Exact Analysis of Energy-Aware Queueing Systems with Setup Times

Vincent J. Maccio, Douglas G. Down

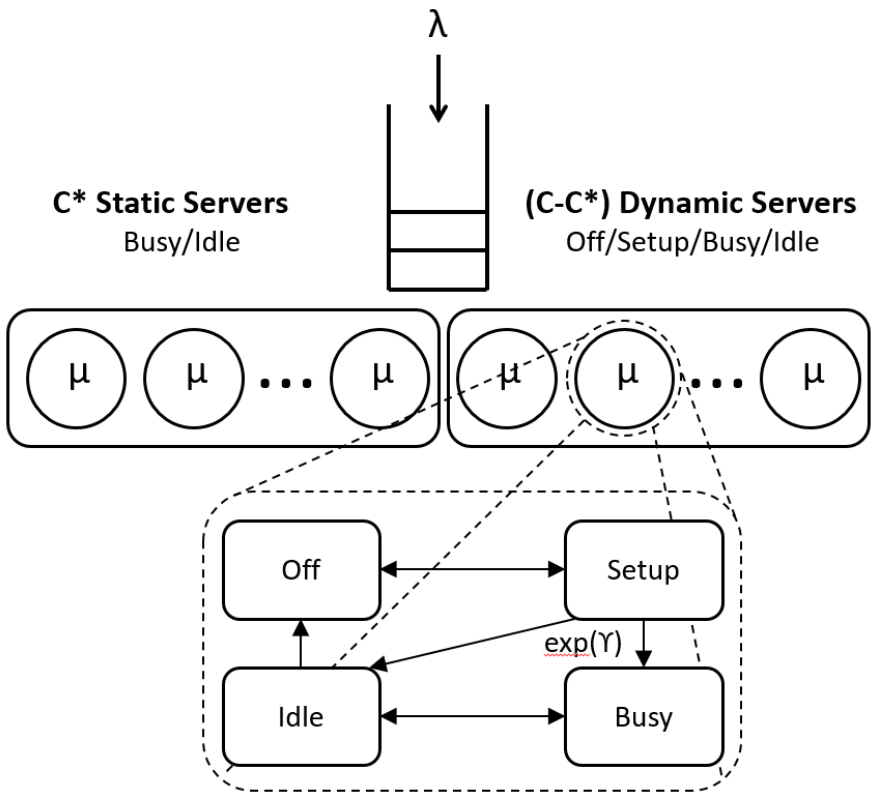
Department of Computing and Software
McMaster University
Hamilton, Ontario, Canada

Motivation

- Server farms use a lot of energy
- Systems are designed to handle peak loads
  - Some servers spend a lot of their time idle
- When a server idles it *still* uses a lot of energy
- Why not just turn them off?
  - Overhead, system performance suffers, could use more energy in the long run, etc.

Model

- C homogeneous servers
- Jobs arrive following a Possion process with rate λ
- Processing times are exponentially distributed with rate μ
- Setup times are exponentially distributed with rate γ
- C^* servers will always remain on



Policies

- Determine the number of static servers, when does each dynamic server turn on/off
- These values may be be given explicitly:
  - With $C = 2$ let $C^* = 1$, the second server will turn on if there are 3 or more jobs in the system and turn off if there is 1 or less
- Policy criteria may also be given:
  - The ith server turns on if there are i or more jobs waiting in queue
  - Servers turn off if there are two or move servers idle
  - When there are jobs in queue wait from some time t then begin a server setup

Metrics and Costs

- The expected response time $\mathbb{E}[R]$
- The expected energy cost $\mathbb{E}[E]$
  - A weighted sum of the expected number of servers multiplied with a corresponding cost: $\mathbb{E}[E] = r_{idle}\mathbb{E}[N_{idle}] + r_{setup}\mathbb{E}[N_{setup}]$
  - *Busy* and *off* servers are excluded
- Example cost functions: $\mathbb{E}[E]\mathbb{E}[R]$ or $\mathbb{E}[R] + \beta\mathbb{E}[E]$

Problem and Approach

- The set of potential policies to study can be intimidating
- Derive structural properties of the optimal policy which allow us to reduce this set
- From this reduced set, select specific policies to analyse further
- Basic properties:
  - The optimal policy is a threshold policy
  - Optimal decisions are always made the moment an event occurs

Structural Property: Non-Idle Server Q/A

Q Given a cost function increasing in $\mathbb{E}[R]$ and $\mathbb{E}[E]$, if there are i jobs in the system, does it ever make sense to turn the ith server off?

- Keeping a server on now could cause more to turn on in the future?

Structural Property: Non-Idle Server Q/A

Q Given a cost function increasing in $\mathbb{E}[R]$ and $\mathbb{E}[E]$, if there are i jobs in the system, does it ever make sense to turn the ith server off?

- Keeping a server on now could cause more to turn on in the future?

A No, it is always suboptimal to turn a server off if there is a job to be processed, regardless of the weight on the energy cost

Structural Property: Bulk Setup Q/A

- Consider a policy with the following "turn on" behaviour
  - Whenever the system determines to turn on another server on, it begins the setup process of all available dynamic servers

Q A much less complex system to analyse, but is it reasonable?

Structural Property: Bulk Setup Q/A

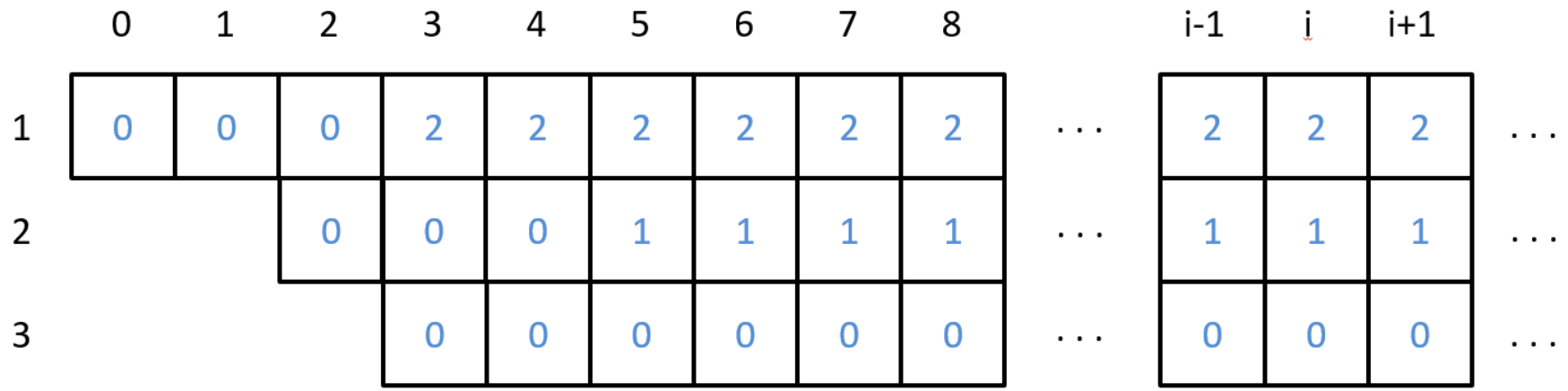
- Consider a policy with the following "turn on" behaviour
  - Whenever the system determines to turn on another server on, it begins the setup process of all available dynamic servers

Q A much less complex system to analyse, but is it reasonable?

A When the cost function is linear in $\mathbb{E}[R]$ and $\mathbb{E}[E]$, the optimal policy uses the bulk setup scheme

The Bulk Setup Policy

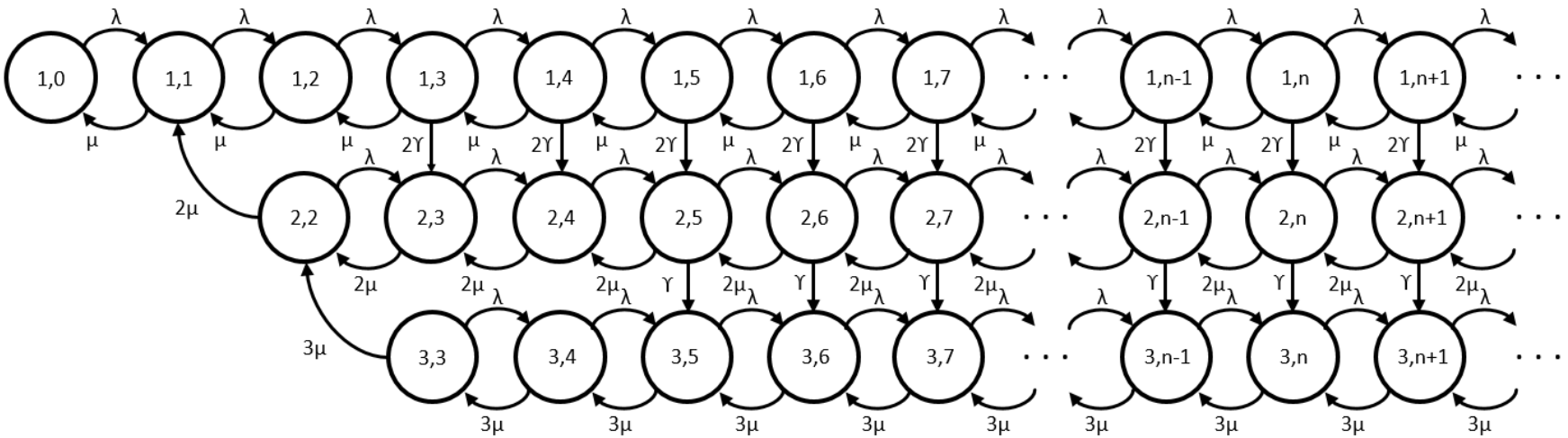
- C^* servers always remain on
- Dynamic servers turn off when they idle
- If i servers are on, all servers will be moved to setup if there are $ik + C^*$ or more jobs in the system



Graphical representation with $C = 3$, $C^* = 1$, and $k = 2$

The Bulk Setup Policy

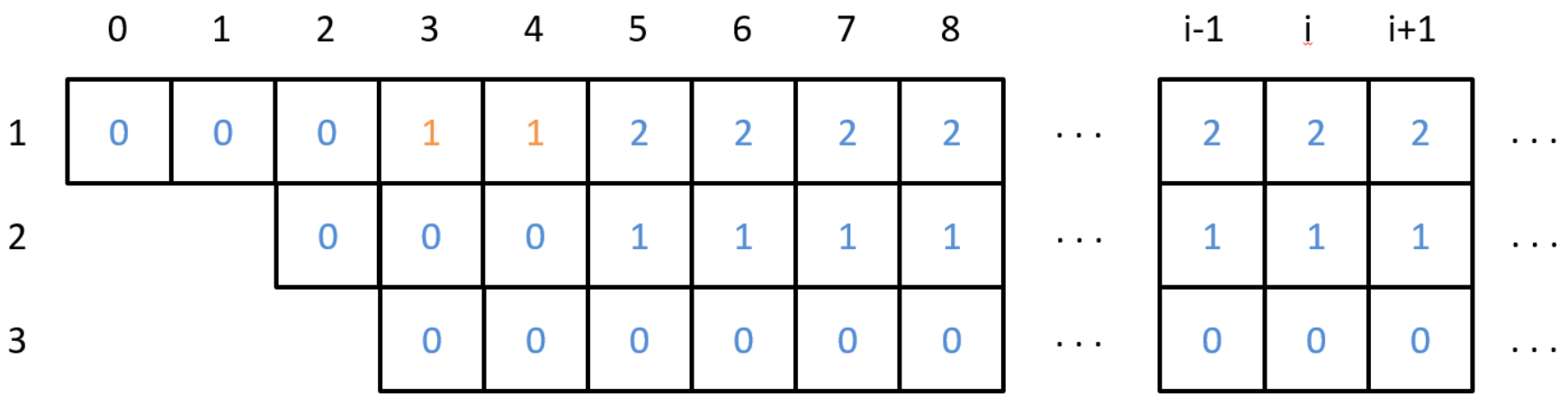
- C^* servers always remain on
- Dynamic servers turn off when they idle
- If i servers are on, all servers will be moved to setup if there are $ik + C^*$ or more jobs in the system



Graphical representation with $C = 3$, $C^* = 1$, and $k = 2$

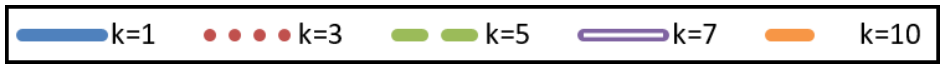
The Staggered Threshold Policy

- C^* servers always remain on
- Dynamic servers turn off when they idle
- The ith dynamic server with begin its setup when there are $ik + C^*$ or move jobs in the system

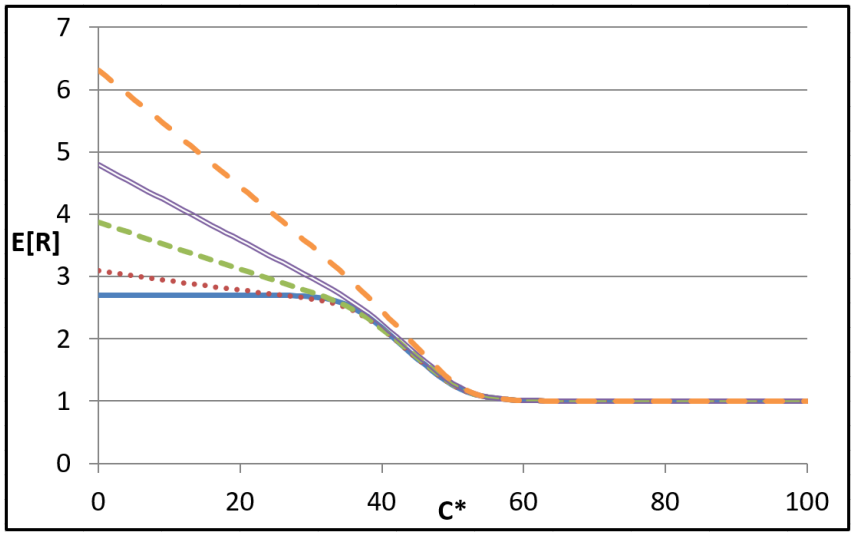


Graphical representation with $C = 3$, $C^* = 1$, and $k = 2$

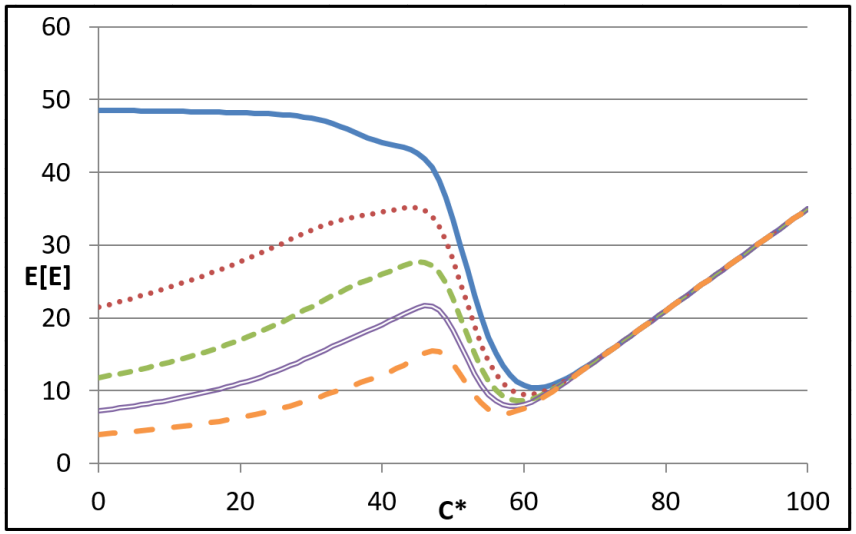
Bulk Setup: Numerical Results



$C = 100,\ \lambda = 50,\ \mu = 1,\ \gamma = 0.01$

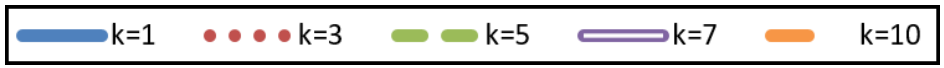


Expected Response Time vs C^*

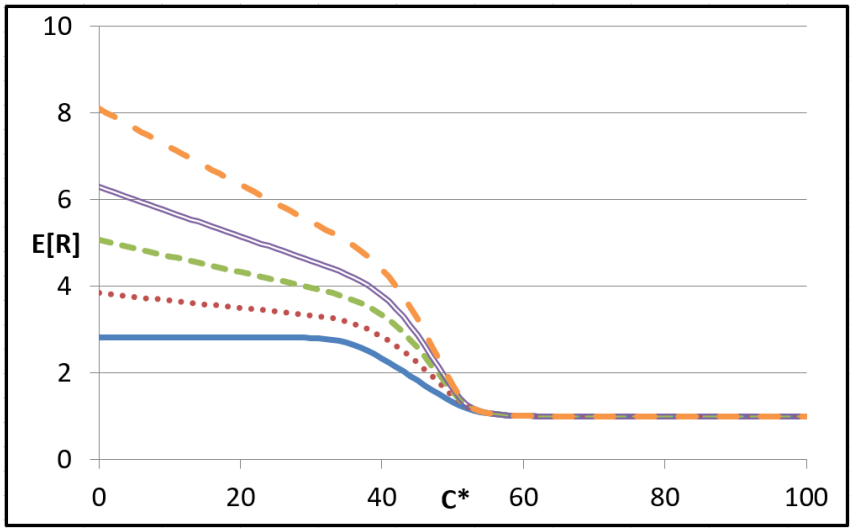


Expected Energy Cost vs C^*

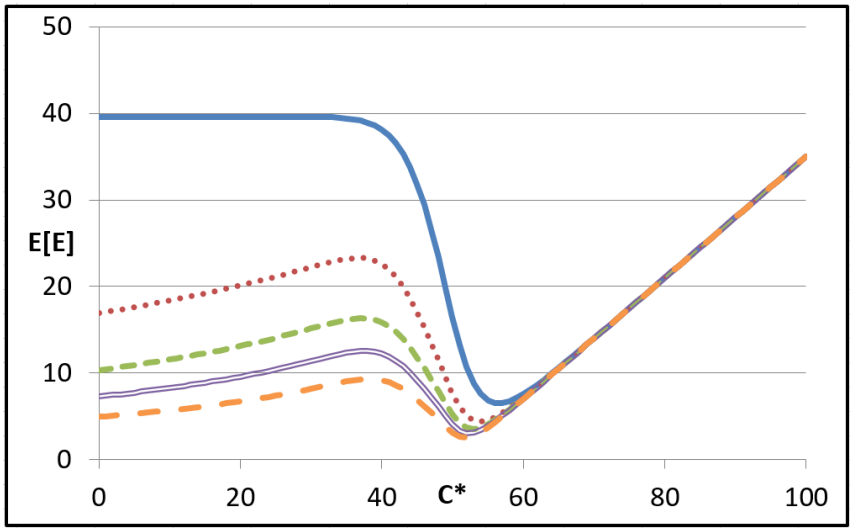
Staggered Threshold: Numerical Results



$C = 100,\ \lambda = 50,\ \mu = 1,\ \gamma = 0.01$



Expected Response Time vs C^*



Expected Energy Cost vs C^*

Observations

- For the choice of C^* there is a *sweet spot* around $\rho + \sqrt{\rho}$
- The larger the choice of k the lower the energy cost
- Around $C^* = \rho + \sqrt{\rho}$, $\mathbb{E}[R]$ is insensitive to k, and close to the lower bound of $1/\mu$

Observations

- For the choice of C^* there is a *sweet spot* around $\rho + \sqrt{\rho}$
- The larger the choice of k the lower the energy cost
- Around $C^* = \rho + \sqrt{\rho}$, $\mathbb{E}[R]$ is insensitive to k, and close to the lower bound of $1/\mu$
- A near optimal solution for these systems is the degenerate solution of an $M/M/C^*$ queue where $C^* = \rho + \sqrt{\rho}$

Conclusion

- Derived structural properties to shrink our set of potential policies
- We then chose two policies to study further
- Performed an exact analysis on the underlying CTMCs
- Determined that the degenerate solution of an $M/M/C^*$, is a reasonable solution

The End

Questions?