

A Hybrid Reinforcement Learning Approach to Autonomic Resource Allocation (2006)

Paper: Gerald Tesauro, Nicholas K. Jongt, Rajarshi Das and Mohamed N. Bennanit
Presentation: Nat McAleese (nm583)

What is autonomic computing?

“computing systems that can manage themselves given high-level objectives from administrators”

An IBM marketing buzzword? (probably) [0]

Totally inevitable due to system complexity?

Elastic scaling?

[0] Kephart, Jeffrey O., and David M. Chess. "The vision of autonomic computing." *Computer* 36.1 (2003): 41-50.

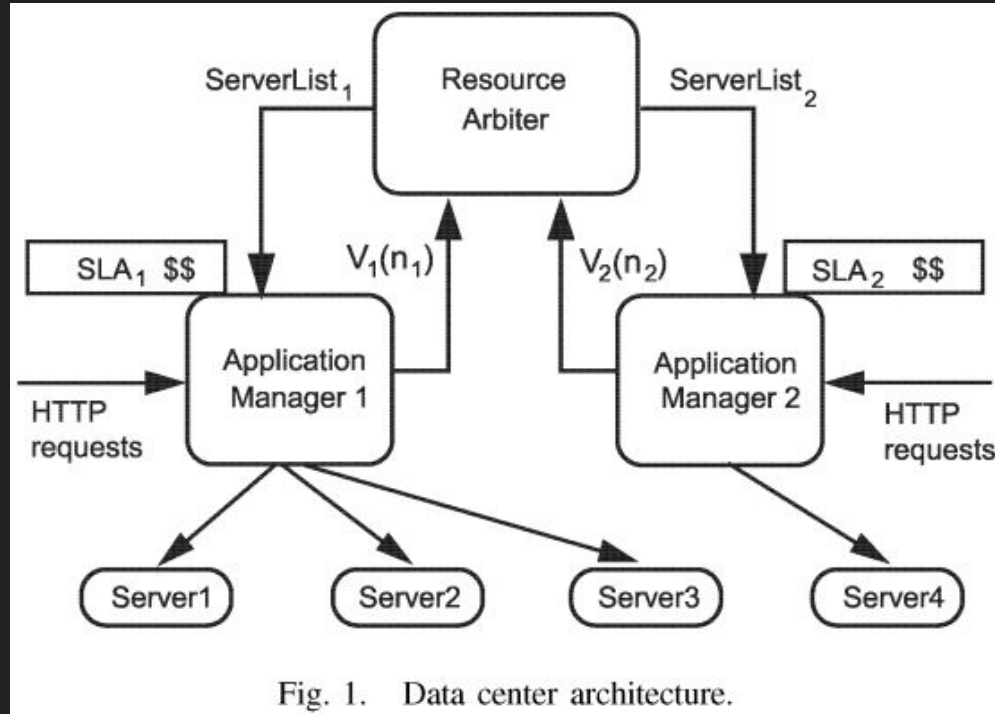
Problem

Maximise system-wide utility, which is assumed to be based on system performance relative to SLAs (Service Level Objectives)

Considers the sum of utilities from application specific utility functions.

For example web services are penalised if they fall below some latency, batch jobs improve linearly with more added resources

The scheme



Critical Detail

“Given enough training samples, RL can converge to the correct value function V^π associated with any fixed policy π , and that the new policy whose behavior greedily maximizes V^π is guaranteed to improve upon the original policy.” [0]

[0] G. Tesauro et al.: A Hybrid Reinforcement Learning Approach to Autonomic Resource Allocation, ICAC, 2006.

The scheme

Each application manager submits a utility function, $\hat{U}_i(\mathbf{R}_i)$ computed by evaluating the learnt $Q(s, a)$ for all a .

The resource arbiter then optimises
[0]

$$\mathbf{R}^* = \arg \max_{\mathbf{R}} \sum_i \hat{U}_i(\mathbf{R}_i) \text{ s.t. } \sum_i \mathbf{R}_i = \bar{\mathbf{R}}, \quad (1)$$

But it involves solving an NP-hard mixed integer programming model at the resource arbiter.

Could it help with adding/removing processes? Is there an (unintentionally) adversarial component? Need more data.

The network

3 inputs - current demand λ , n_{t-1} , n_t

12 sigmoid hidden units

1 output.

It's tiny.

[0] G. Tesauro et al.: A Hybrid Reinforcement Learning Approach to Autonomic Resource Allocation, ICAC, 2006.

Results

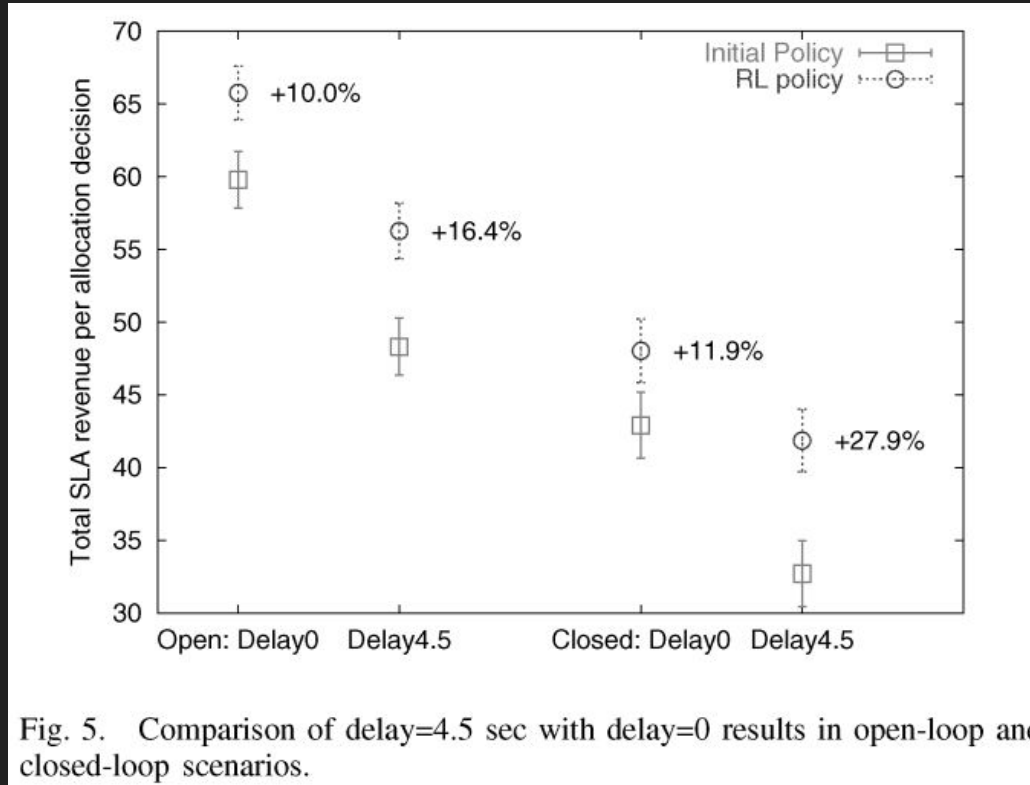


Fig. 5. Comparison of delay=4.5 sec with delay=0 results in open-loop and closed-loop scenarios.

Future work?

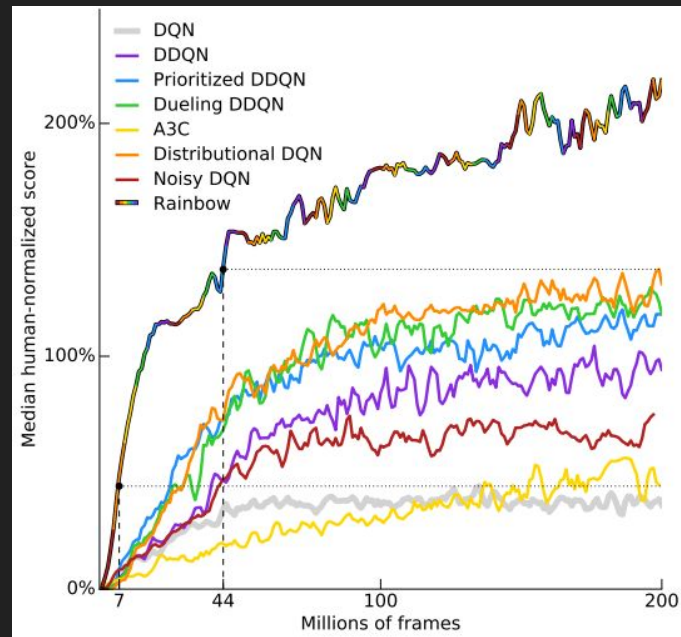
These approaches could be improved by *loads* of recent developments. I'd start with applying some (or all) of the recently discovered optimisations for reinforcement learning [0]:

And recent work to improve the raw performance
Of fully connected nets [1]

The real problem is (in my opinion) is
lack of a standard benchmark.

[0] Hessel, Matteo, et al.
"Rainbow: Combining Improvements in Deep Reinforcement Learning."
arXiv preprint arXiv:1710.02298 (2017).

[1] Klambauer, Günter, et al. "Self-Normalizing Neural Networks."
arXiv preprint arXiv:1706.02515 (2017).



Criticism

Does the system design make any sense at all? Is it more scalable? Their previous paper does not present a coherent argument for why this scheme is useful with *learned* utility functions, instead arguing that they are intuitive to design.

Compatibility with software that's actually used - OpenStack is becoming increasingly popular, and has been the target of similar work.

Standard benchmarks

Standard benchmarks.

Criticism

From next week, the papers that lack a standard benchmarking approach:

Dutreilh, Xavier, et al. "Using reinforcement learning for autonomic resource allocation in clouds: towards a fully automated workflow." ICAS 2011, The Seventh International Conference on Autonomic and Autonomous Systems. 2011.

LaCurts, Katrina, et al. "Cicada: Introducing Predictive Guarantees for Cloud Networks." HotCloud 14 (2014): 14-19.

Roy, Nilabja, Abhishek Dubey, and Aniruddha Gokhale. "Efficient autoscaling in the cloud using predictive models for workload forecasting." Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011.

Delimitrou, Christina, and Christos Kozyrakis. "Quasar: resource-efficient and QoS-aware cluster management." ACM SIGPLAN Notices. Vol. 49. No. 4. ACM, 2014.

The CherryPick guys tried though! - used, SparkKM, SparkReg TPC-DS, TPC-H, TeraSort. Unfortunately these are just batch processing workloads. - Alipourfard, Omid, et al.

"CherryPick: Adaptively Unearthing the Best Cloud Configurations for Big Data Analytics." NSDI. 2017.