Uncertainty-driven Ensemble Forecasting of QoS in Software Defined Networks

Kostas Kolomvatsos\textsuperscript{1}, Christos Anagnostopoulos\textsuperscript{2}, Angelos Marnerides\textsuperscript{3}, Qiang Ni\textsuperscript{3}, Stathes Hadjiefthymiades\textsuperscript{4}, Dimitrios Pezaros\textsuperscript{2}

\textsuperscript{1}University of Thessaly, \textsuperscript{2}University of Glasgow, \textsuperscript{3}University of Lancaster, \textsuperscript{4}University of Athens

22\textsuperscript{nd} IEEE Symposium on Computers and Communications
03-06 July 2017, Heraclion, Crete, Greece
Outline

• SDNs Management
• The proposed framework
• The ensemble forecasting scheme
• The uncertainty management mechanism
• Experimental evaluation
SDNs Management (1/2)

- SDN controllers are responsible for performing various network tasks
- They are connected to applications through the northbound interface and to devices through the southbound interface
- They incorporate monitoring functionalities to collect time-series network performance data
- Example metrics: latency, link utilization, switch buffer occupancy, etc
SDNs Management (2/2)

- During the functioning the amount of data becomes huge
- We want to derive predictive analytics on top of the data
- Our aim is to use the analytics to secure the QoS
The proposed framework

- We manage various *Network Performance Parameters* (NPPs)
- We propose a module that supports real time decision making
- Our module involves:
  - A Type-2 Fuzzy Logic System (T2FLS) for uncertainty management (it derives alerts about the presence of QoS violation events)
  - The T2FLS derives the *Potential of Violation* (PoV)
  - A combination of responses derived through large-scale predictive analytics
  - A combination of multiple aggregated time series forecasting results
- Our goal:
  - Provide forecasting analytics to the SDN controller
  - For each NPP we provide an aggregated value
  - The T2FLS informs the SDN controller for the presence of an event
The ensemble forecasting scheme (1/3)

- We consider a set of estimators
- Examples: auto-regressive estimator, double and triple exponential smoothing, weighted and cumulative moving average, etc
- We adopt 28 estimators
- The ensemble scheme involves the aggregation of multiple estimators for each NPP
- The final aggregated value is derived through an aggregation function on top of historical values

\[ \hat{e} = f(e_1, \ldots, e_{|\varepsilon|}) \]
The ensemble forecasting scheme (2/3)

- We adopt a linear aggregation function
  \[ \hat{e} = \sum_{i=1}^{\left|\mathcal{E}\right|} w_i e_i \quad \sum_{i} w_i = 1 \]
- Each estimator has a specific weight
- Our model produces a vector of aggregated estimations (a value for each NPP)
- Weights are defined based on estimators’ performance
- The performance is affected by the estimation error \( |\hat{e}_i^t - r_k^t| \)
- \( \hat{e}_i^t \) is the estimation and \( r_k^t \) is the real observation for the \( k \)-th NPP
The ensemble forecasting scheme (3/3)

- We adopt a sliding window approach
- We determine the weight of an estimator based on its performance in the window
- Weights are based on the average forecasting error $\mu_i$
- We define a convex combination rule for weights definition
  \[
  w_i = \frac{1 - \mu_i}{\sum_{j=1}^{T} 1 - \mu_j}
  \]
- The mechanism assigns high weight to estimators with low average error
Uncertainty management (1/2)

• We want to ‘fire’ the update on the orchestration process of the SDN controller
• Uncertainty is present on how the aggregated estimation depicts a high potential of QoS violation
• We propose the T2FLS for such purposes
• The T2FLS linearly maps the inputs to the outputs
• It adopts as set of rules

\[ R_j: \text{If } u_{1j} \text{ is } A_{1j} \text{ and/or } u_{2j} \text{ is } A_{2j} \text{ and/or } ... \text{ and/or } u_{lj} \text{ is } A_{lj} \text{ Then } v_{1j} \text{ is } B_{1j} \text{ and } ... \text{ and } v_{zj} \text{ is } B_{zj} \]

• \( u_{ij} \) are the inputs, \( v_{kj} \) are the outputs and \( A_{ij} \) and \( B_{kj} \) are membership functions
Uncertainty management (2/2)

- Membership functions in Type-2 FLSs are intervals

- Our T2FLS has $|M|$ inputs; $M$ is the set of NPPs
- Each input corresponds to the aggregated measurement for an NPP
- We consider three linguistic values: Low, Medium, High
- The output is the PoV
- When PoV is over a pre-defined threshold the T2FLS fires an event to the SDN controller
Experimental evaluation (1/3)

- Experimental setup
  - We focus on: link utilization $\beta$, average switches buffer size $\varepsilon$ and average latency $\alpha$
  - We adopt two distributions to produce values for each metric: Uniform, Exponential ($\lambda=0.5$, $\lambda=2.0$)
  - We adopt three decision thresholds: $\beta_T$, $\varepsilon_T$, $\alpha_T$
  - When NPPs violate the thresholds, there is an indication of QoS violation
  - We adopt known performance metrics for our model like:
    - Precision
      $$\pi = \frac{TP}{TP+FP}$$
    - Recall
      $$\rho = \frac{TP}{TP+FN}$$
    - Accuracy
      $$\psi = \frac{TP+TN}{TP+TN+FP+FN}$$
    - F-measure
      $$\phi = 2\frac{\pi \cdot \rho}{\pi + \rho}$$

TP: true-positive, TN: true-negative, FP: false-positive, FN: false-negative
Experimental evaluation (2/3)

• We run 1,000 simulations for 1,000 rounds per simulation
• At t, we collect the realization for NPPs and execute the proposed scheme
• Three scenarios are evaluated:
  • Scenario A: $\beta_T = \varepsilon_T = \alpha_T = 0.5$
  • Scenario B: $\beta_T = 0.3$, $\varepsilon_T = \alpha_T = 0.7$
  • Scenario C: $\beta_T = 0.7$, $\varepsilon_T = \alpha_T = 0.3$
Experimental evaluation (3/3)

- Results (|E| is the number of estimators)

<table>
<thead>
<tr>
<th></th>
<th>Scenario A</th>
<th>Scenario B</th>
<th>Scenario C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Uniform</td>
<td>Exponential</td>
<td>Uniform</td>
</tr>
<tr>
<td></td>
<td>λ=0.5</td>
<td>λ=2.0</td>
<td>λ=0.5</td>
</tr>
<tr>
<td>π</td>
<td>0.96</td>
<td>0.95</td>
<td>0.79</td>
</tr>
<tr>
<td>ρ</td>
<td>0.76</td>
<td>0.69</td>
<td>0.83</td>
</tr>
<tr>
<td>φ</td>
<td>0.85</td>
<td>0.80</td>
<td>0.81</td>
</tr>
<tr>
<td>ψ</td>
<td>0.76</td>
<td>0.70</td>
<td>0.74</td>
</tr>
</tbody>
</table>
Thank you for your attention!