**MDP mapping for workflow problem**

**Problem Statement**

A workflow (process) is a multi-step procedure toward a goal. Thus, a workflow contains multiple components. Each component can be assigned to different virtual machines on cloud with different cost associated. Availability of various virtual machines is dynamic. Assume we have done the workflow partition, input only contains Sequential pattern workflow. Mapping Markov Decision Process to model this problem and design a solution to gain workflow VM assignment recommendation.

**Input**: A workflow graph G

**Output**: A components assignment schedule for the workflow

**Sequential Pattern Case**

* Stable workflow with fixed number of components. Only one virtual machine can be assigned for a components.
* Sequential pattern for workflow.
* Price and execution time are the two primary concerns for MDP cost.
* Availability of various virtual machines for next state is only depend on current state and action will be taken. Availability list will be provided by virtual machine service. And reliability of virtual machine will be also provided.

**Workflow Model**

We model workflow applications as a Directed Acyclic Graph (DAG). Let n be the total number of components. Let T be the finite set of components.

Let m be the total number of virtual machine available. There are a set V of virtual machine, where Vi = 1 represents that this virtual machine is active and has been bound to a components while Vi = 0 means that this virtual machine is not active.

**MDP Model for Workflow Problem**

**State**

Definition: A state s∈S consists of current execution components and assigned virtual machine.

So S is a 2-dimensional mn matrix. With m rows represent components and n columns represent virtual machine.

represents that currently i components is executed under j virtual machine.

represents that this component is not active.

**Action**

Definition: An action a∈A in the model is to allocate a time slot on a virtual machine to a component.

So A is a 2-dimensional mn matrix. With m rows represent components and n columns represent virtual machine.

represents that allocating component i to virtual machine j with time slot a.

**Reward**

Definition: R(s,a,s') is the immediate reward obtained from taking action a at state s and transitioning to state s'. In our case reward represent cost associated. In our case, the cost is consist of two main factors, dollar mount involved, C and execution time, T.

**Transition**

Definition: A transition incurred by an action from one state to another will be success in probability of p(s|a,s'). In our case, we define reliability of a virtual machine as the transition probability.

**Solution**

The solution to an MDP is called a policy and it specifies the best action to take for each of the states. Although the policy is what we are after, we will actually compute a value function. MDP can be solved by using a standard dynamic programming algorithm such as value iteration and policy iteration. Value iteration computes a new value function for each state based on the current value of its next state. Value iteration proceeds in an iterative fashion and can converge to the optimal solution quickly.

**Workflow Objective function:**

**Workflow Optimal solution:**

Table 1. Algorithm

**Algorithm 1**. MDP workflow components assignment and schedule algorithm

**Input**: A workflow graph G

**Output**: A components assignment schedule recommendation for the workflow

1. Initialize s=, U()=R()
2. Repeat
3. Request processing time, price and availability of virtual machine from services
4. Update
5. For :
6. Record the policy set:
7. Record s and
8. Until goal state

**Discussion and Improvement**

**Large state space and action space**

In our model, since the virtual machine service environment is dynamic, besides indeterminacy if the input workflow is consist of complicated workflow pattern, like conditional, the state space and action space for MDP model will be very large, which will lower down the MDP performance greatly.

**Response time**

As we know, to model the entire workflow as an optimization problem will produce large scheduling overhead, especially for the problem with two dimension constraints such as time and cost.

At this stage, our model consider the cost as primary constraint. Sometimes, response time is also an important factor to measure. So in later research, time should be taken into consideration as another constraint besides cost.

**Transition Probability**

In term of transition probability, the original meaning of transition probability in MDP is to map imprecise or uncertain of state transition. However, for web or online service, research tend to treat this process as deterministic, for it can be reserved in advance. So our model consider reliability as only factor for mapping transition probability.

**POMDP**

Partially Observable Markov Decision Processes (POMDP), is a special case for MDP. Due to partial observability of service and system states, the POMDP approach provides better solutions for the QoS-aware service composition in dynamic workflow environments. Along with provenance information for service composition and selection, the POMDP method can be used for modeling a variety of real-world workflow processes.