Mapping Applications onto VMs onto PMs in Cloud Computing

By Zack Meeks, CS111 Winter 2017

Cloud computing is one of the most recent tools to become available to both individuals and institutions which allow them to stay at the forefront of innovation and to save money (aka maximize profit). Cloud computing is simply the outsourcing of computer tasks and applications to machines housed in server warehouses around the globe by cloud services companies whose main purpose is to supply their clients with robust computation solutions that in many cases save the clients both money and trouble.

While individuals or institutions have their own concerns about how much money to spend on number crunching and other computing tasks, their time restraints also dictate how much they will spend on cloud services for their computing needs. To top things off, market forces of supply and demand are also at play which complicates decision-making for cloud services clients.

Similarly, cloud services companies' pricing strategies are complicated by market forces as well because they want to appear cheaper than their competitor and have a wider profit margin and deliver competitive results. Cloud services companies try to achieve the above trifecta through the optimization of resources. These resources being energy, server space, physical machine usage, and throughput. Energy is the most costly resource despite current optimizations. While cloud services offer a multitude of services, their basic business model can be seen as the two fundamental tasks of mapping client applications to virtual machines and then mapping those virtual machines (VMs) onto real physical machines (PMs) somewhere in the world. It is often assumed, as it was in the research paper that we're looking at, that the mapping of applications to VMs is considered before starting to solve the issue of VMs to PMs. The solving of these mapping issues are handled by what the cloud service literature calls "the VM configuration manager" and "the VM placement manager."

It is easy to see how not having enough resources available to meet client demand as well as the under-utilization of resources is both problematic and undesirable to cloud services providers. Similarly it is easy to see that both mapping problems should have perfect solutions that can be solved for in a finite amount of time. The tricky part, and why there is so much research going on in these two areas, is that these solutions are by no means trivial!

The problem of initial VM placement can be formulated as an integer programming problem which is NP-hard. We say "initial" because generally speaking, very few programs

entail static resource demand. Hence, VMs may change and following that the PMs will change (and we haven't even begun considering the problem of mapping VMs to PMs). Assuming that the number of VMs to be mapped are quite high and that no known polynomial solution currently exists, we see that a perfect solution is not ideal and would be incredibly impractical. Thus cloud service providers aim to provide the most effective heuristic solution that research has found and that is congruent with their current basic framework (i.e. Amazon EC2 doesn't guarantee performance, while some cloud services do at a more premium cost to the client). Naturally, a lot of research is ongoing in this field as cloud service providers have a lot to gain with with tiny-fractional improvements on current heuristics.

In this *initial placement of VMs* section of the paper "Mapping Virtual Machines onto Physical Machines in Cloud Computing," many of the solutions were a heuristic flavor of dynamic programming ("knap sack" or "bin packing"). Other optimization techniques utilized throughout the course of optimizing cloud computing (and the said paper) are "utility and reward functions", "stochastic models", "genetic algorithms", "neural networks", "support vector machines", and as previously mentioned "integer and linear programming."

Most of these optimizations come into play when dealing with dynamic mapping (and remapping ad infinitum) of the VMs onto the PMs; this then brings up the issue of scheduling the tests for remapping and the question of what triggers the remapping. We can see that analyzing resource demands has the potential to strain resource supply (it costs energy and computation time) and additionally the reallocation of resources isn't done free of charge (it costs energy, computation time, and impacts network throughput among other things). Researchers have sought in this area to come up with a good enough solution that maximizes profit without being perfect.

It is easy to think of cloud computing services as just a commodity, which it essentially is as it is unlikely that a noticeable deviation from market price will be tolerated. All the same, we see that different companies are trying to differentiate themselves by having slightly different pricing strategies and/or marketing strategies, which often then play a factor in deciding which algorithms to use (as we saw earlier in the EC2 versus performance guarantee example). All these things play into how different companies tackle their goal of achieving higher value and wider margins. Yet the diversification of products (or the semblance there of) is leading to new areas requiring more research as it is presently unknown if the diversification of products on the cloud side is leading to increased performance on the client side, and to what degree new algorithms can make as the branches of the product lines spread. It is possible that the cost

against economies of scale that diversification impacts is not worth as much as it is proposed to enhance. We know that basic reallocation scheduling is a problem that still has its share of optimization room for enhancement, and thus we see with abundant clarity that reallocation scheduling with new diversifications and pricing strategies entail that this is one of the more demanding areas requiring much more research.

Bibliography:

Pietri, Ilia, and Rizos Sakellariou. "Mapping Virtual Machines onto Physical Machines in Cloud Computing: A Survey." *ACM Computing Surveys (CSUR)*. ACM, Dec. 2016. Web. 11 Mar. 2017.