WESTERN SYDNEY UNIVERSITY

Reliable and Energy-Efficient Resource Provisioning and Allocation in Cloud Computing

Yogesh Sharma, Bahman Javadi, Weisheng Si
School of Computing, Engineering and Mathematics
Western Sydney University, Australia

Daniel Sun
Data61-CSIRO, Australia

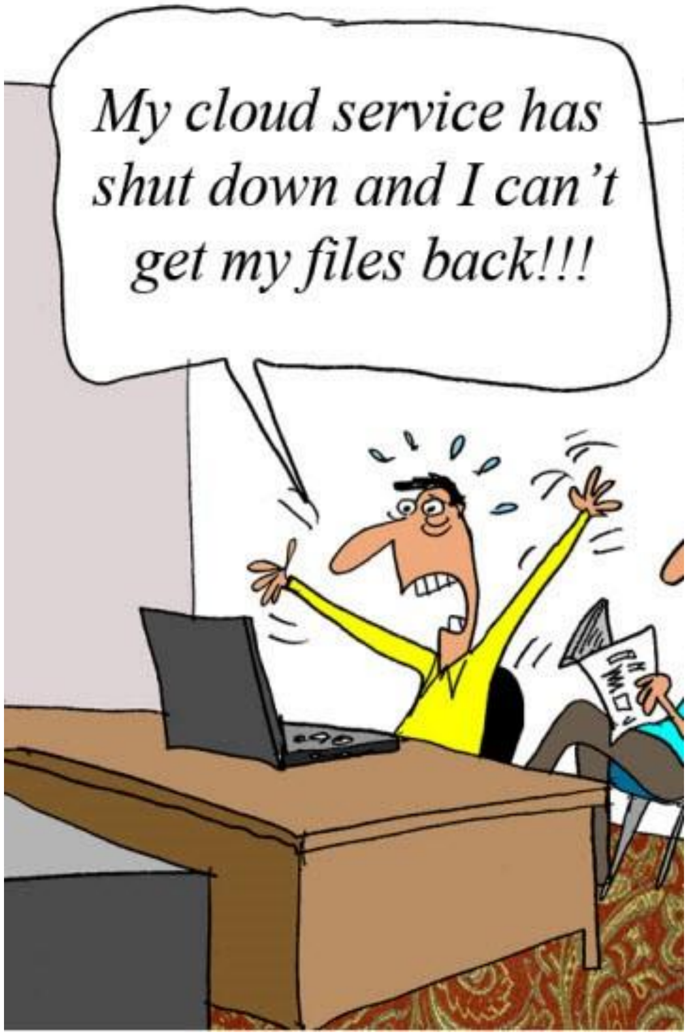
WESTERN SYDNEY UNIVERSITY

Agenda

1. Introduction
2. Reliability Model
3. Task Execution Model
4. Energy Model
5. Resource Provisioning and Allocation Policies
6. System Architecture
7. Simulation Configuration Parameters
8. Results and Conclusions

Reliability

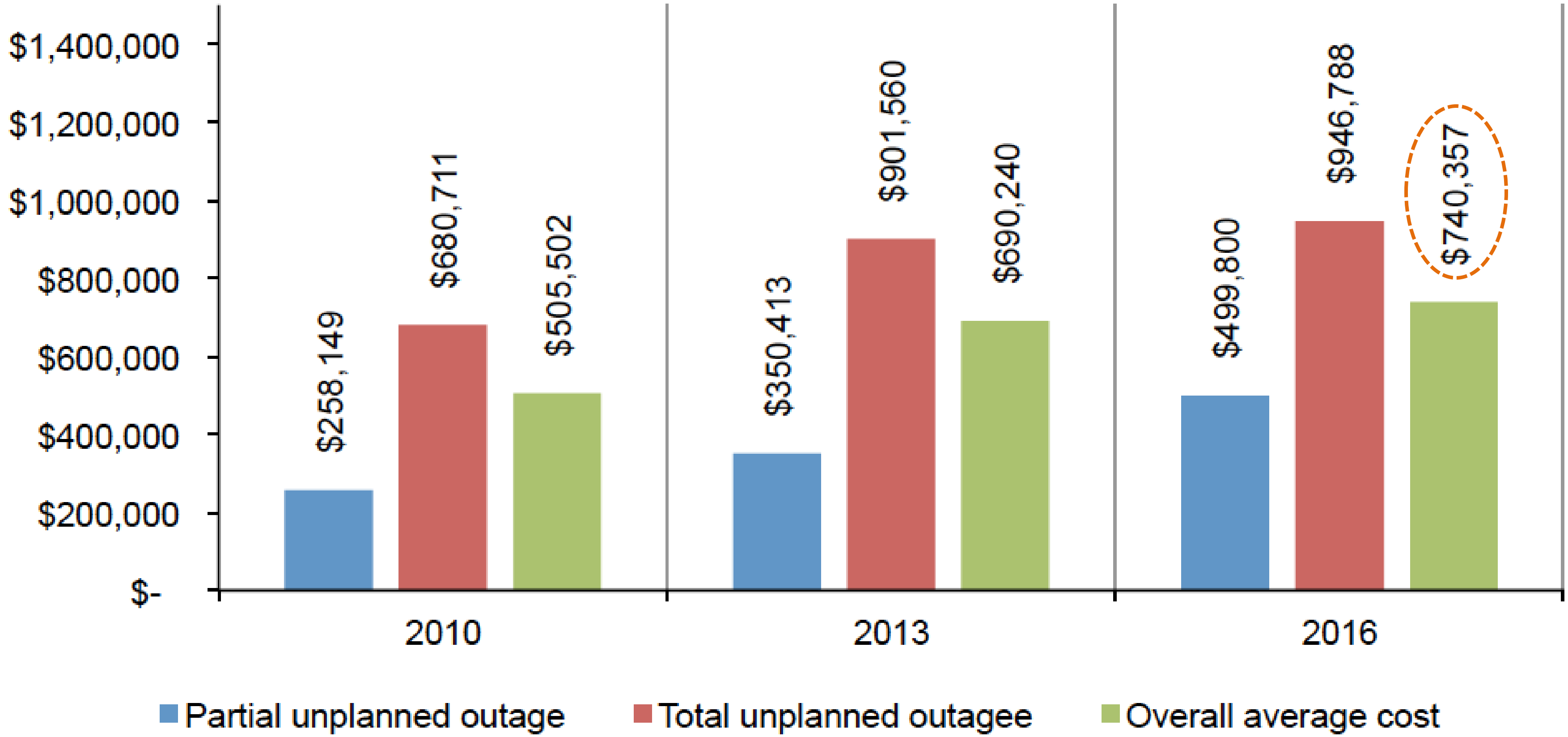
WESTERN SYDNEY UNIVERSITY



- Critical challenge in Cloud Computing environments.
- Service failures have huge impact on service providers such as:
  - Business Disruption
  - Lost Revenues
  - Customer Productivity Loss

WESTERN SYDNEY UNIVERSITY

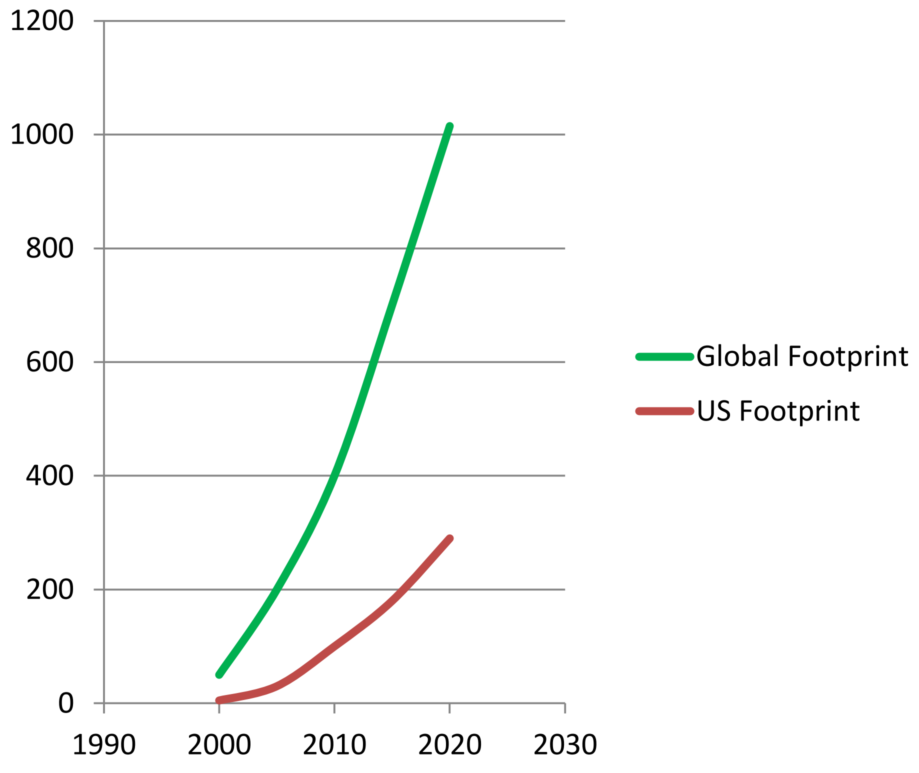
Cost of Cloud Outage



* Ref: Calculating the Cost of Data Center Outages, Ponemon Institute© Research Report, 2016

WESTERN SYDNEY UNIVERSITY

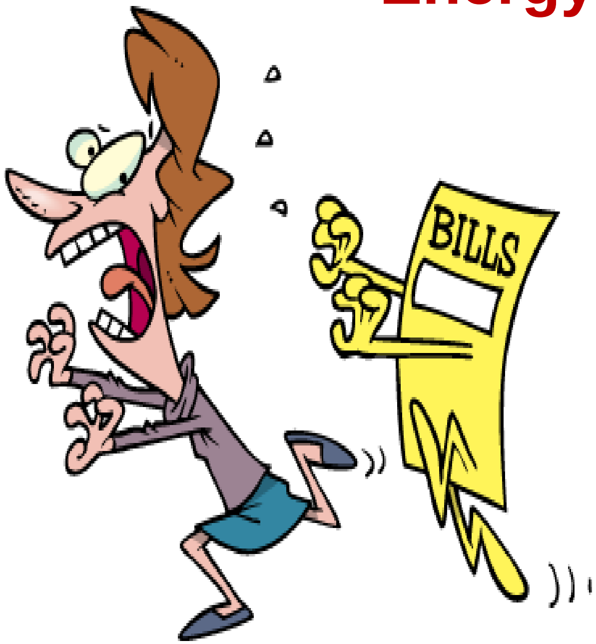
Energy Consumption



Data centers consumption will reach
300 billion kWh
in U.S. and
1012.02 billion kWh
worldwide by 2020

WESTERN SYDNEY UNIVERSITY

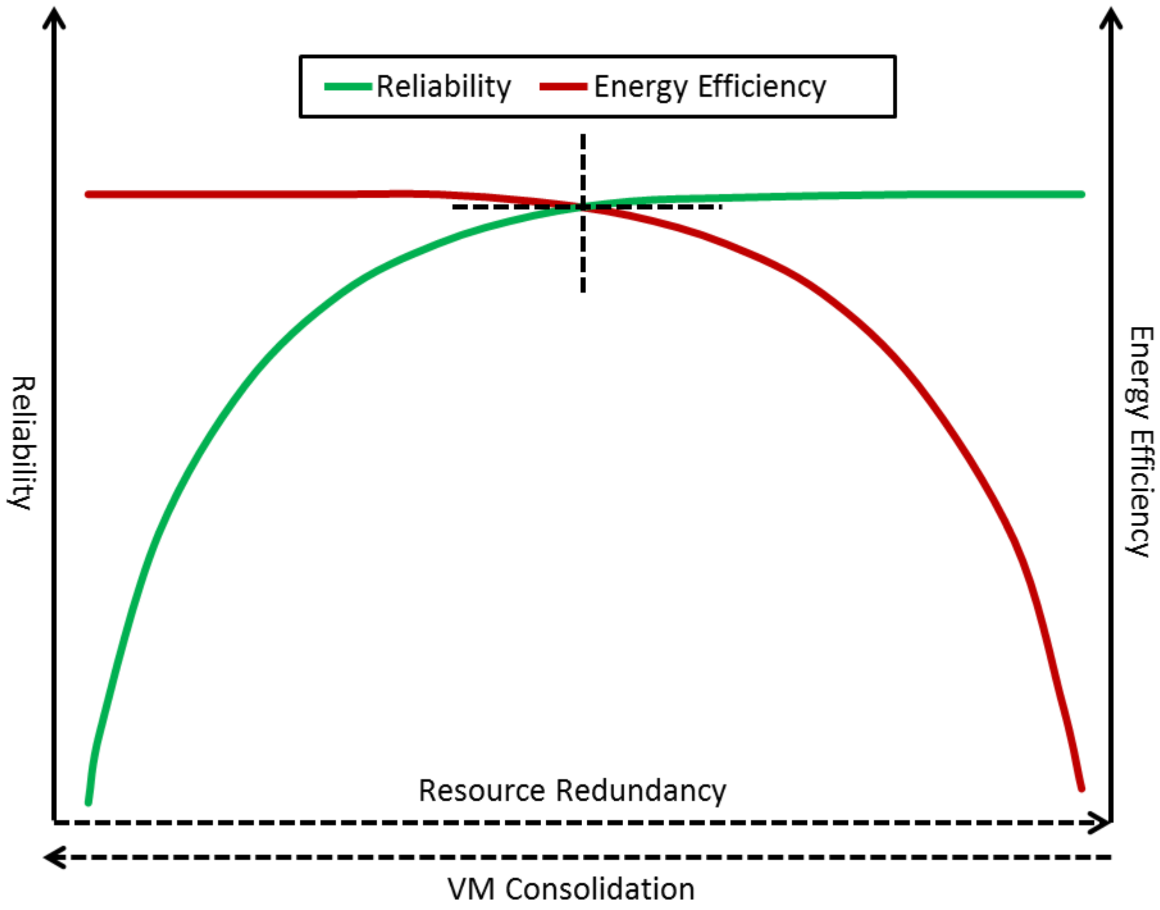
Energy Cost and Carbon Footprint



Electricity bill accounts for 20% of a US data center's Total Cost of Ownership (TCO)

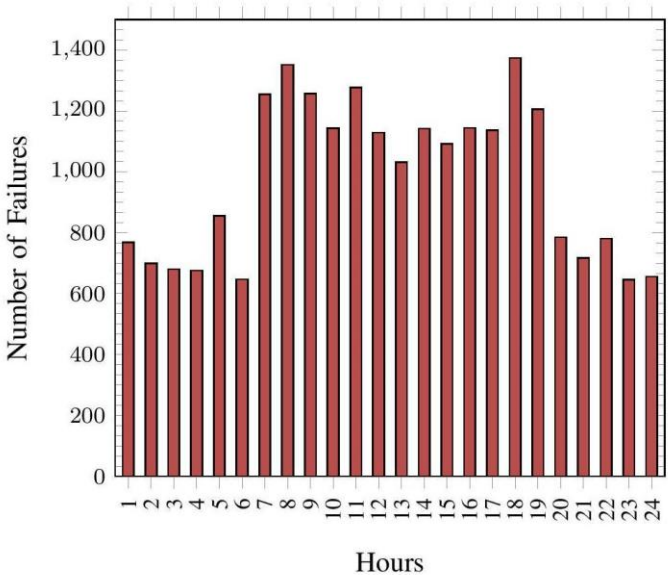
Cloud based data centers in U.S. emit 100 million metric tonne of carbon content each year and will increase to 1034 metric tonne by, 2020.

Reliability and Energy-Efficiency Trade-off



Reliability Model

WESTERN SYDNEY UNIVERSITY



- System utilization/activity and occurrence of failures are correlated.
- Linear hazard rate/failure rate directly proportional to the utilization following Poisson distribution is

$$\lambda_{ij} = \lambda_{max_j} u_i^{\beta}$$

- λ_{max_j} : Hazard rate at maximum utilization, u_{max} of a node j
- $MTBF_{max_j}$: MTBF at maximum utilization

$$\lambda_{max_j} = \frac{1}{MTBF_{max_j}}$$

WESTERN SYDNEY UNIVERSITY

Reliability Model

- Probability *(Reliability)* with which vm_i running on node n_j with utilization u_j with hazard rate λ_{ij} will finish the execution of a task t_i of length l_i is

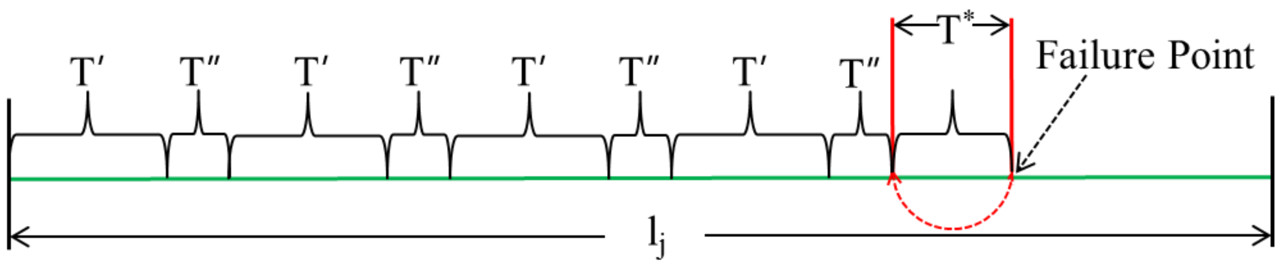
$$R_{vm_{ij}} = e^{-(\lambda_{ij})l_i}$$

- Probability with which a node n_j will finish the execution of all the m running VMs

$$R_j = \prod_{i=1}^{m} R_{vm_{ij}}$$

WESTERN SYDNEY UNIVERSITY

Finishing Time with Checkpointing



- *T'* : Checkpoint Interval
- *T"* : Checkpoint overhead i.e. time taken to save a checkpoint

$$T' = \sqrt{2 * T'' * MTBF_j}$$

- *T** : Duration of a lost part of a task that needs to be re-executed
- $T^{\#}$: Part of the task executed before the occurrence of failure
- N'_{ij} : Number of Checkpoints before a failure on a node n_j for task t_i

WESTERN SYDNEY UNIVERSITY

Finishing Time with Checkpointing

- N'_{ij} : Number of Checkpoints before a failure on a node n_j for task t_i

$$N'_{ij} = \left\lfloor \frac{T^{\#}_{ij}}{T'_j} \right\rfloor$$

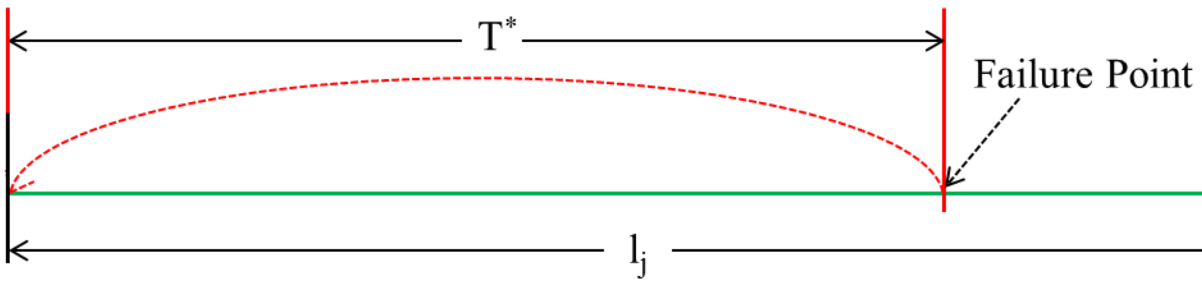
- Length of the Lost part, T^* will be calculated as

$$T^*_{ij} = \left(\frac{T^{\#}_{ij}}{T'_j} - N'_{ij} \right) * T'_j$$

- Finishing Time of a task after the occurrence of n failures under checkpointing scenario will be calculated as the sum of N'_{ij}, T^*_{ij} and time to return (TTR).

$$T^{\$}_{ij} = \begin{cases} l_i + \sum_{k=0}^{n} T^*_{(ij)_k} + \left(T'' * \sum_{q=0}^{m} N'_{(ij)_q} \right) + \sum_{k=0}^{n} TTR_{(ij)_k} \ , & k, q > 0 \\ l_i \ , & Otherwise \end{cases}$$

Finishing Time without Checkpointing



- Finishing Time of a task after the occurrence of *n* failures under without checkpointing scenario will be calculated as the sum of T^{*}_{ij} and time to return (TTR).

$$T^{\$}_{ij} = \begin{cases} l_i + \sum_{k=0}^{n} T^{*}_{(ij)_k} + \sum_{k=0}^{n} TTR_{(ij)_k} \ , & k > 0 \\ l_i \ , & Otherwise \end{cases}$$

WESTERN SYDNEY UNIVERSITY

Energy Model

- The proposed power model is a CPU utilization based model while operating at the maximum frequency.

- P_{max_j} , P_{min_j} is the maximum and minimum power consumption by a node n_j, respectively. $frac_j$ is the fraction of P_{max_j} , P_{min_j} .

- The power consumption at utilization u_j is

$$P_j(u_i) = \left(frac_j * P_{max_j}\right) + \left(\left(1 - frac_j\right) * P_{max_j} * u_i\right)$$

WESTERN SYDNEY UNIVERSITY

Energy Model

- Energy is the amount of power consumed per unit time.

- Energy consumption by a vm_i executing running on a node n_j while executing a task of length l_i in the presence of failures is given as

$$E_{vm_{ij}} = \left(P_j(u_i) * l_i\right) + E_{waste_{ij}}$$

- $E_{waste_{ij}}$ is the energy wastage because of the failure overheads

WESTERN SYDNEY UNIVERSITY

Energy Wastage with Checkpointing

- $E_{checkpoint}$: Energy consumption while saving checkpoints. Power consumption while saving a checkpoint is $1.15 * P_{min}$.
- $E_{re-execute}$: Energy Consumption while re-executing the lost part of a task because of failures.

$$E_{waste_{ij}} = E_{checkpoint_{ij}} + E_{re-execute_{ij}}$$

$$E_{checkpoint_{ij}} = \begin{cases} 1.15 * P_{min_j} * \left(T'' * \sum_{q=0}^{m} N'_{(ij)_q} \right), & q > 0 \\ 0, & otherwise \end{cases}$$

$$E_{re-execute_{ij}} = \begin{cases} P_j(u_i) * \sum_{k=0}^{n} T^*_{(ij)_k}, & k > 0 \\ 0, & otherwise \end{cases}$$

WESTERN SYDNEY UNIVERSITY

Energy Wastage with Checkpointing

- $E_{checkpoint}$: Energy consumption while saving checkpoints. Power consumption while saving a checkpoint is $1.15 * P_{min}$.
- $E_{re-execute}$: Energy Consumption while re-executing the lost part of a task because of failures.

$$E_{waste_{ij}} = E_{checkpoint_{ij}} + E_{re-execute_{ij}}$$

$$E_{checkpoint_{ij}} = \begin{cases} 1.15 * P_{min_j} * \left(T'' * \sum_{q=0}^{m} N'_{(ij)_q} \right), & q > 0 \\ 0, & otherwise \end{cases}$$

Energy wastage without checkpointing

$$E_{re-execute_{ij}} = \begin{cases} P_j(u_i) * \sum_{k=0}^{n} T^*_{(ij)_k}, & k > 0 \\ 0, & otherwise \end{cases}$$

Resource Provisioning and VM Allocation

Four resource provisioning and VM allocation line algorithms have been proposed.

- Reliability Aware Best Fit Decreasing (RABFD)
- Energy Aware Best Fit Decreasing (EABFD)
- Reliability-Energy Aware Best Fit Decreasing (REABFD)

As a baseline policy Opportunistic Load Balancing (OLB) or Random policy has been used.

WESTERN SYDNEY UNIVERSITY

Reliability Aware Best Fit Decreasing (RABFD)

- All VMs will be sorted in decreasing order according to their utilization.
- All physical resources will be sorted in increasing order according to their current hazard rate corresponding to the current utilization.
- VM with highest utilization level will get allocated to resource with minimum current hazard rate.

Reliability Aware Best Fit Decreasing (RABFD)

Function RELIABILITYAWARE(R)

1. for all j $\in$ R do
2. $\lambda_j \leftarrow r_j$.calculateCurrentHazardRate()
3. end for
4. for all j $\in$ R do
5. $R_{sorted} \leftarrow \lambda_j$.sortHazard-rateIncreasing()
6. endfor
7. return R_{sorted}

WESTERN SYDNEY UNIVERSITY

Energy Aware Best Fit Decreasing (EABFD)

- All VMs will be sorted in decreasing order according to their utilization.
- All physical resources will be sorted in increasing order according to their current power consumption corresponding to the current utilization.
- VM with highest utilization level will get allocated to the resource with minimum current power consumption.

Energy Aware Best Fit Decreasing (EABFD)

Function ENERGYAWARE(R)

for all j ϵ R do

  $P_j \leftarrow r_j$.calculateCurrentPowerConsumption()

end for

for all j ϵ R do

    $R_{sorted} \leftarrow P_j$.sortPowerIncreasing()

endfor

return R_{sorted}

WESTERN SYDNEY UNIVERSITY

Reliability and Energy Aware Best Fit Decreasing (REABFD)

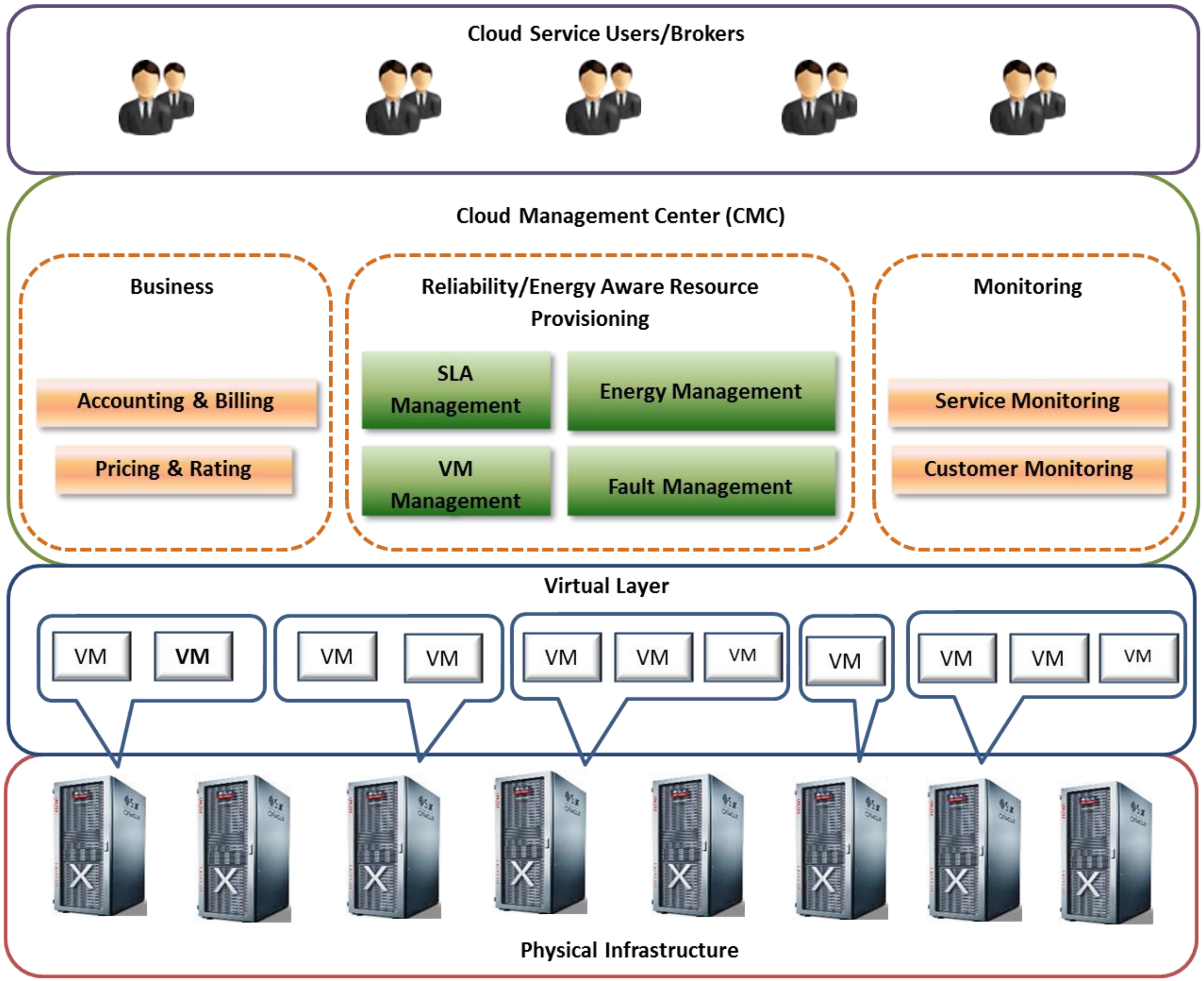
- The ratio of MTBF and power consumption has been used to rank each resource.
- All physical resources will be sorted in decreasing order according to the ratio.
- VM with highest utilization level will get allocated to the resource with highest ratio.

Reliability and Energy Aware Best Fit Decreasing (REABFD)

Function RELIABILITYANDENERGYAWARE(R)

1. for all j ϵ R do
2. $MTBF_j \leftarrow r_j$.calculateCurrentMTBF()
3. $P_j \leftarrow r_j$.calculateCurrentPowerConsumption()
4. $\Psi_j \leftarrow (MTBF_j)/(P_j)$
5. end for
6. for all j ϵ R do
7. $R_{sorted} \leftarrow \Psi_j$.sortMTBFPowerRatioIncreasing()
8. endfor
9. return R_{sorted}

System Architecture



WESTERN SYDNEY UNIVERSITY

Workload Parameters

To generate workload, Bag of Task (BoT) applications have been considered.

SNo	Parameter	Distribution	Values
1.	Inter-Arrival Time	Weibull	Scale = 4.25, Shape = 7.86
2.	Number of Tasks per Bag of Task	Weibull	Scale = 1.76, Shape = 2.11
3.	Average runtime per Task	Normal	Mean = 2.73, SD = 6.1

Failure Generation Parameters

- Real Failure Traces have been used to add Failures in simulated cloud computing systems.
- Failure information has been gathered from Failure Trace Archive (FTA)
- FTA is a public repository that has failure traces of different architectures gathered from 26 different sites.
- In this work, LANL traces gathered from Los Alamos National Laboratory between 1996-2005 have been used.

WESTERN SYDNEY UNIVERSITY

Physical Node Parameters

- To gather power profiles of the physical machines, spec2008 benchmark has been used.
- Node type has been chosen on the basis of the node information provided in the failure traces.

SNo	Node Type	Cores	Memory (GB)
1.	Intel Platform SE7520AF2 Server Board	2	4
2.	HP ProLiant DL380 G5	4	16
3.	HP ProLiant DL758 G5	32	32
4.	HP ProLiant DL560 Gen9	128	128
5.	Dell PowerEdge R830	256	256

Average Reliability

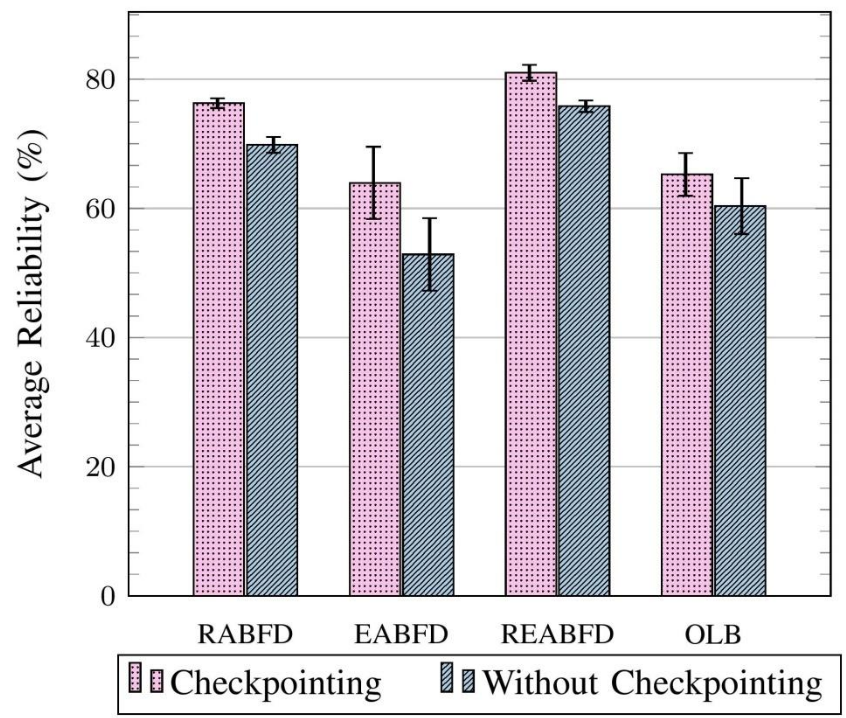
- The reliability with which application has been executed on provisioned resources

REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	5%	6%
OLB	16%	15%
EABFD	17%	23%

Checkpointing **vs** Without Checkpointing

- Policies using checkpointing gives better reliability by 5% to 9% than without checkpointing.



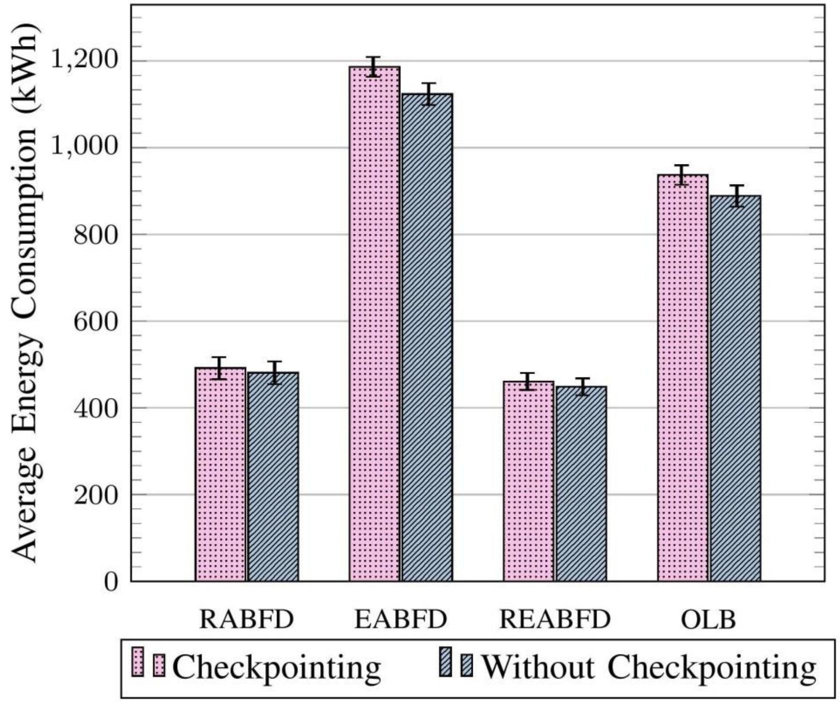
Average Energy Consumption

- Energy consumption incurred by the provisioned resources.

REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	7%	7%
OLB	50%	15%
EABFD	61%	50%

Checkpointing **vs** Without Checkpointing



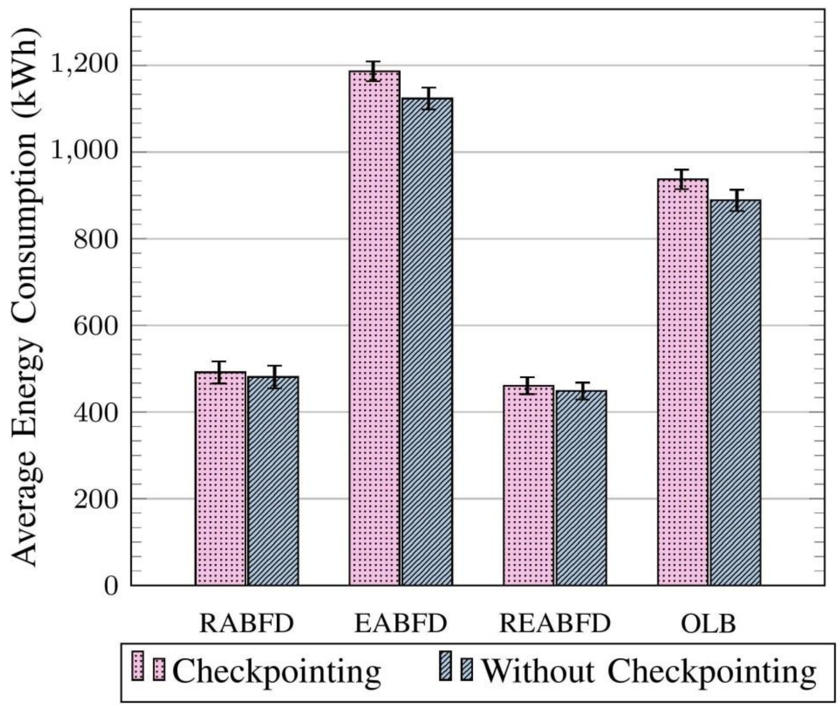
- Policies using checkpointing consumes more energy by 2% to 5% than without checkpointing.

Average Energy Consumption

- Energy consumption incurred by the provisioned resources.

REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	7%	7%
OLB	50%	15%
EABFD	61%	50%



In-fact, better not to use any policy and keeps allocation random, if reliability will not be considered

WESTERN SYDNEY UNIVERSITY

Average Energy Wastage

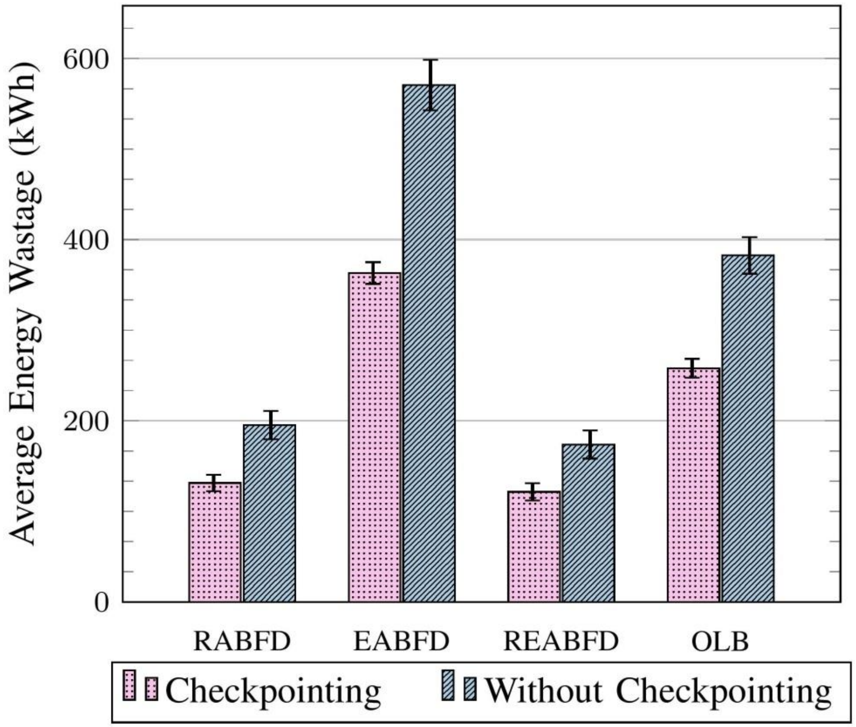
- The amount of energy wasted because of the failure overheads.

REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	8%	11%
OLB	53%	54%
EABFD	67%	70%

Checkpointing **vs** Without Checkpointing

- Wastage has been observed more by 36% in the absence of checkpointing because of large re-execution overheads



WESTERN SYDNEY UNIVERSITY

Average Turnaround Time

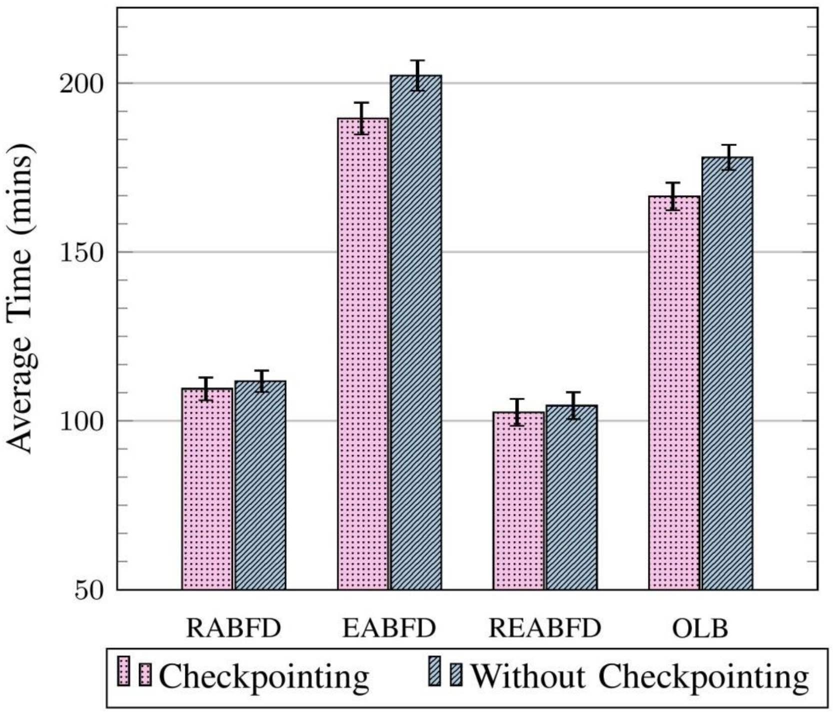
- It is the time taken by each task of BoT application to finish.

REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	7%	7%
OLB	39%	39%
EABFD	46%	46%

Checkpointing **vs** Without Checkpointing

- Better turnaround time has been achieved by 7% while using checkpointing.



WESTERN SYDNEY UNIVERSITY

Deadline-Turnaround Time Fraction

- It is the margin by which the turnaround time has been exceeded from the deadline.

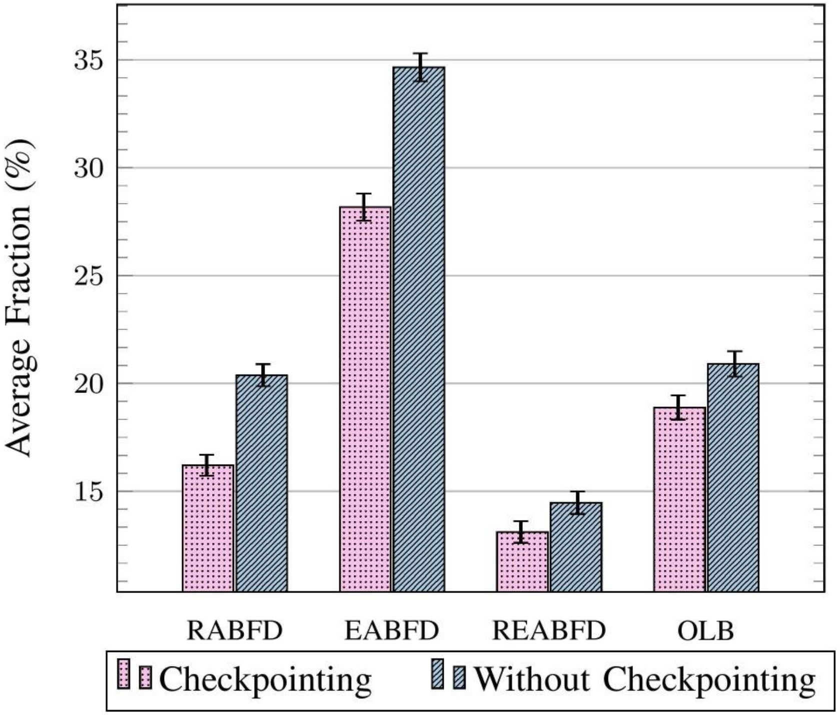
REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	3%	6%
OLB	6%	7%
EABFD	15%	20%

Checkpointing **vs** Without Checkpointing

- For scenarios without checkpointing, the makespan has been exceeded more by 7% in comparison to checkpointing.

- Re-execution has been found higher by 36% for without checkpointing scenario.



WESTERN SYDNEY UNIVERSITY

Average Benefit Function

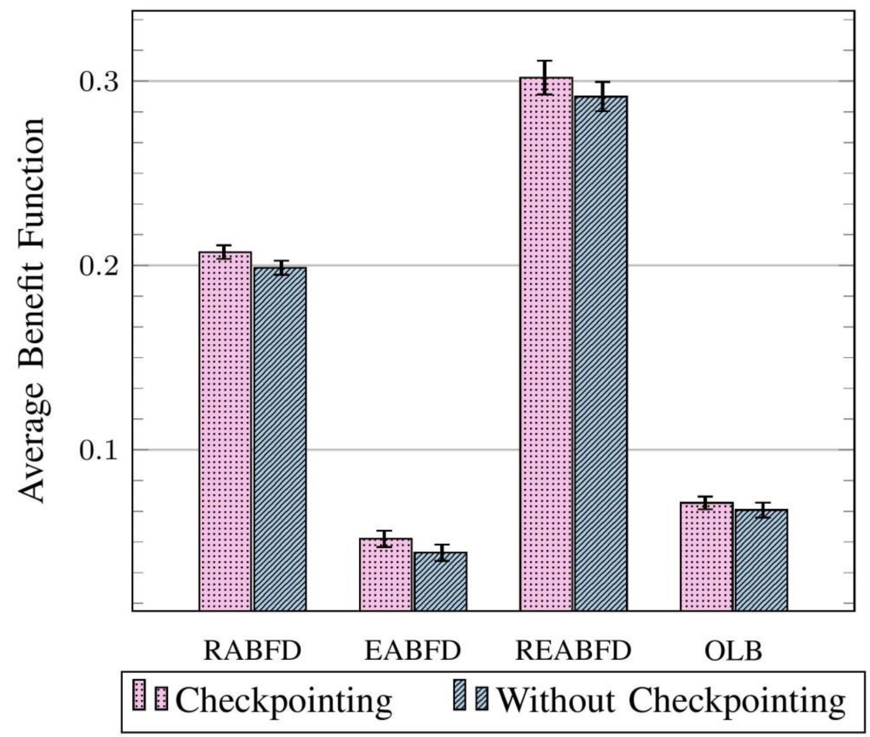
- It is ratio of reliability and energy consumption of the system.

REABFD vs other policies

Policy	Checkpointing	Without Checkpointing
RABFD	29%	34%
OLB	76%	85%
EABFD	82%	78%

Checkpointing **vs** Without Checkpointing

- Scenarios using checkpointing gives better benefit function upto 14% than without checkpointing.



Conclusion and Future Work

- While giving emphasis only to the energy optimization without considering reliability factor, results are contrary to the expectation.
- More energy consumption has been experienced due to the energy losses incurred because of failure overheads.
- Reliability-Energy Aware Best Fit Decreasing (REABFD) policy outperforms all the other policies.
- It has been revealed that by considering both energy and reliability factors together, both factors can be improved better than being regulated individually.
- In future, machine learning methods will be used to predict the occurrence of failures.
- By using failure prediction results, VM migration and consolidation mechanism will be adopted to further optimized the fault tolerance and energy consumption.

Thank You