Differences between the two papers

Although both our VMR paper and the USC paper try to solve the problem of energy efficient virtual machine replication, they have different approaches. The goal of our VMR paper is to make redundant deployment of VMs in anticipation of PM failures by creating R copies of each VM and placing them in physical machines with a storage capacity constraint, where no two copies of the same VM are stored in the same PM for fault tolerance purposes, while minimizing the power consumption on the switches of the data center networks. This paper doesn’t have an algorithm or a method to choose how many copies (R-1) each VM will have. Instead, R is just a chosen value given to the minimum cost flow (MCF) algorithm. In this paper, each VMs and their replicas are assumed to be of the same size. This paper doesn’t consider resource requirements needed by the VMs. Also, Unlike the USC paper, this paper doesn’t consider Service Level Requirements (SLAs). By transforming the data center network to a flow network, VMR is equivalent to the MCF problem and therefore could be solved optimally. Power consumption is further reduced via PM consolidation. PM consolidation aims to move the VM replica copies into a smaller number of PMs while maintaining the mentioned constraints and maintaining the minimum cost flow power consumption. PM consolidation aims to turn off PMs that do not store any VMs to save power consumption. Server consolidation does not incur extra cost on top of VM replication because VM replicas are not physically placed in the data center until server consolidation is done, at which point the final placement of all VM replicas is decided.

The goal of the USC paper is to have as few servers as possible turned on, with each ON server being highly utilized. The IT infrastructure provided by the datacenter owners must meet various SLAs established with the clients that may be resource related, performance related, or quality of service related. Generating multiple copies of a VM and placing them on different servers is one of the basic ways to increase the service reliability. In this paper, the authors propose to exploit all the VM copies for servicing requests. This paper considers Memory bandwidth and CPU cycle requirements of each VM. So, unlike our VMR paper, each VM can be different from each other, with different memory BW and CPU cycle requirements. Also, in our VMR paper, VM replicas are the same as the original copy whereas in the USC paper, memory BW provided for each copy of a VM is the same, and the total CPU cycles provided for all of the VM copies is equal to those provided to the original copy. The proposed algorithm in this paper is based on the dynamic programming methods and local search methods. Unlike our VMR paper, this paper’s proposed algorithm determines the copies for each VM, where the number of copies is less than or Equal to Li(The maximum number of copies for client i). The dynamic programming method determines the number of copies for each VM and places them on servers whereas the local search method tries to minimize the energy cost by turning off under-utilized servers. For the system model, this paper considers a data center of k different server types with different CPU cycles, memory bandwidth and energy cost whereas our VMR paper considers all the servers to be of the same type. This paper focuses on the VM controller semi-static optimization. The semi-static procedure considers the active set of VMs, previous assignment solution, feedbacks generated from power, thermal and performance sensors, and workload prediction to generate the best VM placement solution for the next epoch. VM placement problem is considered with the objective of minimizing the total energy consumption in a decision epoch while servicing all VMs in the cloud computing system. The formulation of this problem is called MERA for Multi-dimensional Energy-efficient Resource Allocation.

Furthermore, Generalized assignment Problem and Bin Packing Problem are both reduced to the MERA problem in this paper. In this paper a heuristic for solving the MERA problem is presented. It is an algorithm based on dynamic programming used to determine the number of copies for each VM and to assign these VMs to the servers, with the goal of minimizing the total energy cost of the active servers. The dynamic programming method first determines the set of candidate servers for copies of the VM to be placed, then uses dynamic programming to find the placement with the least cost Also, to improve the results, a local search method is considered to minimize the number of active servers as much as possible. The local search method servers are turned off based on the utilization and VMs are placed on the rest of the servers, if possible.

In conclusion, both papers have some distinct differences. Our VMR paper uses minimum cost flow to determine where the VM copies are placed and a server consolidation algorithm to consolidate servers (where consolidation does not incur an extra cost). In the USC paper, the authors reduced both the general assignment problem and the bin packing problem to the MERA problem in a special case and it uses dynamic programming and local search method to place VMs and consolidate servers. In our VMR paper, the number of copies each VM will have is chosen and not determined with an algorithm whereas in the USC paper, the number of copies each VM has is determined by the dynamic programming method with a maximum value of Li. The USC paper also considers SLAs whereas our VMR paper does not. Also, in our VMR paper each VM is of the same size, without the consideration of resource requirements where all the VM copies are also the same as the original. In the USC paper each VM can be different, with different resources requirements i.e. memory bandwidth and CPU cycles, where all the copies of any VM have the same memory BW as the original and the sum of all the CPU cycles of the copies must equal the CPU copies of the original VM. Our VMR paper focuses on data centers with a fat tree topology with a set of homogenous servers, all with the same capacity. The USC paper considers data centers with K different server types with different attributes such as memory bandwidth, CPU cycles and energy consumption.