INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING
ICPP/2021/CHICAGO/USA
acm In-Cooperation
sighpc
AUGUST 9-12, 2021

Context-aware Data Operation Strategies in Edge Systems for High Application Performance

Tanmoy Sen and Haiying Shen
Department of Computer Science
University of Virginia

UNIVERSITY of VIRGINIA

Intelligent Cognitive Assistants (ICAs): The Future

Edge Computing

Intelligent Cognitive Assistants

Machine Learning/AI

- ICAs assist working, learning, transportation, healthcare, an smart city
  - Traffic accide
  - Parking suggestion
  - Detect heart attack
  - Detect Covid-19

Intrado

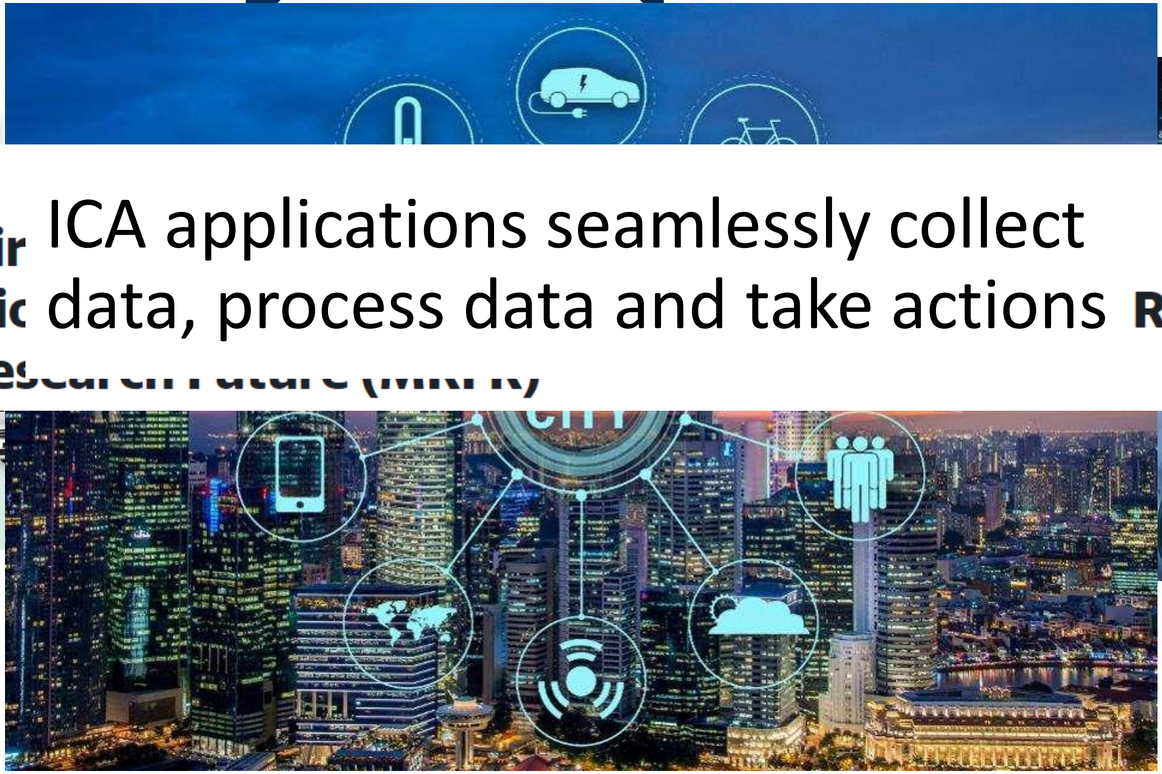
Intelligent Vir

47,259.2 Millic

Report

by Market Res

ICA applications seamlessly collect data, process data and take actions



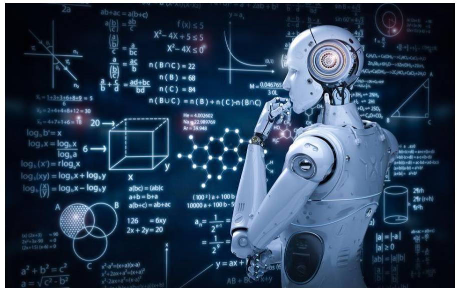
INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

acm In-Cooperation sighpc

Intelligent Cognitive Assistants (ICAs): The Future

Edge Computing

Machine learning/AI



- constrained power and the bandwidth
- power and bandwidth consuming

- Challenge for the marriage for ICAs:
  - Achieve low job latency with low power and bandwidth consumption
- Focus on data operations

INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

acm In-Cooperation sighpc

Related Work and Novelty

- Data placement: where to store sensed data (ICFEC'17, ASAC'18, TC'19)
  - Only on source data
  - Still consumes high bandwidth and power
- Data collection: decrease the transmitted data samples (ICCPS'15, IACC'15, ICPADS'18, TMC'19)
  - Do not consider influence on AI accuracy
- Novelty of our system: Context-aware Data Operation System (CDOS)
  - Overcome the limitations
    - Data sharing and placement (CDOS-DP)
    - Context aware data collection (CDOS-DC)
    - Data redundancy elimination (CDOS-RE)

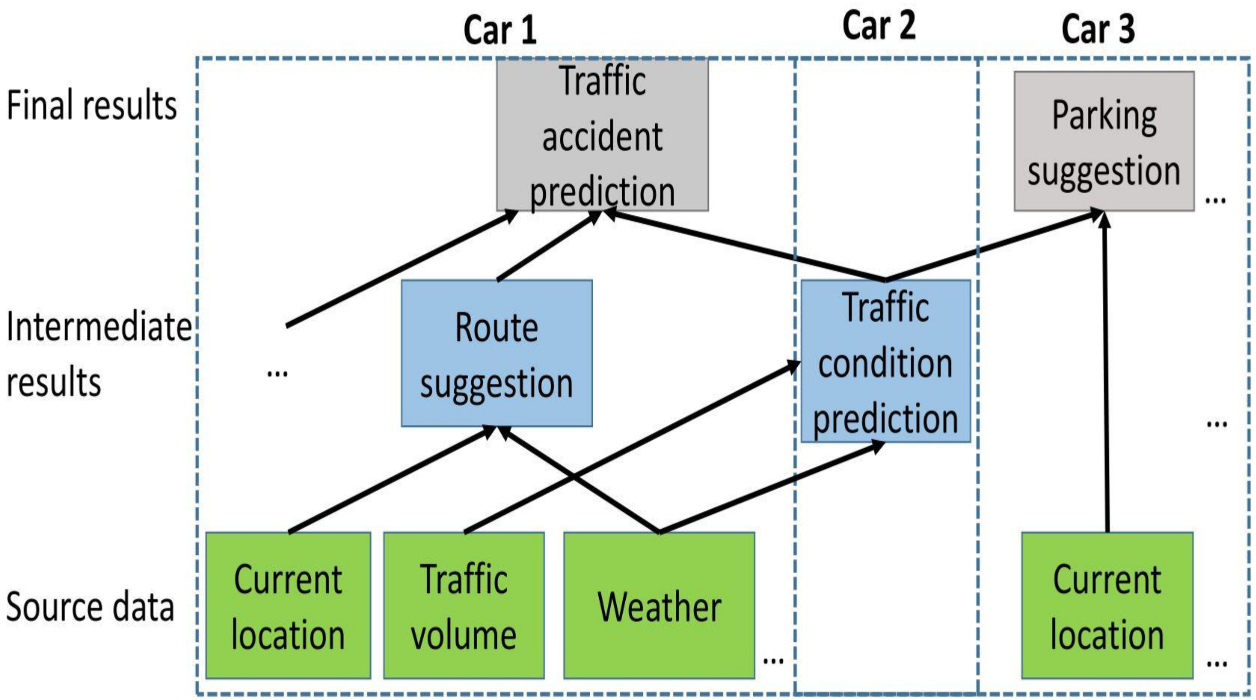
Data Sharing and Placement

Challenge

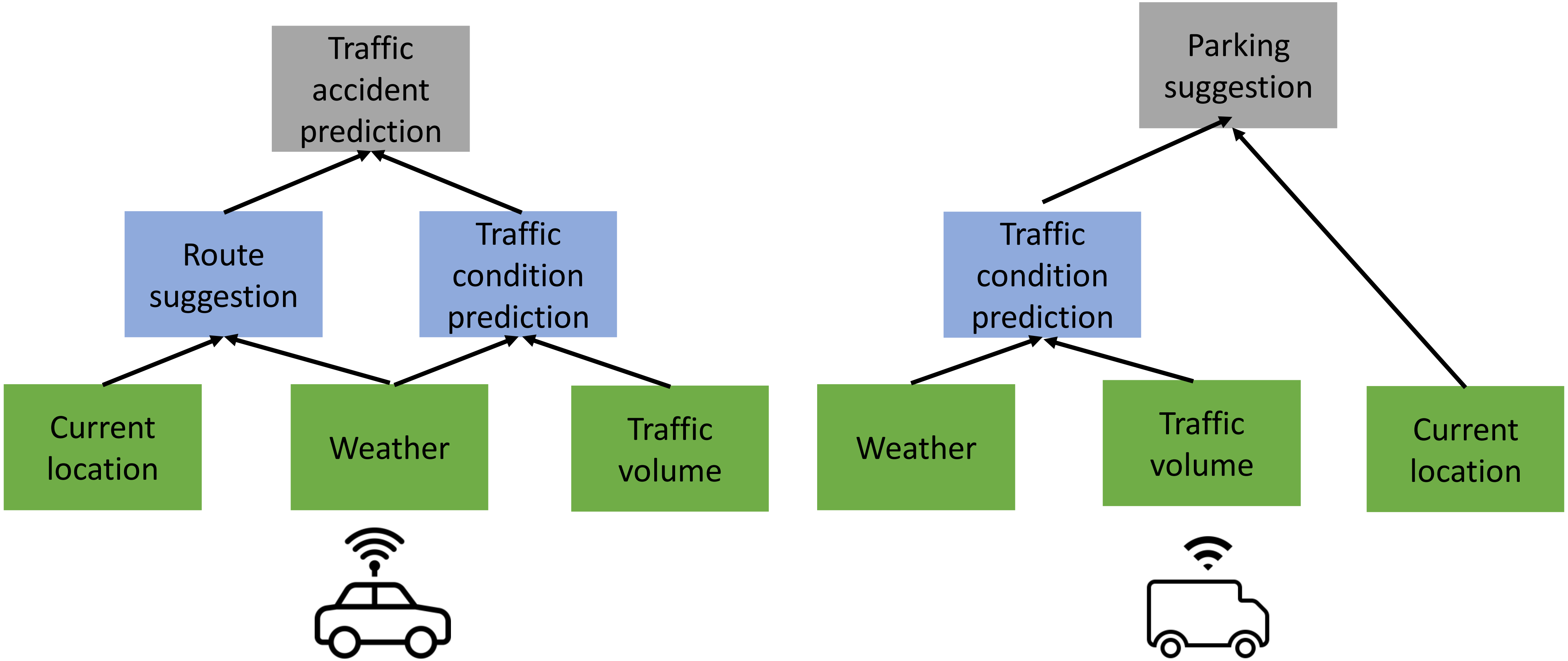
- Collecting source data and sharing source data still consume high power and bandwidth

Rationale

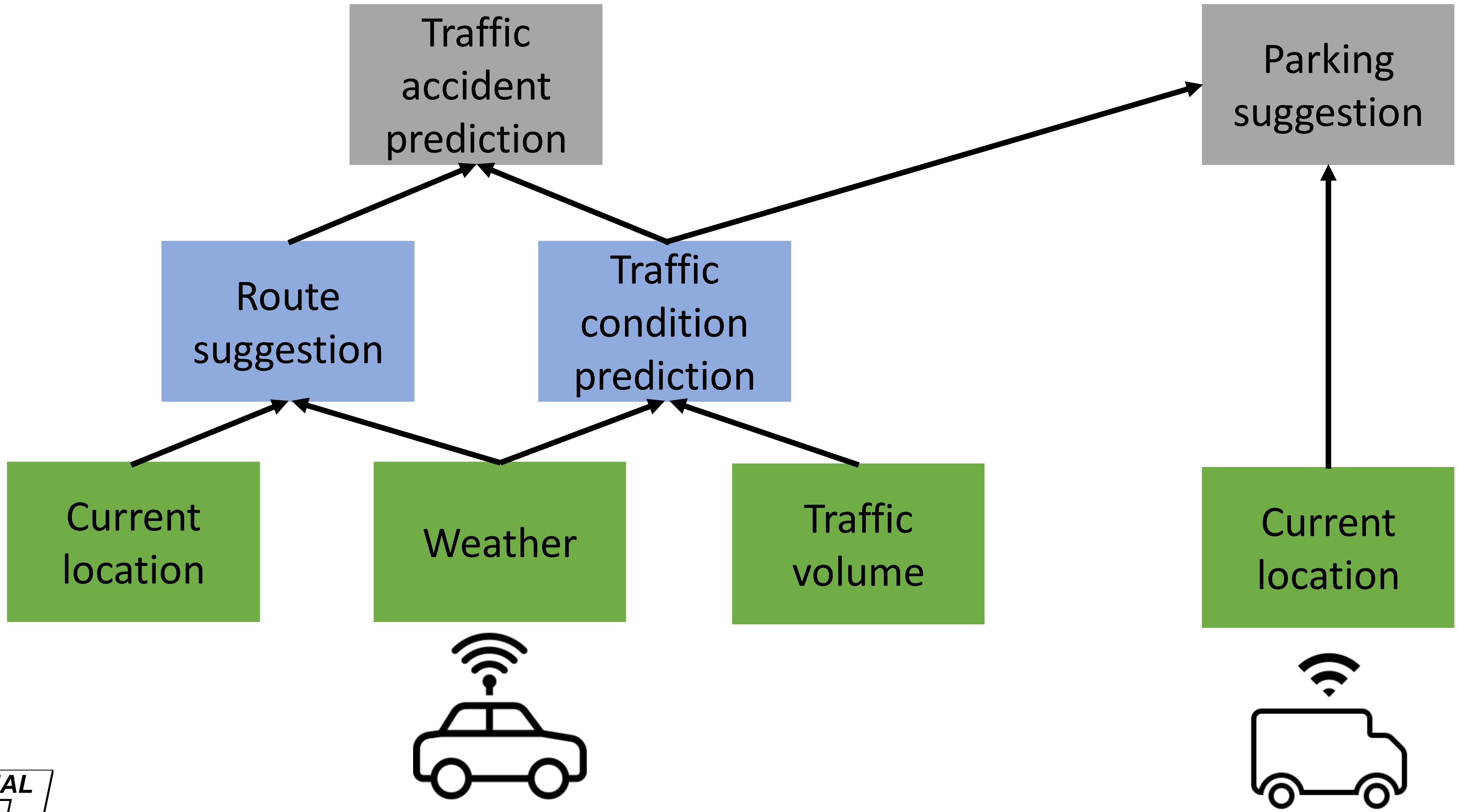
- Intermediate and final data results may be shared by many jobs



Data Sharing and Placement (cont.)



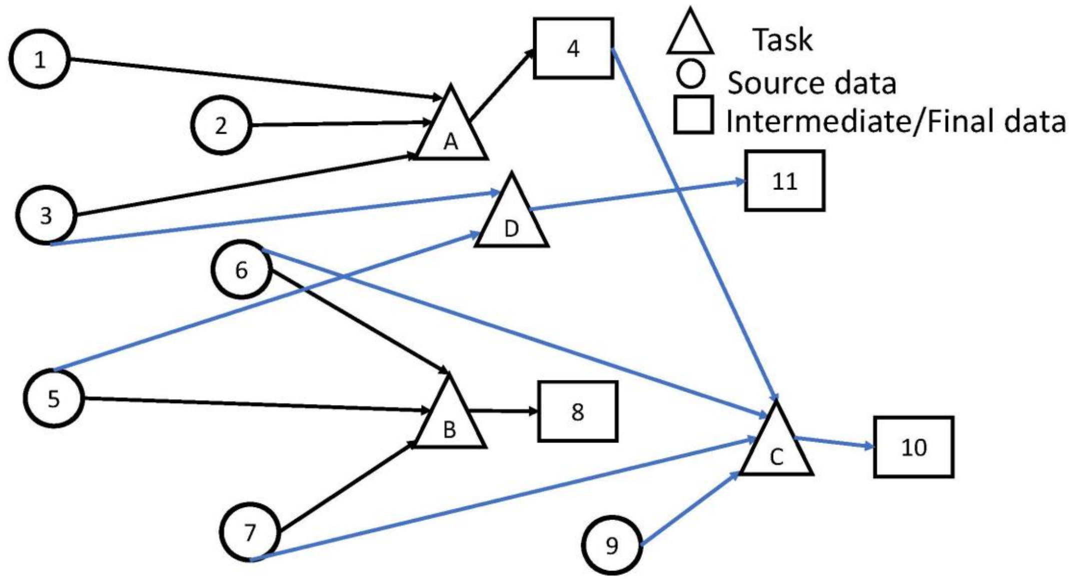
Data Sharing and Placement (cont.)



Data Sharing and Placement (cont.)

Strategy

- Storing intermediate and final computation results for sharing
- Use dependency graph
- Placement: linear programming problem with aim to minimize communication overhead and latency



Communication overhead for storing and fetching data

$$\text{Min: } \forall_{n_s \in N} \forall_{d_j \in D_g, n_g \in N_g} \boxed{C(n_g, n_s, d_j, N_d^{d_j})} \cdot \boxed{L(n_g, n_s, d_j, N_d^{d_j})} \cdot \boxed{x(d_j, n_s)}$$

Selected node

Time latency for storing and fetching data

Context aware Data Collection

Challenge:

- Reduce data sampling frequency without compromising AI accuracy

Context-related Factors

- Abnormality of data
- Priority of Events
- Data Weight on Computation Result
- Context of an Event

Constant objects

Frequency: Low

Pedestrian

High

Context aware Data Collection (cont.)

Challenge

- Reduce data sampling frequency without compromising decision making accuracy

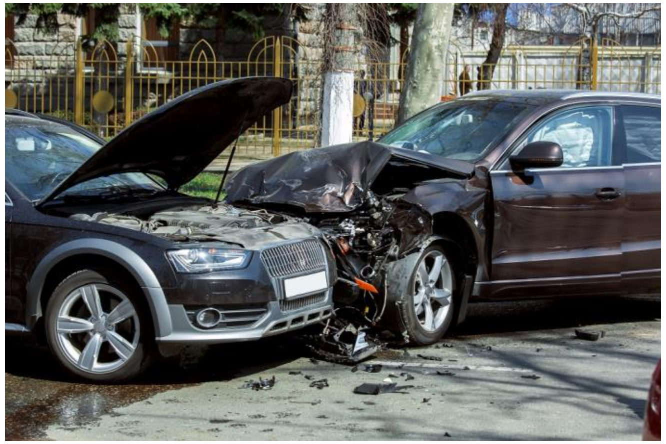
Context-related Factors

- Abnormality of data
- Priority of Events
- Data Weight on Computation Result
- Context of an Event

Traffic prediction

Frequency: Low

Car accident prediction



High

Context aware Data Collection (cont.)

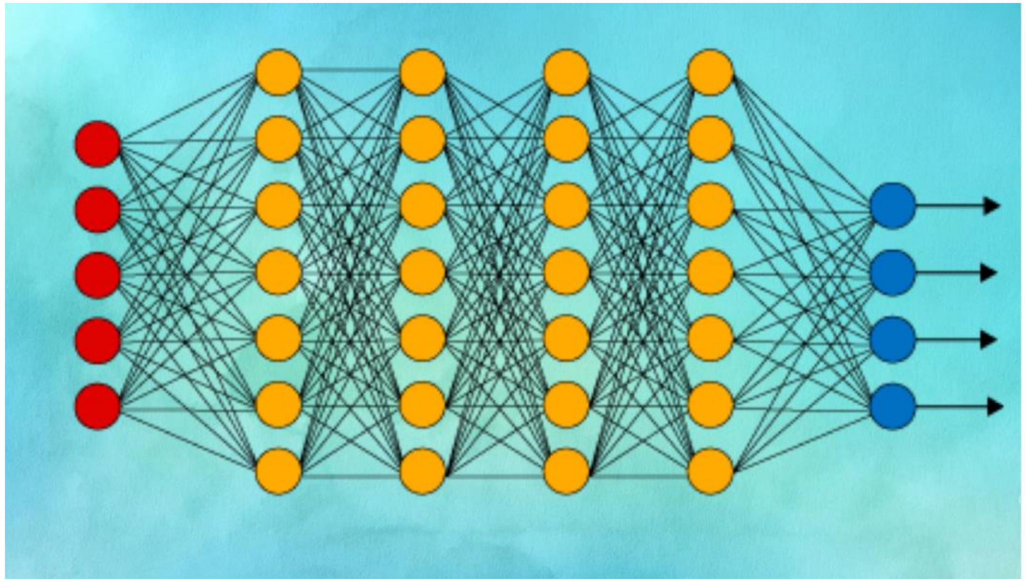
Challenge

- Reduce data sampling frequency without compromising decision making accuracy

Context-related Factors

- Abnormality of data
- Priority of Events
- Data Weight on Computation Result
- Context of an Event

Weight of each input



Weight: Time>temperature for traffic prediction

Context aware Data Collection (cont.)

Sunny weather, light traffic

Frequency: Low

Rainy weather, moderate traffic

High

Challenge

- Reduce data sampling frequency without compromising decision making accuracy

Context-related Factors

- Abnormality of data
- Priority of Events
- Data Weight on Computation Result
- Context of an Event

Context aware Data Collection (cont.)

Strategy

- Change data collection frequency based on cumulative weight of the four factors

$$\boxed{W_{d_j}} = \sum\nolimits_{e_i \in E_j} \boxed{w^1_{d_j} \cdot w^2_{e_i} \cdot w^3_{d_j,e_i} \cdot w^4_{e_i},} \ (0 < W_{d_j} \leq 1)$$

Context aware Data Collection (cont.)

Strategy

- Change data collection frequency based on cumulative weight of the four factors

$$\boxed{W_{d_j}} = \sum\nolimits_{e_i \in E_j} \boxed{w^1_{d_j} \cdot w^2_{e_i} \cdot w^3_{d_j,e_i} \cdot w^4_{e_i},} (0 < W_{d_j} \leq 1)$$

- Additive linear increase multiplicative decrease (AIMD) algorithm to tune the collection time interval

$$\boxed{T_{t+1}} = \begin{cases} T_t + \alpha / \eta W_{d_j} \ (\alpha \geq 1), & \text{all errors are within their limits} \\ \dfrac{T_t}{\beta + \eta W_{d_j}} \ (\beta \geq 1), & \text{otherwise.} \end{cases}$$

Data Redundancy Elimination

Challenges

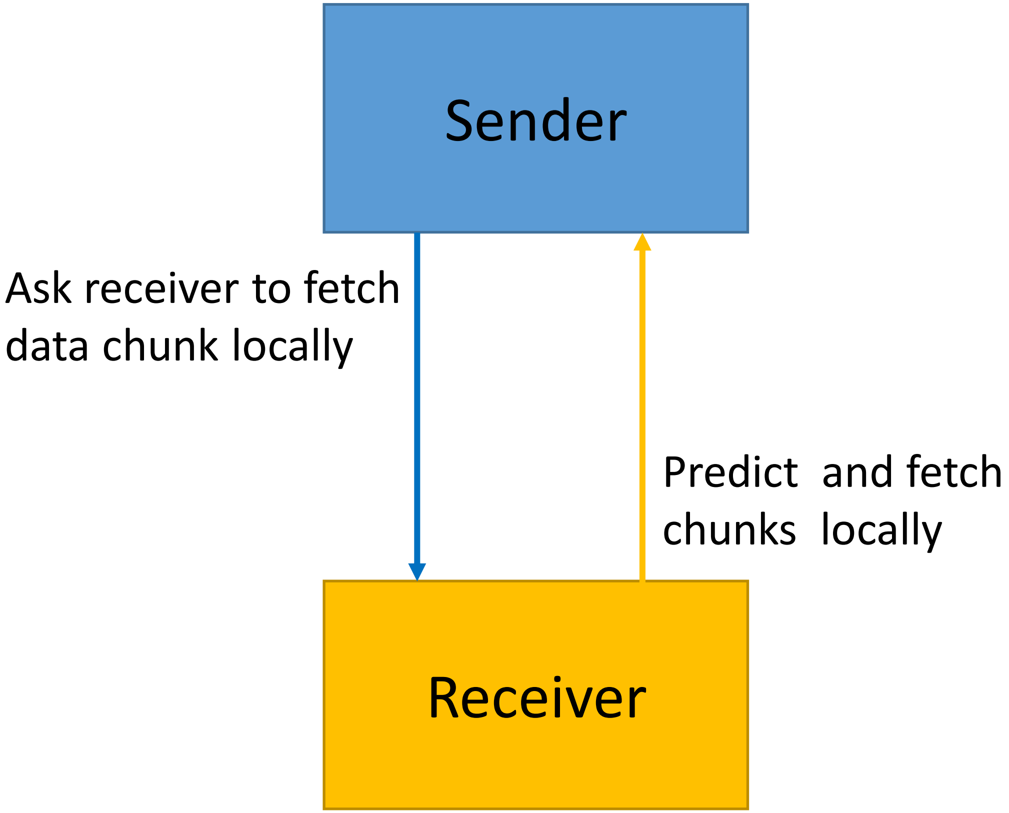
- Data transmission between nodes (edge, fog and cloud nodes) generate high bandwidth overhead and delay

Rationale

- Data redundancy in the data stream

Strategy

- Redundancy elimination

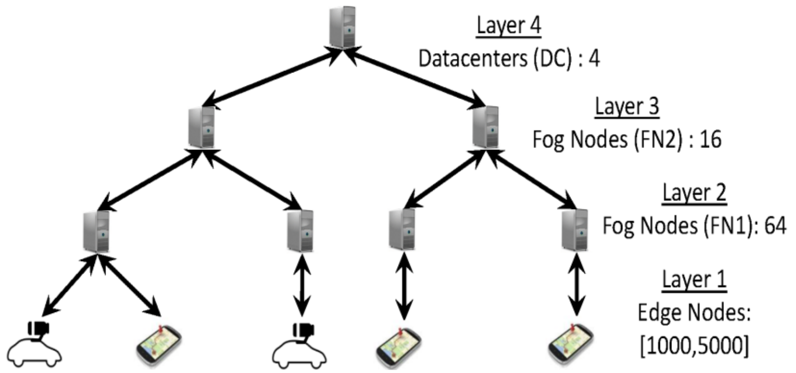


Experimental Setup

- Simulation on iFogSim: 5000 edge nodes

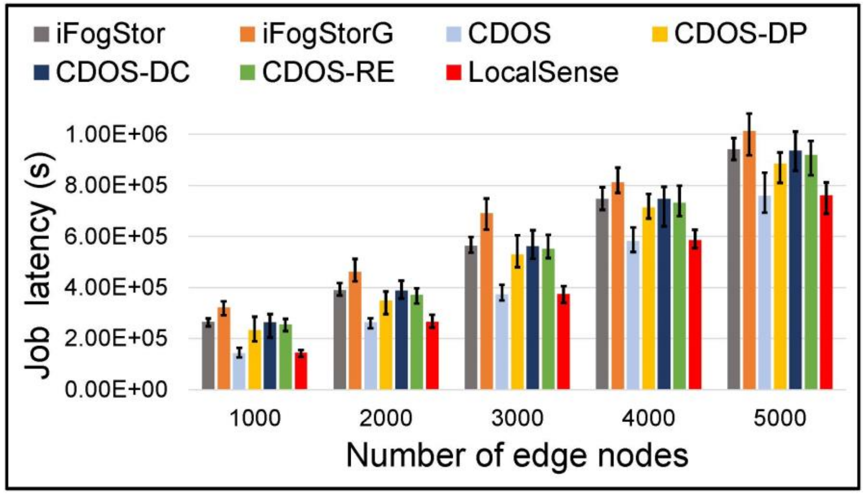
Table 1: Simulation parameters.

Edge node (EN)		Fog node (FN1 & FN2)	
Storage capacity	10MB- 200MB	Storage capacity	150MB- 1GB
Edge-FN1 network bandwidth	1 Mbps- 2 Mbps	FN1-FN2 network bandwidth	3 Mbps- 10 Mbps
Idle/Busy power	1/10 MW	Idle/Busy power	80/120 MW

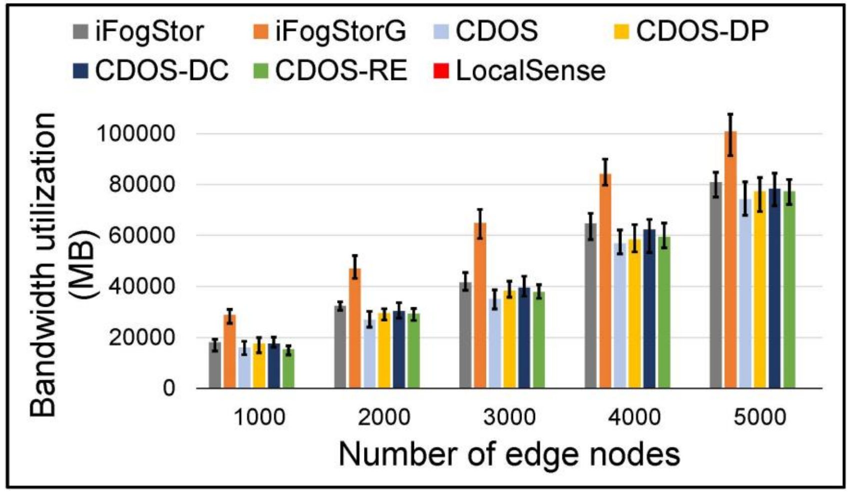


- Real device testbed
  - 5 Raspberry-Pi devices
- Compared methods
  - iFogStor (ICFEC'17)- finds data hosts that minimizes data transmission latency
  - iFogStorG (ASAC'18)- partitions the system to sub-graphs and finds the optimal data placement in each partition
  - LocalSense - each edge node senses all of its needed source data

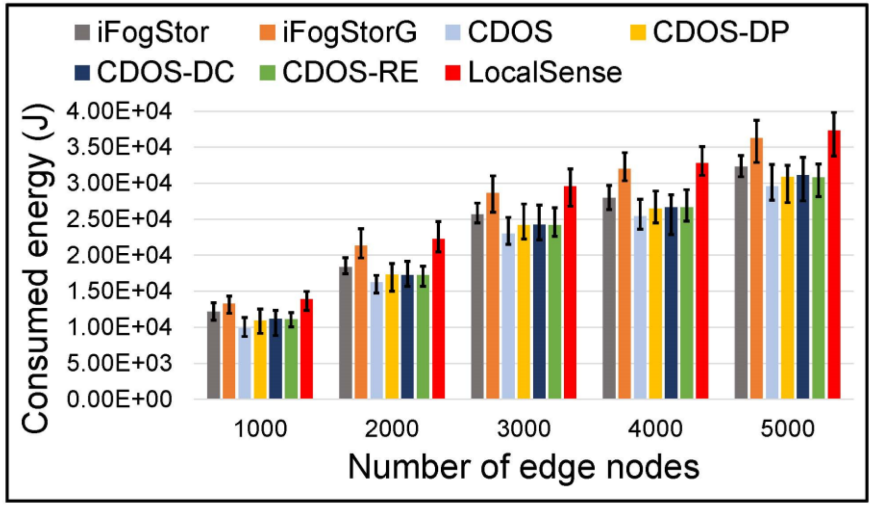
Experimental Results (cont.)



23%-55% improvement



21%-46% improvement

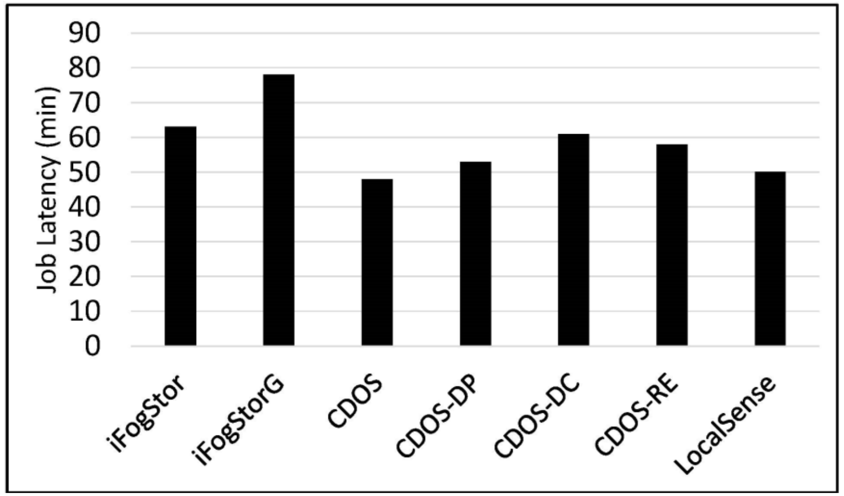


18%-29% improvement

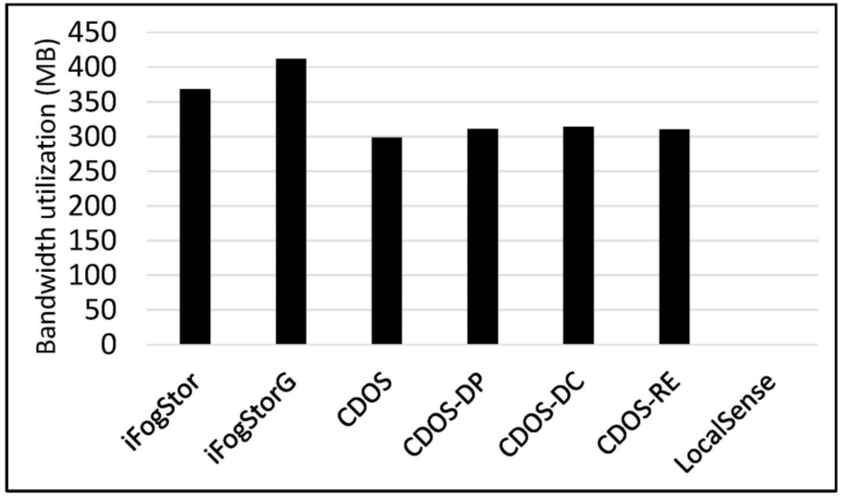
INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

acm In-Cooperation sighpc

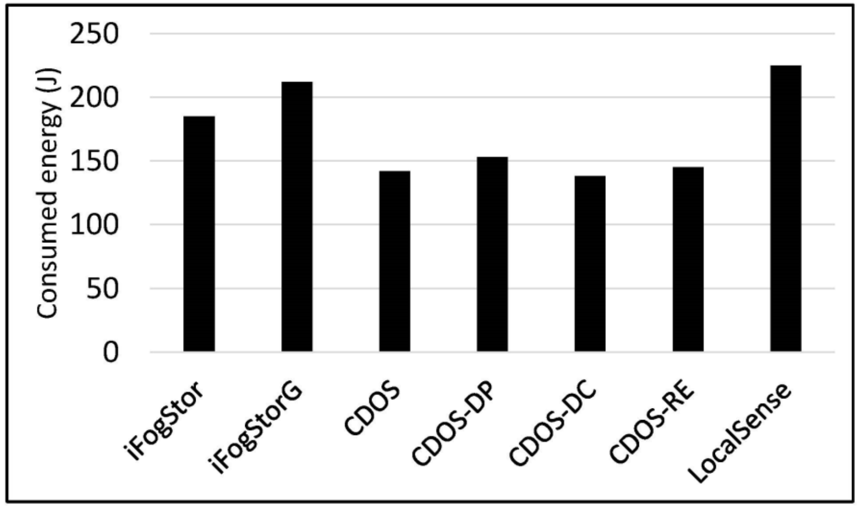
Experimental Results



26% improvement



29% improvement



21% improvement

Conclusion

Edge Computing → ← ML/AI

- Motivation: Reduce communication latency, job latency, power consumption and bandwidth consumption for AI jobs on the edge
- Approach: **<u>C</u>ontext-aware <u>D</u>ata <u>O</u>peration <u>S</u>ystem (CDOS)**
  - Data sharing and placement
  - Data collection
  - Redundancy elimination
- Future work: jointly consider job scheduling and data operations

Let ICAs assistant you!

Thank you!

Questions & Comments?

Tanmoy Sen

ts5xm@virginia.edu

University of Virginia