Chair of Network Architectures and Services
Department of Computer Engineering
Technical University of Munich

Beyond Mean: Spatio-Temporal Modeling of Queue Utilizations and Flow Latencies Using T-GNNs

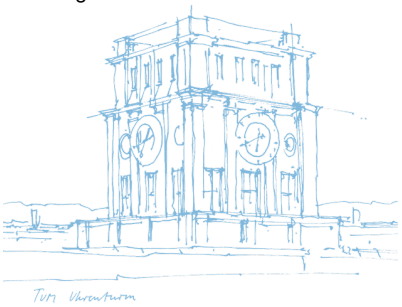
Max Helm, Benedikt Jaeger, Christopher Pfefferle, and Georg Carle

October 4, 2023

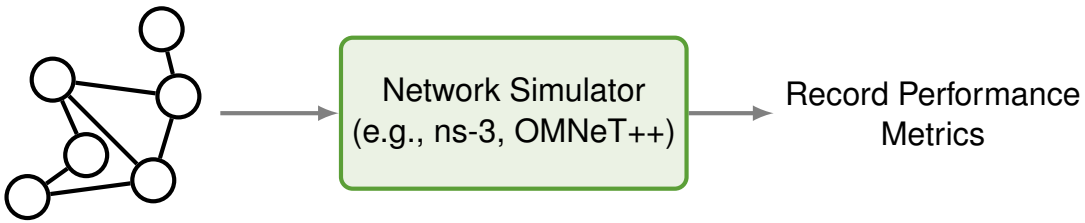
ITC 35

Turin, Italy

Chair of Network Architectures and Services
Department of Computer Engineering
Technical University of Munich



Motivation

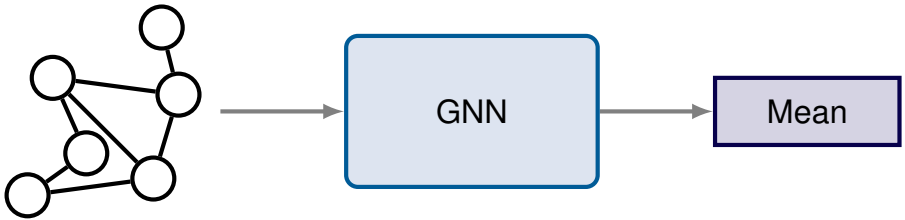

Network Simulator
(e.g., ns-3, OMNeT++)
Record Performance
Metrics

Motivation

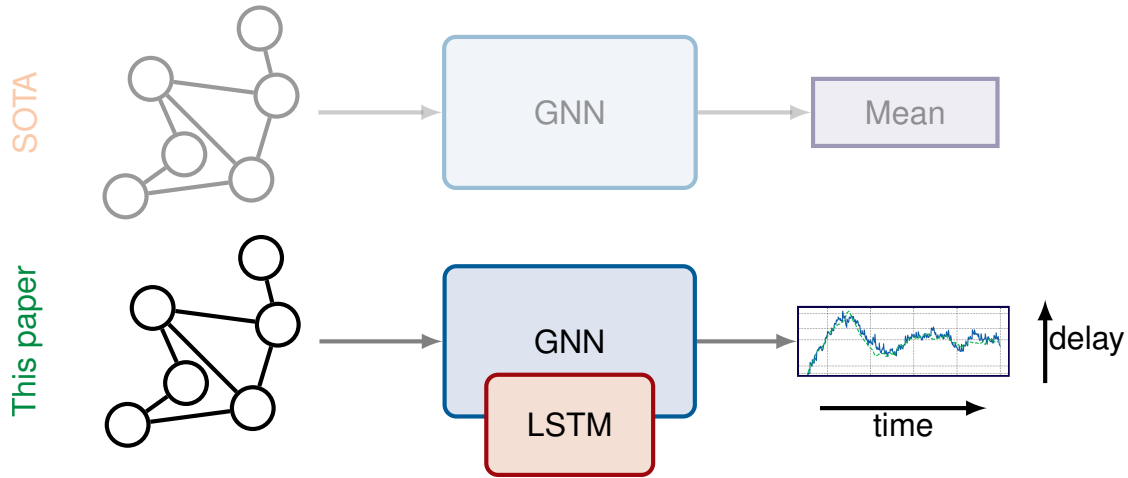
⇒ Save wall clock time by predicting instead of simulating

Motivation

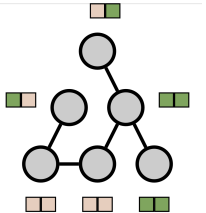
SOTA



Motivation



Background

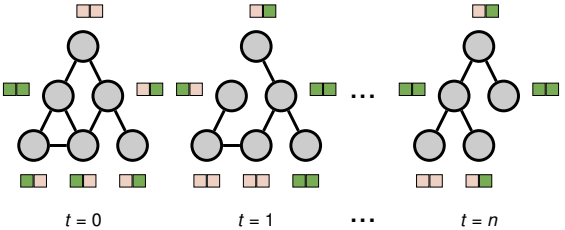


Graph Neural Networks:

- Graph: G = (V, E)

Temporal Graph Neural Networks:

- Set of graph snapshots: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, t)$



Methodology

Datasets

Dataset generation:

- 300 random network topologies (non-isomprohic graphs)
- UDP traffic
- Random, non-repeating on-off traffic models

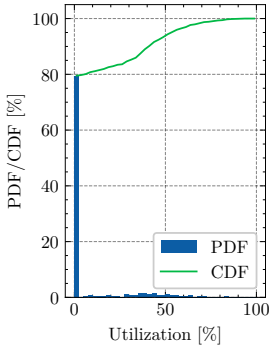
Dataset stats:

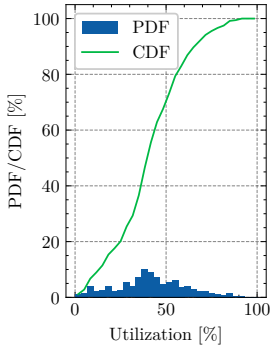
Dataset	Topologies	Snapshots	Queue Events
Train	140	2.5M	12.16M
Test	60	1.1M	5.21M
Evaluation	100	1.8M	8.69M

Methodology

Dataset Details

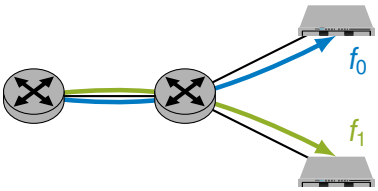
Parameter	Min	Max	Unit
Flow length	3	7	Hops
Flow rate	10	199	Mbit/s
End-to-end flow delay	0.81	196.14	ms
Packet Inter-arrival-time	0.0010	4.10	s

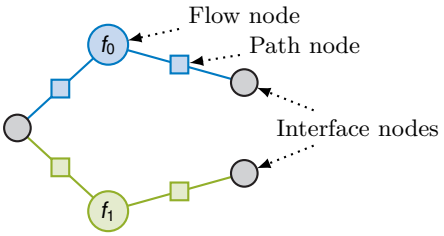




Methodology

Network Encoding



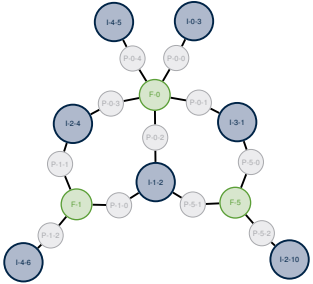


- Interface nodes encode egress interfaces of routers
- Flow nodes encode flow parameters and path
- Path order nodes encode flow direction

Methodology

Network Encoding Dynamic Example

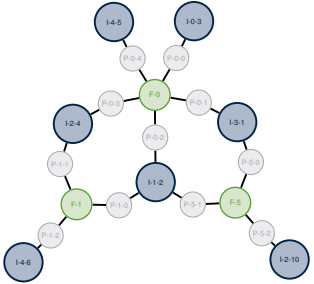
Graph at t=3.54s:



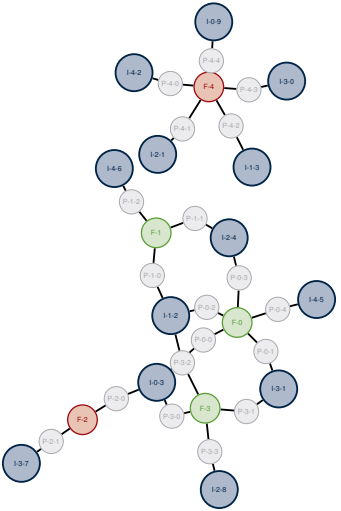
Methodology

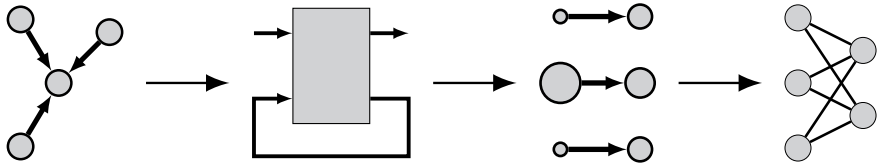
Network Encoding Dynamic Example

Graph at t=3.54s:



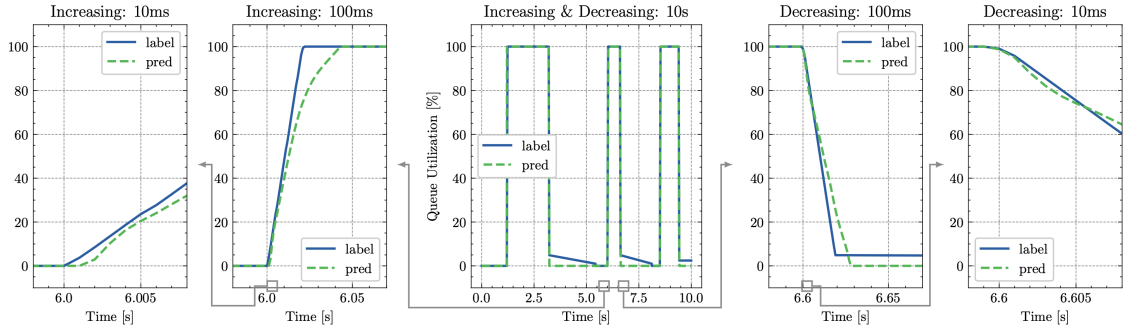
Graph at t=6.45s:




GGNN
LSTM
LayerNorm
FFNN

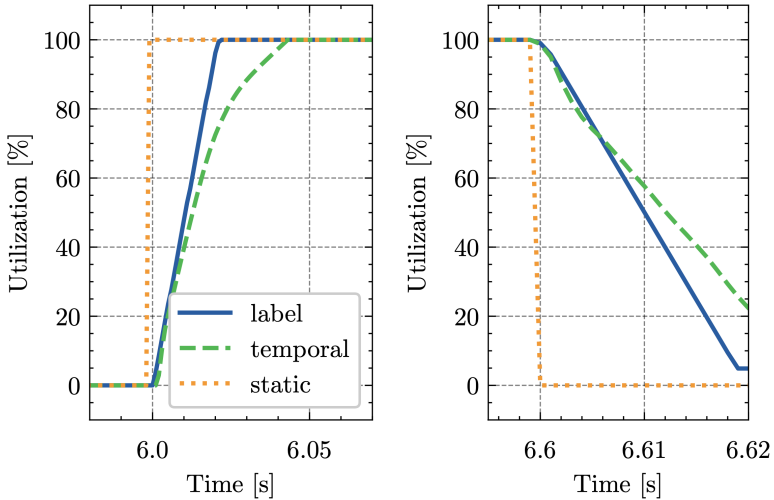
Results

Queue Occupancy Predictions



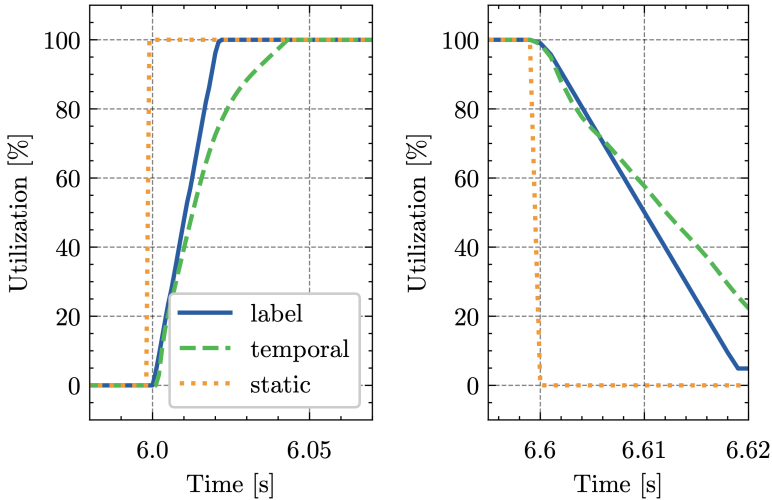
Results

Comparison to a Static GNN



Results

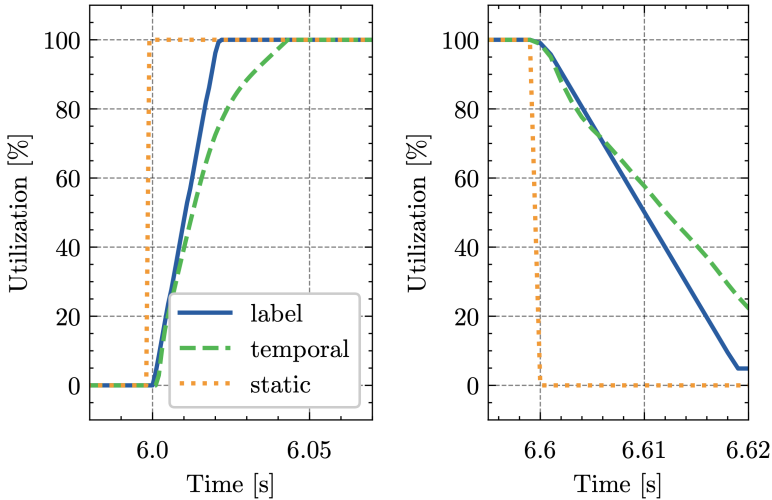
Comparison to a Static GNN



⇒ T-GNN out-performs classic GNN on the task of modeling queue filling- and draining stages

Results

Comparison to a Static GNN



⇒ T-GNN out-performs classic GNN on the task of modeling queue filling- and draining stages

(as expected)

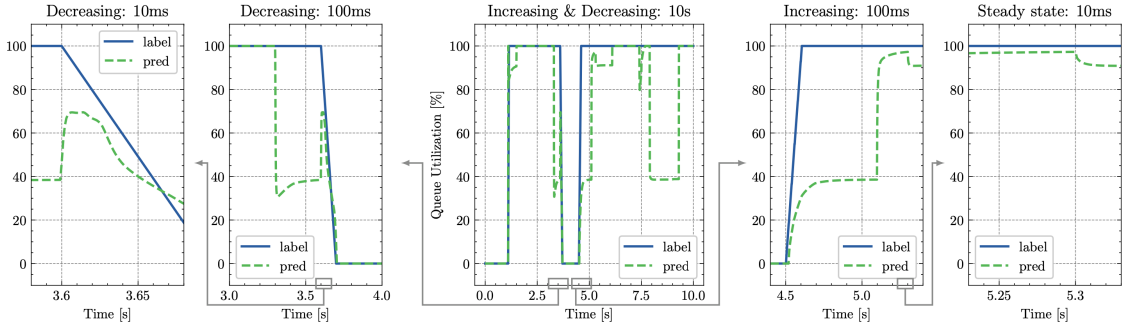
Does the T-GNN model always perform well?

Does the T-GNN model always perform well?

Does it fail under certain circumstances?

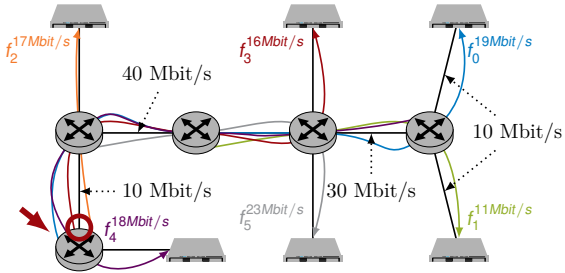
Results

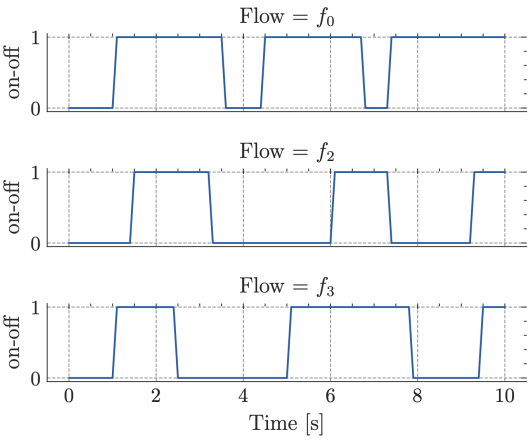
Failure Case Analysis



Results

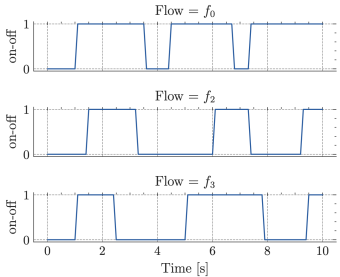
Failure Case Analysis

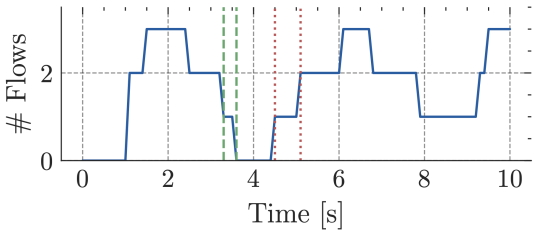




Results

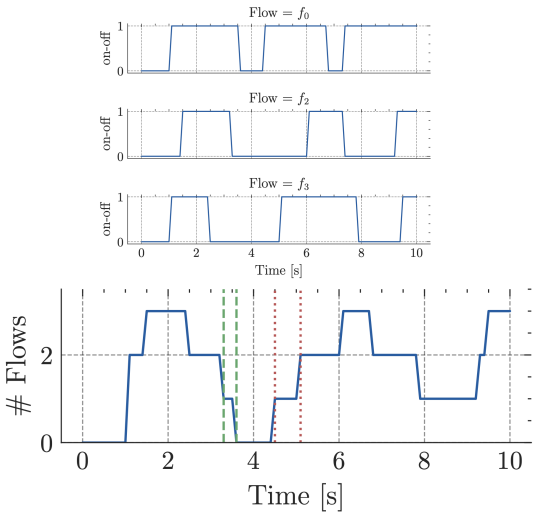
Failure Case Analysis

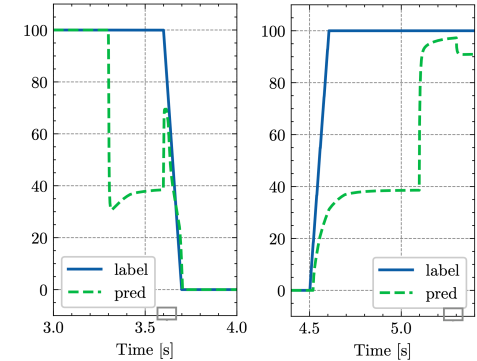




Results

Failure Case Analysis

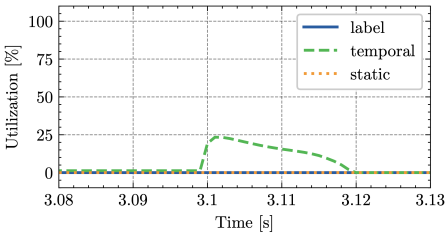




- Drop caused by leaving flow, link still saturated, corrected 300ms later
- Build-up is delayed to wait for second flow, LSTM reserves capacity
- Utilization share equals flow rate ratio

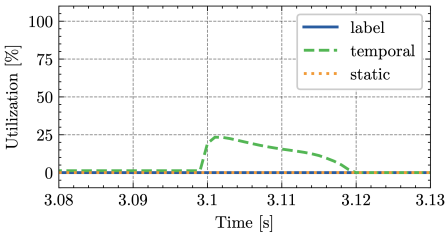
Results

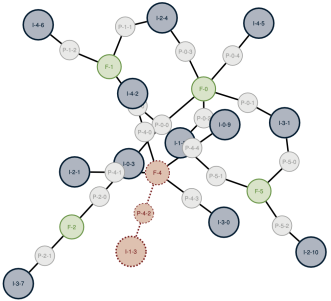
Failure Case Analysis



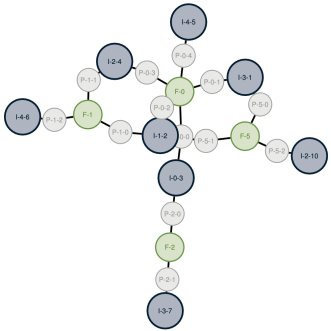
Results

Failure Case Analysis





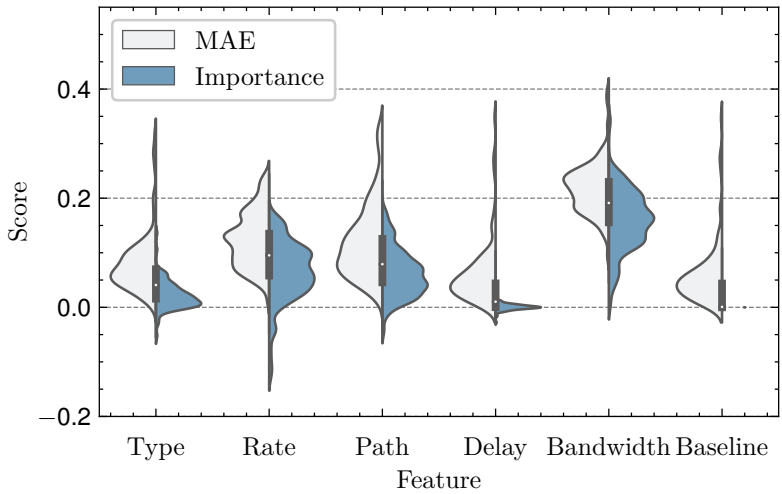
(a) $t = 3.09$



(b) $t = 3.11$

Results

Decision Process: Feature Importance



Conclusion

Contributions:

- Model of temporal queue utilizations using a T-GNN
- Temporal information allows direct derivation of quantiles
- Analysis of shortcomings of the method
- Published dataset

In the paper:

- Comparison to Network Calculus
- End-to-end flow-level delay derivation from queue utilizations
- Placement in relation to other SOTA approaches
- Second order feature importance

helm@net.in.tum.de