一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
MoNA2017-2(2017-06)

エッジ-クラウド連携制御のためのシステム設計の研究

新熊　亮一[†]　加藤　槙悟[††]　上林　将大[††]　池田　泰弘[†††]　川原　亮一[†††]
林　孝典[††††]

† 京都大学大学院情報学研究科 京都市左京区吉田本町
†† 京都大学大学院情報学研究科 京都市左京区吉田本町
††† NTT ネットワーク基盤技術研究所 東京都武蔵野市緑町 3-9-11
†††† 広島工業大学情報学部
E-mail: †shinkuma@i.kyoto-u.ac.jp, ††{skato,mkanbayashi}@icn.cce.i.kyoto-u.ac.jp,
†††{ikeda.yasuhiro,kawahara.ryoichi}@lab.ntt.co.jp, ††††t.hayashi.xk@it-hiroshima.ac.jp

System design for cooperative edge-cloud computing

Ryoichi SHINKUMA[†], Shingo KATO[††], Masahiro KANBAYASHI[††], Yasuhiro IKEDA[†††], Ryoichi KAWAHARA[†††], and Takanori HAYASHI[††††]

† Graduate School of Informatics, Kyoto University Yoshida-honmachi Sakyo-ku, Kyoto, Japan
†† Graduate School of Informatics, Kyoto University Yoshida-honmachi Sakyo-ku, Kyoto, Japan
††† NTT Network Technology Laboratories 3-9-11 Midori-cho,Musashino-shi, Tokyo, Japan
†††† Faculty of Applied Information Science, Hiroshima Institute of Technology
E-mail: †shinkuma@i.kyoto-u.ac.jp, ††{skato,mkanbayashi}@icn.cce.i.kyoto-u.ac.jp,
†††{ikeda.yasuhiro,kawahara.ryoichi}@lab.ntt.co.jp, ††††t.hayashi.xk@it-hiroshima.ac.jp

Abstract Road traffic congestion is still a serious problem in many countries, creating huge economic and environmental impacts. Delivering fine-grained information on road traffic conditions to vehicles is a straightforward solution to the congestion problem. However, researchers have recently pointed out that a central cloud is problematic for realtime information delivery to drivers because of the non-negligible latency between vehicles and central cloud servers, which is caused by the network distance and communication traffic load between them. This report therefore presents a novel system architecture for predictive road-traffic information delivery in which computing resources at the network edge and the central cloud are cooperatively used to analyze sensing data collected by vehicles on the road. In this report, we also present the mathematical problem formulation of the proposed system architecture for ensuring that the system could successfully deliver road-traffic information at realtime without overflowed computational and network loads. The numerical examination using a real dataset and a realistic network emulator validates our system.
(This work is under review by an IEEE conference. This report has been published without review process.)

Key words cloud-edge interoperation, road-traffic information delivery, prediction using machine learning

Background

- Road traffic congestion causes economy-wide costs across UK, France, Germany, & USA
  - $200.7 billion in 2013
  - $293.1 billion by 2030
- Delivery of "predictive" road-traffic information to drivers (human or robotic)
  - Data collection: roadside cameras, VANET, mobile crowd sourcing (MCS)
  - Traffic prediction: machine learning by central cloud

Real-time delivery is infeasible because of latency in network between vehicles and central cloud

Proposed solution

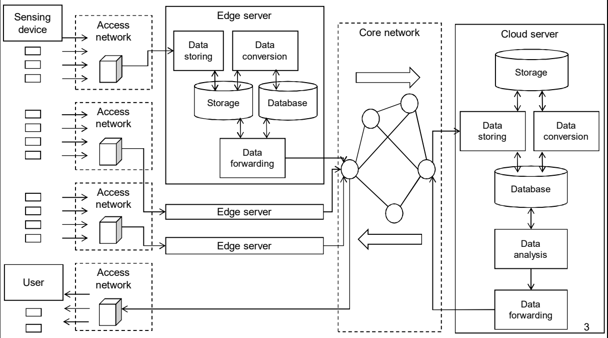
- Interoperation between two computational entities:
  1. Central cloud: long latency / generous computational resources
  2. Edge (or fog): short latency / limited computational resources

Goals

- Design of cloud-edge interoperation system for real-time delivery of predictive road-traffic information
- Problem formulation for ensuring its feasibility

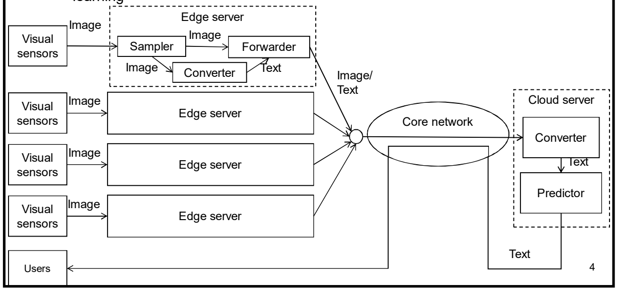
General model of proposed system

- Sensor devices, access networks, edge servers, core network, cloud server, and users



Model for problem formulation

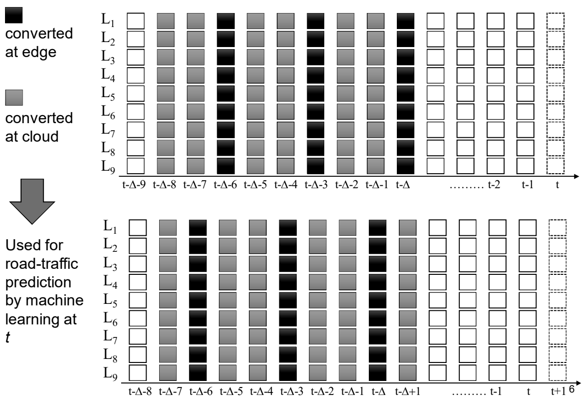
- Edge servers:
  - convert images received from sensors to structured text data **at sampling rate R_e**
  - forward unconverted images to cloud server **at sampling rate R_c**
- Cloud server:
  - converts images received from edge to structured text data
  - uses structured text data **at sampling rate $R = R_e + R_c$** for prediction by machine learning



Parameter definitions

R_e	Sampling rate at edge servers
P_e	Image processing speed in no. of images per slot at edge servers
N_e	Total no. of edge servers
N_L	No. of locations of visual sensors belonging to each edge server
R_c	Sampling rate at cloud server
P_c	Image processing speed in no. of images per slot at cloud server
θ	Transferring throughput of core network in no. of images per slot
R	Data rate used for prediction ($= R_e + R_c$)
t_p	Time consumed for prediction in slot
Δ	Backward parameter for prediction
A_p	Prediction accuracy
A_p'	Required prediction accuracy

Example of time-slot sequence



Problem formulation

- Edge processing should not overflow

$$P_e \geq N_L R_e, \qquad (1)$$

- Cloud processing and core-network throughput should not become bottleneck

$$\min(P_c, \theta) \geq N_L N_e R_c, \qquad (2)$$

- Δ in previous slide should be determined so that processing and forwarding data are completed before *t*

$$\Delta \geq \max(N_L R_e/P_e, 1/\theta + N_L N_e R_c/P_c, N_L N_e R_c/\theta + 1/P_c) + t_p, \qquad (3)$$

- Δ should be minimized for accurate prediction; as Δ increases, past data with lower time-correlation is used for prediction

$$\min_{R_e, R_c} \Delta \qquad (4)$$

$$\text{s.t.} A_p > A'_p,$$

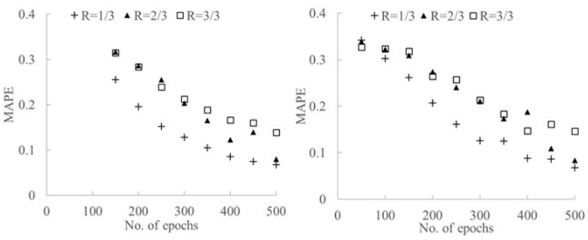
Evaluation of prediction accuracy, A_p

- Road-traffic dataset [10]
  - Portland-Vancouver Metropolitan region
  - 210 locations
  - Jan 1st to 2nd, 2016
- Machine learning method
  - Deep neural network (DNN) [11,12]

$R = R_e + R_c$	1/3	2/3	3/3
Structure	input layer, hidden layer × 3, output layer		
Input units	630	1260	1890
Hidden units	945	1890	2835
Output units	210		
Activate function	ReLU function		
Loss function	MSE		
Optimizer	Adam		
Batch size	100		

Prediction accuracy results (1)

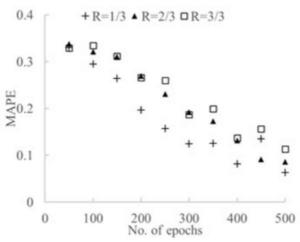
- As no. of epochs in learning process of DNN increases, prediction accuracy is improved



(a) Δ = 2 (b) Δ = 3

Prediction accuracy results (2)

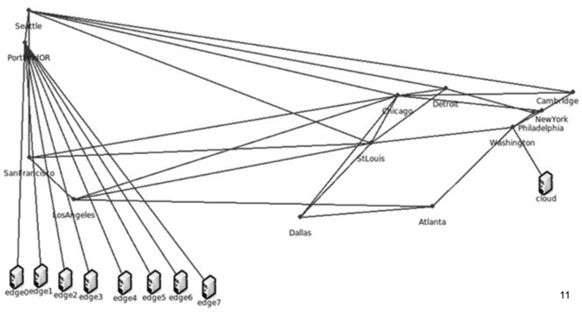
- Prediction accuracy is not sensitive to *R* and Δ



(c) Δ = 4

Evaluation of throughput, θ

- Emulator: Common Open Research Emulator (CORE) [13]
- Network topology: Rocketfuel dataset [14]

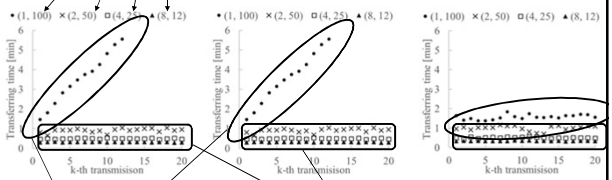


Parameters for network evaluation

Network	
Topology	Rocketfuel dataset AS7018 (AT&T) [14]
Location of cloud server	Washington, DC
Locations of edge servers	Portland, OR
Max. no. of hops between edge and cloud servers	5
Min. no. of hops between edge and cloud servers	3
No. of nodes in core network	13
No. of links in core network	27
Latency between edge servers and access gateway of core network	30ms
Background traffic	
Arrival distribution	Exponential ($\lambda = 0.2$)
Connection-time distribution	Lognormal ($\mu = 2.0$, $\sigma = 0.5$)
Transferred files	
Transferred file size	60MB
Data arrival rate	1/3, 2/3, or 3/3
Total no. of locations	100, 200, or 400
Total no. of edge servers	1, 2, 4, or 8
File transfer protocol	SCP (Secure Copy)
Environment	
OS	Ubuntu 14.04 64bit
CPU	2.40 GHz × 12
Memory	94.4 GiB

Network latency results (1)

(No. of edge servers, No. of locations per edge server)



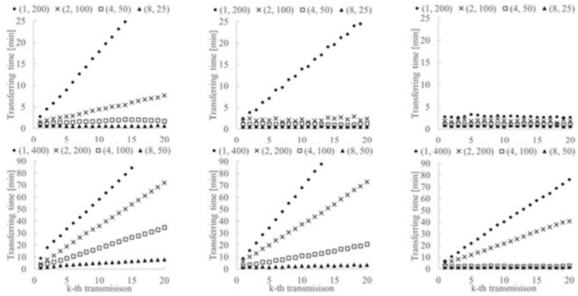
(a) R_c = 3/3 (b) R_c = 2/3 (c) R_c = 1/3

Transferring time keeps increasing:
Network overflowed

Transferring time is stable
Network not overflowed

Network latency results (2)

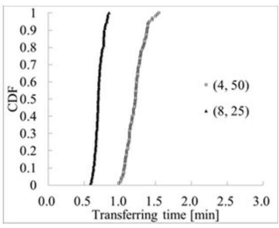
- As no. of locations increases, overflowed easily
- R_c should be set minimum (=1/3) to suppress increase of latency

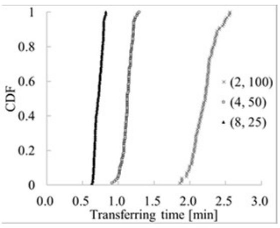


(a) R_c = 3/3 (b) R_c = 2/3 (c) R_c = 1/3

Distribution of network latencies

- Only `not-overflowed' cases are plotted
- Latencies are within three minutes at longest





Discussion of optimal sampling rate setting

$$\Delta \geq \max(\overbrace{N_L R_e/P_e}^{(a)}, 1/\theta + \overbrace{N_L N_e R_c/P_c}^{(b)}, \underbrace{N_L N_e R_c/\theta}_{(c)} + 1/P_c) + \underbrace{t_p}_{(d)}, \qquad (3)$$

$$\min_{R_e, R_c} \Delta \qquad (4)$$

$$\text{s.t.} \underbrace{A_p > A'_p}_{(e)},$$

a. Processing for identifying vehicles from camera image [4] consumes time even using multiple processers; R_e should be set to 1/3.
b. Central cloud has generous computational resources; R_c can be 3/3.
c. Time for transferring image data easily increases because of overloaded data traffic on core network; R_c should be set to 1/3.
d. Time for prediction is ignorable as long as learning process has been completed in advance.
e. Prediction accuracy is not sensitive to R and Δ.

R should be set to 1/3: (R_c=1/3,R_e=0) or (R_c=0,R_e=1/3)

Conclusion & Future work

- Background: road traffic congestion cause huge economic & environmental impacts
- Solution: realtime delivery of predictive road-traffic information
- Proposed: cloud-edge interoperation system
- Results:
  1. Problem formulation for ensuring system feasibility
  2. Numerical results of prediction accuracy using DNN
  3. Numerical results of network latency using emulator & real network topology
  4. Suggestion of optimal sampling rate setting at cloud &edge
- Future work:
  i. Evaluation using other datasets
  ii. System implementation & experiment

References

1. The future economic and environmental costs of gridlock in 2030, An assessment of the direct and indirect economic and environmental costs of idling in road traffic congestion to households in the UK, France, Germany and the USA Report for INRIX, Cebr, July 2014.
2. R. Yu, Y. Zhang, S. Gjessing, W. Xia, K. Yang, Toward cloud-based vehicular networks with efficient resource management, IEEE Network, 27(5), pp.48-55, Oct 2013.
3. J. Wan, J. Liu, Z. Shao, A. V. Vasilakos, M. Imran, K. Zhou, Mobile crowd sensing for traffic prediction in internet of vehicles, Sensors, 16(1), p.88, Jan 2016.
4. Y. Wen, Y. Lu, J. Yan, Z. Zhou, K. M. Deneen, P. Shi, An Algorithm for License Plate Recognition Applied to Intelligent Transportation System, IEEE Transactions on Intelligent Transportation Systems, Vol.12, No.3, Sept 2011.
5. A. Lakas, M. Shaqfa, Geocache: sharing and exchanging road traffic information using peer-to-peer vehicular communication, IEEE 73rd Vehicular Technology Conference (VTC Spring), pp. 1-7, May 2011.
6. Y. Lv, Y. Duan, W. Kang, Z. Li, F. Y. Wang, Traffic flow prediction with big data: a deep learning approach, IEEE Transactions on Intelligent Transportation Systems, 16(2), pp.865-873, April 2015.
7. K. Sasaki, N. Suzuki, S. Makido, A. Nakao, Vehicle control system coordinated between cloud and mobile edge computing, 55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Sept. 2016.
8. R. Deng, R. Lu, C. Lai, T. H. Luan, Towards power consumption-delay tradeoff by workload allocation in cloud-fog computing. IEEE International Conference on Communications (ICC), pp. 3909-3914, Jun 2015.
9. M. Aazam, E. N. Huh, Fog computing and smart gateway based communication for cloud of things, IEEE International Conference on Future Internet of Things and Cloud (FiCloud), pp.464-470, Aug 2014.
10. PORTAL, https://portal.its.pdx.edu/home.
11. M. W. Gardner, S. R. Dorling, Artificial neural networks (the multilayer perceptron) - a review of applications in the atmospheric sciences. Atmospheric environment, 32(14), pp.2627-2636, Aug 1998.
12. Chainer MNIST example, https://github.com/pfnet/chainer/ blob/master/examples/mnist/train_mnist.py
13. J. Ahrenholz, C. Danilov, T. R. Henderson, J. H. Kim, CORE: A real-time network emulator, IEEE Military Communications Conference (MILCOM), pp.1-7, Nov 2008.
14. N. Spring, M. Ratul, and T. Anderson, The causes of path inflation, ASIGCOMM '03 Proc. the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, Aug 2003.