Latency-Aware Strategies for Placing Data Stream Analytics onto Edge Computing
Alexandre da Silva Veith, Marcos Dias de Assuncao, Laurent Lefèvre

To cite this version:
Alexandre da Silva Veith, Marcos Dias de Assuncao, Laurent Lefèvre. Latency-Aware Strategies for Placing Data Stream Analytics onto Edge Computing. HotEdge’18 - USENIX Workshop on Hot Topics in Edge Computing, Jul 2018, Boston, United States. pp.1, 2018. hal-01875999

HAL Id: hal-01875999
https://inria.hal.science/hal-01875999
Submitted on 18 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Latency-Aware Strategies for Placing Data Stream Analytics onto Edge Computing

Alexandre da Silva Veith, Marcos Dias de Assunção, Laurent Lefèvre
Avalon, LIP, ENS Lyon, University of Lyon - Lyon, France
alexandre.veith@ens-lyon.fr

Scenarios
Monitoring of operational infrastructure
Anomaly detection, fraud detection
Social networks
Internet of Things (IoT)

Applications
Data Source
Operators
Sink
Directed dataflows whose vertices are operators that execute a function over the incoming data and edges define how data flows between them

Resource and Application Models
Example of operator placement
Resource 1
Dispatching
Op. 1
Data transfer
Op. 2
Network
Op. 3
Message
Op. 4
Directly connected
Not directly connected
Operators - communication and computation services
Services - M/M/1 queues

Finding Application Patterns
Initial setup
Pattern recog.
Transform
Series hierarchy
Phases
Phase 1
Phase 2
Phase 3
Phase 4
- Find the dataflow split points (green)
- Edge (red) and cloud (blue) placements

Response Time Rate (RTR) Strategy
1) BFS-Traversal algorithm = operator sequence
2) For each operator in the sequence
   - Response time estimation for all resources
   - Host with shortest response time is elected to host the operator

RTR with Region Patterns (RTR+RP) Strategy
Regions definition
Candidate placement
Final placement
Operator sequence = RTR
Candidate placement = destination infrastructure of the message
- Only cloud = cloud
- Edge and/or cloud = edge
Response time estimation - only for edge candidates

Response time (RT): the total time taken for a message to traverse a path

Latency-Aware Decisions
Operator Characteristics and Requirements (CPU, Memory, Bandwidth)
Minimize the response times in all paths
Edge and Cloud Capabilities (CPU, Memory, Bandwidth, Latency)

Analysis
Our approach: Microbenchmarks
- 95% better
Complex Applications
- 50% faster in average
- 52% of reduction in the communication for sinks located in the edge

Conclusions
- Dynamically movement of operators across edge and cloud
- 50% faster in generic and complex dataflows

Reference

Experimental setup
Cloudlet 2
Gateway
Internet
Cloud
Edge devices: 2 cloudlets with 20 Raspberry Pi 2 (4,74 MIPS at 1GHz and 1GB of RAM)

Cloud: Two AMD RYZEN 7 1800x (304,51 MIPS at 3.6GHz and 1TB of RAM)
LAN:
- Latency: U(0.015-0.8)ms
- Bandwidth: 100 Mbps
WAN:
- Latency: U(65-85)ms
- Bandwidth: 1 Gbps

Complex Applications - Investigate the outcomes for generic and multiple path applications

Acknowledgment: This work was performed within the framework of the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program “Investissements d’Avenir” (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).