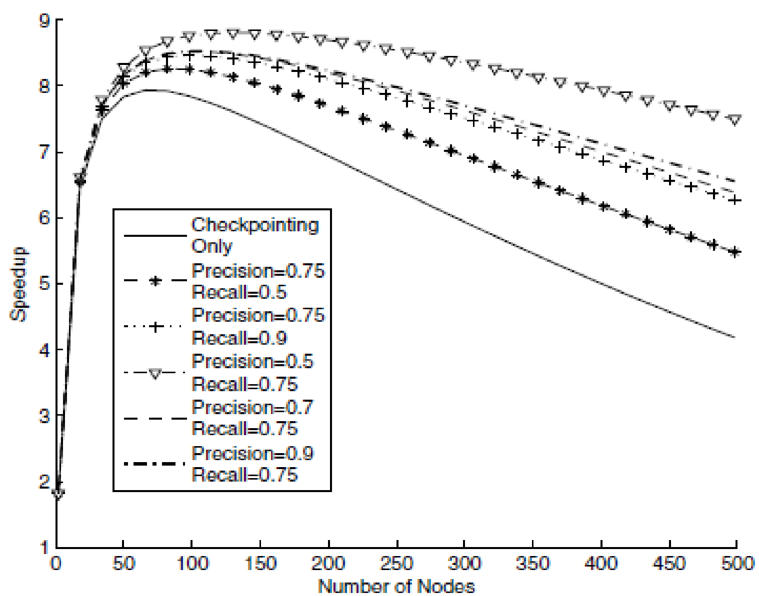
$$S_{f,c,m}^{A} = \frac{\lambda_P (1 - recall)\tau'}{e^{\mu\lambda_P(1-recall)}\left(e^{(\tau' + O_{ckp})\lambda_P(1-recall)} - 1\right)\left(1 - \alpha + \frac{\alpha}{P}\right)\left(1 + \frac{recall \times 2\lambda_P O_{ckp}}{precision}\right)}$$

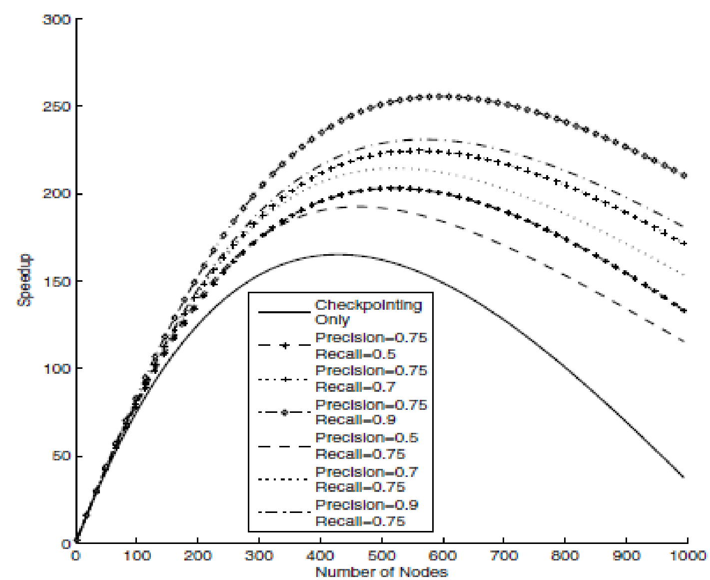
$$S_{f,c,m}^{G} = 1 - \alpha + \frac{P}{\frac{e^{\mu\lambda_P(1-recall)}}{\lambda_p(1-recall)}\left(e^{(\tau' + O_{ckp})\lambda_P(1-recall)} - 1\right)\frac{1}{\tau'} + \frac{recall \times 2\lambda_P O_{ckp}}{precision}} - (1-\alpha)P$$

ILLINOIS INSTITUTE OF TECHNOLOGY

Use of the models to assess failure prediction

- Recall can not only prevent work loss, but also reduce the frequency of checkpointing
- Precision only reduces unnecessary process migration overhead





Conclusions

- Have derive new reliability-aware scalability models by extending Amdahl's law and Gustafson's law
  - considering failures and fault tolerance techniques
- Trace-based simulations have demonstrated that these models can better represent application scalability in failure-present environments
- The models can be used to demonstrate the benefits of fast recovery and proactive failure prevention via process migration


Questions