An Autonomous Approach for the Scheduling of Virtual Machines Migration during Peak Loads

Martin Duggan, Jim Duggan, Enda Howley, Enda Barrett.
Department of Information Technology, National University of Ireland, Galway.
Keywords: Computer Science and Information Technology;

Abstract
Live virtual machine migration can have a major impact on how a cloud system performs, as it consumes significant amounts of network resources such as bandwidth. In this paper, we propose an autonomous network aware live migration strategy that observes the current demand level of a network and performs appropriate actions based on what it is experiencing. The Artificial Intelligence technique known as Reinforcement Learning acts as a decision support system, enabling an agent to learn optimal scheduling times for live migration while analysing current network traffic demand.

1. Introduction
Analysing and predictions of cloud network traffic in real time is becoming more prevalent to efficiently utilise limited resources during peak hours in cloud data centres. The aim of our research is to develop an autonomous network aware live migration strategy, that observes the current demand level of a network and perform an appropriate actions to better utilise cloud resources.

2. Related Work
Artificial Intelligence (AI) and Machine Learning techniques provide an opportunity for the development of a more cognitive cloud system. Reinforcement learning [1] has been successfully applied to a broad class of problems. Duggan et al. (2016) has applied reinforcement learning to live migration to decrease energy consumed in data centres [2]. The agent learns to select a VM to be migrated from an over-utilised host based on an energy-aware reward that will result a less migrations, energy consumption and energy costs. Work prior to this research Duggan et al. [3] used Reinforcement learning to determine optimal times schedule live migration from over-utilised host. This work is built upon in the research. Duggan et al [4] also examines how an autonomous learning agent decides appropriate times to schedule live migration from under-utilised host by observing cloud traffic demand patterns. They show, by analysing current bandwidth values RL can enhance live migration, reducing energy consumption and improve overall system performance based on a SLA.

3. Reinforcement Learning
Reinforcement Learning (RL) enables an agent to learn through a trial and error process. The agent interacts with its environment, by taking an action and observing the resulting reward from that action as illustrated in Figure 1.

In RL an agent has a set of states $S$ and a set of actions $A$. An Action ($a$) can move the agent from the current state ($s$) to a future state ($s'$), producing an outcome. Outcomes are given in the shape of positive and negative rewards that effect which actions to take. An RL agent will commence knowing nothing of its environment and through outcomes from different actions taken in a state, lead to the creation of its knowledge. After every state-action pair experienced a q-value $Q(s, a)$ is calculated. The following equation is the Q-learning algorithm, one of several algorithms that existing in RL.

$$Q(s, a) = Q(s, a) + \alpha (r + \gamma Qmax(s', a) - Q(s, a))$$

$\alpha$ denotes the alpha parameter which determine how quick an agent will learn. $\gamma$ denotes gamma, which is how an RL agent will perceive future rewards. Both $\alpha$ and $\gamma$ are set from 0 to 1. Qmax is the max value from the future state.

4. RL-Network-Aware VM Migration Strategy
For our research we consider the current bandwidth level and time in which to schedule live migration. An agent with the capabilities to learn from previous iterations the optimal action to perform when network traffic is high could better utilise cloud resources at peak times. We now describe the implementation of RL as a decision support system to the live migration process in the cloud’s IaaS infrastructure. To integrate RL into the IaaS
environment we first created a novel state and action space for the agent to observe and maximise rewards based on actions performed. The state-action space is denoted as $\langle bw, d, a \rangle$, where the current bandwidth level $bw$, the current direction of the network traffic $d$, and $a$ represents the action. The bandwidth level is determined by current usage with regard to threshold $t$ ranging from 0 to 100. The current direction of the network traffic is calculated by a simple moving average formula of the previous two time steps of bandwidth utilisation. The future state $(S' = bw_{t+1}, d_{t+1})$ are represented as follows: the future bandwidth level $bw_{t+1}$ is calculated by the moving average of the current bandwidth level and the previous. The future direction is determined by the moving average value of future $bw_{t+1}$ and current $bw_t$. The action as space is represented as wait or migrate. Should the RL agent select to wait then no migration will be deployed in that time-step. However, if the agent decides to migrate then a scheduled group of VM migrations will commence. The Reward ($r$) the agent receives on current and future bandwidth levels. If the agent chooses to migrate, it will receive a negative reward based on current bandwidth level. If the agents decides to wait until the next time step to migrate it will receive a negative reward based on the future bandwidth level.

5. Results
Figure 2, highlights the agent’s performance of 20,000 learning phases. In the graph when the agent’s action values are closer to 100%, this indicates the greater amount of migrations that are occurring. The closer the value is to 0%, highlights the percentage of the time the agent chose to wait (not migrate). The AvgWorkload is the network traffic occurring at each time step. The higher the bandwidth is the 100%, indicates the heavier traffic load occurring and closer the value is to 0%, indicates the lower traffic load occurring. As can be seen in figure 2, the agent learns to migrate when more resources are available. Figure 3, shows the agent is learning to maximise the rewards over the AvgWorkload. The plot highlights the learning rate of the agent. In the first few hundreds runs the agent receives severe negative rewards, as it is still exploring the environment. As the learning continues, the plot shows the rewards begin to decrease to a less severe negative reward and converge upon an average reward value.

6. Conclusion
This paper proposes an autonomous network aware VM migration strategy that acts as a decision support system. An RL-based approach learns to select the optimal time to schedule live migration from an over-utilised host. The learning agent observes real time the network resource usage and decides on the optimal action to perform. We believe the results shows the adaptive nature of Reinforcement Learning, through scheduling live migration when varying amount of network congestion arises.

7. References