On the Design of Resource Allocation Algorithms for Low-Latency Video Analytics

Víctor Valls (Yale University), Heesung Kwon (U.S. Army Research Laboratory), **Tom La Porta** (Pennsylvania State University), **Sebastian Stein** (University of Southampton), **Leandros Tassiulas** (Yale University)

Motivating Application

Video analytics that combines object detection and image classification algorithms. The goal is to obtain a detailed description of a scene in real-time; e.g., for video surveillance.

Figure 1. Example of an object detection output.

Procedure: First, an object detection algorithm extracts classes of objects in a video frame. The objects are later on analyzed by specialized image classification algorithms to obtain a more detailed description of the objects.

Setup: Cameras offload video streams to the cloud where the data analytics tasks are carried out and the results delivered to the end-users.

Problem: Low-Latency vs Maximum Resource Utilization

Problem: In the cloud, real-time data analytics tasks have to coexist with throughput intensive applications (e.g., Hadoop). The latter usually congest the network and increase the latency of *all* the tasks/applications running in the system.

Existing solution for low-latency data analytics: Over-provisioning; sacrifice bandwidth and computation resources to keep the congestion low and so obtain low-latency (e.g., JUMP, Fastpass, Heron).

Research question:

Can real-time data analytics coexist with other cloud applications at a zero-throughput cost?

Contributions

- A resource allocation model for data analytics tasks with in-network processing
- An online algorithm that
  a) provides low-latency to data-analytics tasks that have a constant resource demand rate;
  b) maximizes the use of bandwidth/processing resources in the system (even when some data analytics have time-varying resource demand).

Key observation: If there are no queues, data packets cannot “wait”.

No queues ⇒ No congestion ⇒ No delay

Proposed online algorithm:

$$\underset{x_{ij}}{\text{maximize}} \quad \sum_{c \in C \setminus C'} \underbrace{(Q_i^{(c)}[k] - Q_j^{(c)}[k])}_{\text{difference queue length}} \; x_{ij}^{(c)}[k]$$

$$\text{subject to} \quad \lambda_i^{(c)} + \sum_{j \in N} x_{ji}^{(c)}[k] = \sum_{j \in N} x_{ij}^{(c)}[k], \quad \forall c \in C'$$

where C' is the set of data analytics tasks with constant resource demand rate.

Numerical example:

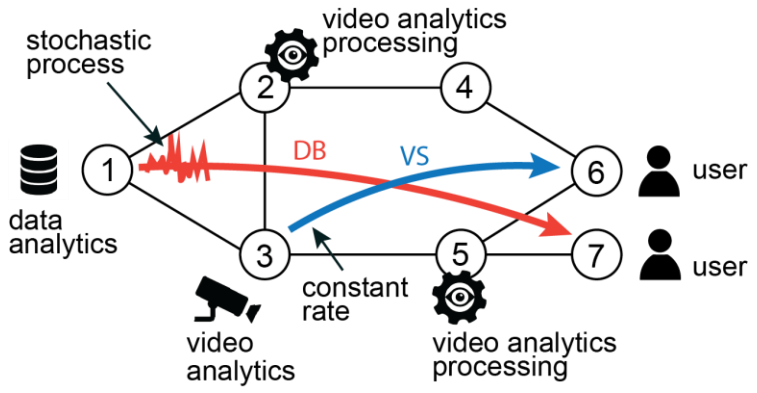

Figure 3. Scenario with a data analytics task (no delay-sensitive) and a video analytics (delay sensitive)

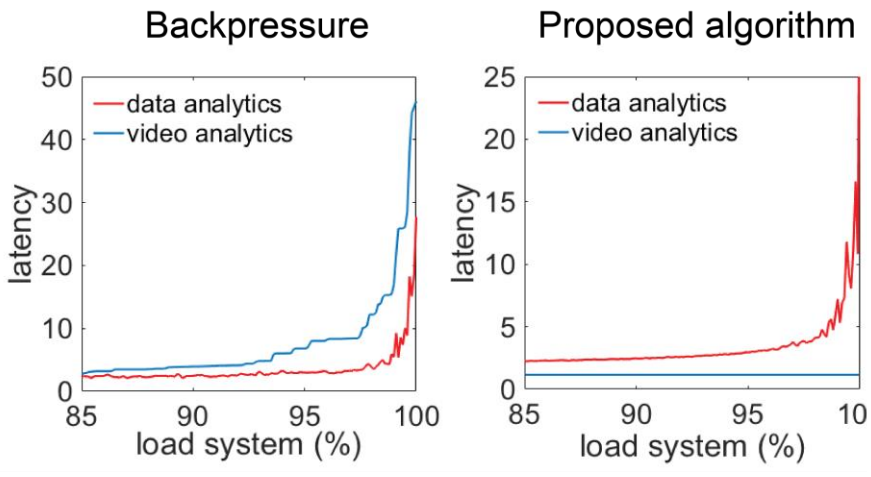

Figure 2. Illustrating the latency (in normalized time slots) against the system load.

Distributed Analytics and Information Science International Technology Alliance
Annual Fall Meeting, September 2018