On Optimal Control for Energy-Aware Queueing Systems

Vincent J. Maccio, Douglas G. Down

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

Topic and Motivation

Modelling and analysis of energy provisioning in data-centers.

- 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.

The Model

- N homogeneous servers
    - Each server can be switched between two states (on/off)
    - A server takes time to turn on
    - Turn on times are exponentially distributed with rate γ
    - Servers can be turned off instantaneously
    - Setups may be allowed to be interrupted
- Jobs arrive to a central queue and are served FIFO
    - Jobs arrive according to a Poisson process with rate λ
- Jobs sizes follow an exponential distribution with rate μ
- Each server consumes energy at a different rate when it's *BUSY*, *IDLE* or in *SETUP*

An energy-aware system S can be seen as a 4-tuple $(N, \lambda, \mu, \gamma)$

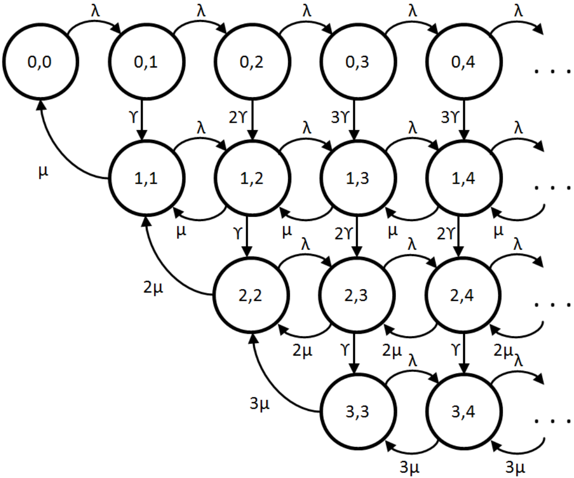
Control and Cost

- System metrics of interest: $\mathbb{E}[R], \mathbb{E}[E]$, and maybe $\mathbb{E}[\mathcal{S}]$
- Cost functions are increasing and continuous in these metrics
  - Example cost functions: $\mathbb{E}[R]\mathbb{E}[E]$, $\mathbb{E}[R] + \beta\mathbb{E}[E]$
- Energy rates depend on if the server is *busy*, *idle*, or in *setup*
- It's assumed $r_{idle} < r_{busy}$ and $r_{idle} < r_{setup}$
- When to turn a server on, when to turn a server off
- Choosing the correct state space, events occur whenever the system changes its state
  - (# of jobs in the system, # of servers on, # of servers in setup)
  - (# of jobs in the system, # of servers on)

Related Research

- Other researchers have analysed similar systems under specific policies
- Staggered setup [1, 2]
  - Turn on: whenever there are more jobs in the queue than there are servers in setup
  - Turn off: whenever a server becomes idle
- (e, d) setup [3]
  - Turn off: turn off e servers when d of them are idle, or if it just turned on with no jobs to serve
  - Turn on: immediately turn an off server on

Staggered Setup Example



Simplified Two Server Example

Turn on criteria

- k_1^{On}: the # of jobs in the system before the first server turns on
- k_2^{On}: the # of jobs in the system before the second server turns on

Turn off criteria

- k_1^{Off}: the # of jobs left in the system before the first server turns off
- k_2^{Off}: the # of jobs left in the system before the second server turns off

We know $k_1^{On} \leq k_2^{On}$ and $k_1^{Off} \leq k_2^{Off}$, but that is not enough.

For example, is $k_1^{On} \leq k_1^{Off}$ or is $k_1^{On} > k_1^{Off}$?

My Contribution

Discover and prove structural properties concerning the optimal policy

- Gain confidence in already analysed policies
- Eliminate potential policies on the basis they will always be suboptimal
- Intelligently construct policies which have the shown properties, but are still tractable to analyse

Markov Decision Process

- An energy-aware system can be modelled as a Markov decision process
    - Decision epochs are events/state transitions, and the action set is the number of servers to turn on/off
- Can always run numerical examples (truncation of state space)
- Can prove/derive structural properties of the optimal policy
    - These properties can be obvious, intuitive, or counter intuitive
- These structural properties allow for a tractable analysis of the corresponding CTMC

Basic Properties

- Server setups and turn offs follow a threshold policy
    - If it's optimal to turn on a server when there n jobs in the system, than it's also optimal to turn it on when there are $n+1$ jobs in the system
- Optimal decisions are only made the moment an event occurs
    - It never makes sense to wait a certain amount of time before turning a server off or on

Structural Property: Anticipative Server Q

Consider an energy-aware system with the following properties

- It is a two server system, i.e. $N = 2$
- There are considerable turn on times, i.e. $\gamma << \lambda$
- The load on the system is significant enough to need the second server to be stable, i.e. $\mu < \lambda < 2\mu$
- If the second server is on, it is optimal to turn it off when there are 5 jobs remaining in the system

If the second server is currently off, and there are 3 jobs in the system what is a reasonable thing to do?

Structural Property: Anticipative Server A

Keep it turned off.

- This eliminates a large set of decision variable values
- From the two server example seen previously, we would not need to consider the cases where $k_2^{On} \leq k_2^{Off}$

Structural Property: Non-Idle Server Q

Consider all energy-aware systems under any well formed cost function built on $\mathbb{E}[R]$ and $\mathbb{E}[E]$. If there are n jobs in the system, does it ever make sense to turn the n^{th} server off?

- The energy cost of processing the job is unfavourable?
- Keeping a server on now could cause more to turn on in the future?

Structural Property: Non-Idle Server 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.

- Therefore for all $n > 0$ the case of $k_n^{Off} > n$ can be eliminated. This greatly decreases the complexity of the corresponding CTMC.

The Bulk Setup Policy

Consider a policy with the following “turn on” behaviour

- When there are n servers currently on, the system has a single threshold value k_n^{On}
- When this threshold is reached it begins turning on all remaining servers
- This reduces the number of decision variables from $N(N+1)$ to $2N$

A much less complex system to analyse. But is it reasonable?

Structural Property: Bulk Setup

When setup times are interruptible and the cost function is linear in $\mathbb{E}[R]$ and $\mathbb{E}[E]$, the optimal policy is a bulk setup policy.

Conclusion

- Multiserver energy-aware systems are a topic of interest among researchers, and have practical application to industry
- These structural properties have significantly reduced the set of potential optimal policies
- Allows for tractable policies to be intelligently chosen and analysed

[1] A. Gandhi, S. Doroudi, M. Harchol-Balter and A. Scheller-Wolf. "Exact Analysis of the M/M/k/setup Class of Markov Chains via Recursive Renewal Reward." Proceedings of ACM SIGMETRICS 2013

[2] T. Phung-Duc. "Exact Solutions for M/M/c/Setup Queues" arXiv:1406.3084

[3] X. Xu, and N. Tian. "The M/M/c Queue with (e, d) Setup Time." *Journal of Systems Science and Complexity 21* 2008, pp. 446-455.

The End

Questions?