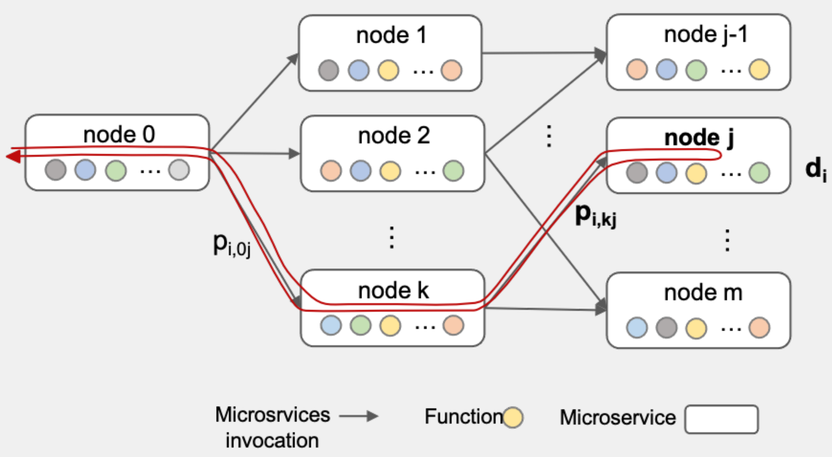
Demonstrating a System for Dynamically Meeting Management Objectives on a Service Mesh

Forough Shahabsamani Kim Hammar Rolf Stadler

KTH Royal Institute of Technology, Sweden

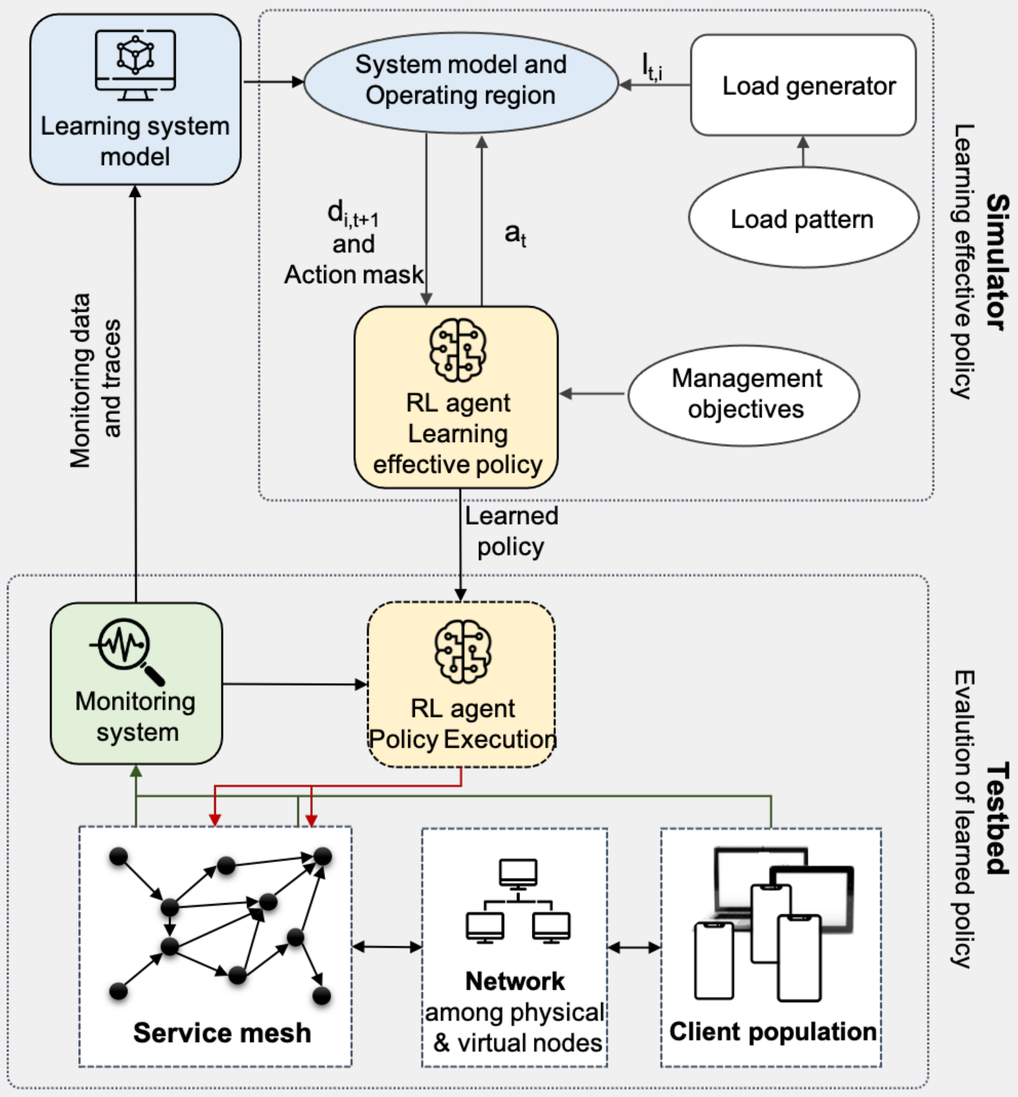
Managing microservice-based applications

- **Service mesh**. Services are built from microservice components, whereby each component has a dedicated functionality.
- **Service**. A service is a contiguous subgraph on a directed graph which we call the service mesh.
- **Service metrics**. We associate different metrics with a service S_i:
  - The (average) end-to-end delay d_i.
  - The offered load l_i.
  - The carried load l_i^c.
  - The utility u_i generated by the service.
- **Management objectives**. Management objectives capture the end-to-end performance objectives for the services on a given service mesh.



Solution framework

1. Define use case, management objective(s) & control actions.
2. Develop the RL model, including the reward function.
3. Define a scenario and run it on the target system.
4. Collect data in the form of system metrics and measurements.
5. Estimate the system model and the operating region.
6. Train and evaluate the RL agent on the simulator.
7. Evaluate the RL agent on the target system.

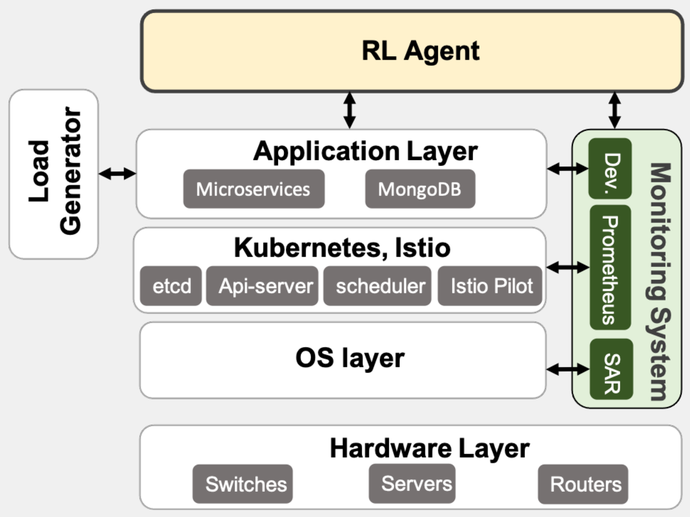


Demonstration

- Evolution of system metrics (e.g. end-to-end delay and throughput) for a given management objective.
- Reaction of the system to changing load conditions.
- Reaction of the system to changes in management objectives.

Target system: KTH testbed

- 12 compute and graphics servers, Linux, Kubernetes, Istio, MongoDB.
- Instrumented for real-time monitoring, learning, and prediction.
- Request generators emulate client populations.



Elements of management objective and controls

- **Management objective elements:**
  - End-to-end response time of a request of a service.
  - Throughput per service.
  - Utility of service metrics.
- **Management objectives for the demonstration:**
  - maximize $\sum_i l_i^c$ while $d_i < O_i$.
  - maximize $\sum_i u_i$ while $d_i < O_i$.
  - maximize l_i^c while $d_i < O_i$ and $l_k^c > l_{min}$ $i \neq k$.
- **Controls:**
  - Request routing per service.
  - Request blocking per service.

References

- F. S. Samani and R. Stadler, "Dynamically meeting performance objectives for multiple services on a service mesh," in 2022 18th International Conference on Network and Service Management (CNSM). IEEE, 2022, pp. 219–225.
- F. S. Samani, K. Hammar, and R. Stadler, "Demonstrating a System for Dynamically Meeting Management Objectives on a Service Mesh", in NOMS 2023 IEEE/IFIP Network Operations and Management Symposium, IEEE, 2023.
- F. S. Samani and R. Stadler, "A Framework for dynamically meeting performance objectives on a service mesh," submitted at IEEE Transactions on Network and Service Management.
- (**Source code**) F. Shahab Samani and K. Hammar, "Software framework for a management system based on RL, " https://github.com/foroughsh/Framework-for-dynamically-meeting-performanc-objectives, 2022.

This research has been partially supported by the Swedish Governmental Agency for Innovation Systems, VINNOVA, through project ANIARA.