Predictive AutoScaling Approaches for Kubernetes

1. Goal: To build and implement a controller that can allocate virtual machines and computing resources for a distributed application on a cloud by forecasting upcoming workload from previous workload in Kubernetes
2. General Process of Microscaling : (Trần 1) Talks about three steps needed to automatically scale our workload :
    - Monitoring resource/workload metrics: The actions of auto-scalers are based on performance indicators of the application obtained in the monitoring phase
    - Analyzing monitored data: Proactive autoscaling uses sophisticated techniques to predict future demands to arrange resource provision (Roy et al. 3), cites the ARMA model to predict workload from previous workloads. I wish to explore other algorithms, including ML approaches and see which one does better.
    - Deciding to scale/descale : (Trần 1) discusses 3 ways in which a cloud app can scale, while (Roy et al. 4) discusses the algorithm they used for vertical scaling, specifically the “look ahead algorithm.” I want to test other online approaches as well, including this one, like MCTS, and maybe also tackle/extend algorithms to hybrid scaling.
3. Potential Problems and possible fixes:
    - Building out a simulator for our application : its well known that to use MCTS algorithm you will need to explore the state space to a certain extend. I might not find an application that can simulate cloud apps, so I might need to build a simulator myself, which is very time-consuming.
    - Problems with mapping workloads to resource requirement : (Roy et al. 4), used a Customer Behavior Modeling Graph too model the overall behavior of customers, which seems to be a bayes net build on historical data, since we don’t have that I’m not sure how I should go about takling this problem.
    - Building the app itself: We need an app that can be dockerised/containerised and deployed in Kubernetes pods. I’m not sure if I should build this myself, and how does taking a prebuilt app affect how it will be deployed in a cloud?

Works Cited

Roy, Nilabja, et al. “Efficient Autoscaling in the Cloud using Predictive Models for Workload Forecasting.” https://www.researchgate.net/publication/221399885_Efficient_Autoscaling_in_the_Cloud_Using_Predictive_Models_for_Workload_Forecasting.

Trần. “A Survey of Autoscaling in Kubernetes.” https://www.researchgate.net/publication/362145963_A_Survey_of_Autoscaling_in_Kubernetes.