Computing-Aware Traffic Steering Working Group
Internet-Draft
Intended status: Informational
Expires: 29 May 2026
25 November 2025

J. Jeong, Ed.
B. Mugabarigira
Sungkyunkwan University

Applicability of Computing-Aware Traffic Steering to Intelligent Transportation Systems

draft-jeong-cats-its-use-cases-05

Abstract

This document describes the applicability of Computing-Aware Traffic Steering (CATS) to Intelligent Transportation Systems (ITS). CATS provides the steering of packets of a traffic flow for a specific service request toward the corresponding service instance at an edge computing server at a service site. CATS are applicable for Computing-Aware ITS including (i) Context-Aware Navigation Protocol (CNP) for Terrestrial Vehicles and Unmanned Aerial Vehicles (UAV), (ii) Edge-Assisted Cluster-Based MAC Protocol (ECMAC) for Software-Defined Vehicles, and (iii) Self-Adaptive Interactive Navigation Tool (SAINT) for Cloud-Based Navigation Services, and (iv) Cloud-Based Drone Navigation (CBDN) for Efficient Drone Battery Charging.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Nowadays, various networked services are provided by leveraging edge computing infrastructure. Either a closest or a lightest edge computing server (simply called an edge server) can be selected to serve a request service. In this trend, Computing-Aware Traffic Steering (CATS) is standardized to provide the steering of packets of a traffic flow for a specific service request toward the corresponding service instance at an edge server at a service site [I-D.ietf-cats-usecases-requirements][I-D.ietf-cats-framework].

This document proposes four use cases about the applicability of CATS for Computing-Aware Intelligent Transportation Systems (ITS). They are (i) Context-Aware Navigation Protocol for Terrestrial Vehicles and Unmanned Aerial Vehicles (UAV) [CNP-Vehicle] [CNP-UAV], (ii) Edge-Assisted Cluster-Based MAC Protocol for Software-Defined

Vehicles (SDV) [ECMAC], (iii) Self-Adaptive Interactive Navigation Tool (SAINT) for Cloud-Based Navigation Services [SAINT], and (iv) Cloud-Based Drone Navigation (CBDN) for Efficient Drone Battery Charging [CBDN].

2. Terminology

This document uses the terminology described in [I-D.ietf-cats-usecases-requirements] and [I-D.ietf-cats-framework]. In addition, the following terms are defined below:

* Context-Aware Navigation Protocol (CNP): It is an application protocol that enables either terrestrial vehicles (i.e., ground vehicles) or Unmanned Aerial Vehicles (UAV) to move in road networks or fly in the sky to maneuver safely without collisions, respectively [CNP-Vehicle][CNP-UAV].

* Edge-Assisted Cluster-Based MAC Protocol (ECMAC): It is a Media Access Control (MAC) protocol that enables Software-Defined Vehicles (SDV) to communicate with each other using Software-Defined Vehicular Networks with edge computing servers [ECMAC].

* Self-Adaptive Interactive Navigation Tool (SAINT): It is an application protocol that guides terrestrial vehicles to navigate efficiently towards their destination through the interaction between the vehicles and the vehicular cloud for navigation services [SAINT].

* Cloud-Based Drone Navigation (CBDN): It is an application protocol for efficient drone battery charging in drone networks by finding globally coordinated drone routes that minimize the total traffic delay in a drone network while reducing the overall Quick Battery-Charging Machine (QCM) congestion level [CBDN].

3. Vehicular Network Architecture

This section explains a vehicular network architecture for vehicles in Computing-Aware ITS.

Software-Defined Vehicles (SDV) include terrestrial vehicles and Unmanned Aerial Vehicles (UAV). The standardization and implementation of SDVs are performed by AUTOSAR [AUTOSAR], Eclipes SDV [Eclipse-SDV], and COVESA [COVESA]. These SDVs need to communicate with each other to avoid collisions or accidents.

Figure 1 shows a Vehicular Network Architecture for Software-Defined Vehicles (SDV) such as terrestrial vehicles and Unmanned Aerial Vehicles (UAV). This vehicular network architecture is based on the vehicular network architecture for IPv6 Wireless Access in Vehicular Environments (IPWAVE) in [RFC9365].

```
                              Vehicular Cloud
               *******************************************
             *                                             *
            *              +------------------+             *
           *               | Cloud Controller |              *
           *               +------------------+              *
           *                         ^                       *
            *                        |                      *
             *                       v                     *
               *******************************************
                 ^   Edge-Cloud1    ^   Edge-Cloud2    ^   Edge-Cloud3
                 | +------------+   | +------------+   | +------------+
                 | |Edge-Server1|   | |Edge-Server2|   | |Edge-Server3|
                 | +------------+   | +------------+   | +------------+
                 |   ^              |   ^              |   ^
                 |   |              |   |              |   |
                 v   V              v   V              v   V
               +---------+         +---------+        +---------+
               |   gNB1  |<------->|   gNB2  |<------>|   gNB3  |
               +---------+         +---------+        +---------+
                    ^                   ^                    ^
                    :                   :                    :
           +-----------------+ +-----------------+   +-----------------+
           |        : V2I    | |        : V2I    |   |       : V2I     |
           |        v        | |        v        |   |       v         |
+--------+ |   +--------+    | |   +--------+    |   |   +--------+    |
|  SDV1  |===> |  SDV2  |===>| |   |  SDV3  |===>|   |   |  SDV4  |===>|
+--------+<...>+--------+<........>+--------+    |   |   +--------+    |
           V2V     ^         V2V        ^        |   |        ^        |
           |       : V2V     | |        : V2V    |   |        : V2V    |
           |       v         | |        v        |   |        v        |
           |  +--------+     | |   +--------+    |   |    +--------+   |
           |  |  SDV5  |===> | |   |  SDV6  |===>|   |    |  SDV7  |==>|
           |  +--------+     | |   +--------+    |   |    +--------+   |
           +-----------------+ +-----------------+   +-----------------+
                 Subnet1              Subnet2              Subnet3
                (Prefix1)            (Prefix2)            (Prefix3)

        <----> Wired Link   <....> Wireless Link   ===> Moving Direction
```

Figure 1: Vehicular Network Architecture for Software-Defined Vehicles

4. Use Cases

This section explains four use cases about the applicability of CATS to Computing-Aware ITS.

4.1. Context-Aware Navigation Protocol

A connected network of automated vehicles on roads can increase the driving safety of driverless vehicles (i.e., autonomous vehicles). The critical level of dangerous situations on the road while driving can be increased by the speed, orientation, and traffic density of the vehicles involved. Therefore, there is a need for a maneuvering mechanism that handles both the current driving vehicle and the oncoming vehicles headed toward an emergency zone (e.g., road hazard and road accident spot).

```
                    Edge Cloud
                  +-------------+
                  | Edge-Server1|
                  +-------------+
                    ^         ^
                    |         |
                    V         V
            +---------+      +---------+           +---------+
            |   gNB1  |<---->|   gNB2  |<---...--->|   gNBn  |
            +---------+      +---------+           +---------+
                 ^              ^                        ^
                 :              :                        :
    +-------------------+ +----------+     +-----------------------------+
    |            : V2I  | |    : V2I |     |            : V2I            |
    |      V2V   v      | |    v     |     |            v                |
    |+----+<-->+----+   | | +----+   |     |+----+V2V +----+ V2V+----+   |
    || CM1|==> | CH1|==>| | | CH2|==>|     || CM1|<-->| CHn|<-->| CM2|==>|
    |+----+    +----+   | | +----+   |     |+----+==> +----+==> +----+   |
    |            ^      | |    ^     | ... |             ^               |
    |            | V2V  | |    | V2V |     |             | V2V           |
    |            v      | |    v     |     |             v               |
    |          +----+   | | +----+   |     |          +----+             |
    |          | CM2|==>| | | CM1|==>|     |          | CM3|==>          |
    |          +----+   | | +----+   |     |          +----+             |
    +-------------------+ +----------+     +-----------------------------+
           Cluster1         Cluster2                  Clustern

    <----> Wired Link   <....> Wireless Link   ===> Moving Direction
```

Figure 2: The Illustration of Context-Aware Navigator Protocol

Context-Aware Navigation Protocol (CNP) enhances the safety of vehicles driving in urban roads [CNP-Vehicle][CNP-UAV]. Firstly, CNP includes a collision avoidance module that builds on both vehicular networks and on-board sensors to track vehicles’ behaviors, and this module analyzes the driving risks to determine the necessary maneuvers in dangerous situations. Secondly, CNP establishes a collision mitigation strategy that limits the severity of collision damages in hazardous road during non-maneuverable scenarios. Through a theoretical analysis as well as extensive simulations, the effectiveness of CNP is shown in terms of the reduction of both communication overhead and the risk of road collisions.

To use CNP, vehicles need to report their mobility information (e.g., vehicle identifier, destination, current position, direction, and speed) to a central cloud or an edge cloud for a CNP-based vehicle collision avoidance service as shown in Figure 2. Service instances at either the edge cloud or the central cloud need to work for the vehicles. The packets with the mobility information per vehicle need to steered to an appropriate service instance for CNP. The service instance needs to provide a appropriate maneuver direction to each vehicle moving on the roadway.

Since the vehicle is moving along the roadway, to serve the vehicle for collision avoidance, a new service instance needs to be selected for it, considering the network delay between the vehicle and service instance and also computing resources for the service instance. For the service instances to continue to compute the maneuvers smoothly, they need to exchange the mobility information as context while the vehicles are moving and change their service instance over time. That is, the context migration should be supported in the CATS infrastructure having the central clouds and the edge clouds to foster service instances.

4.2. Edge-Assisted Cluster-Based MAC Protocol

Vehicular networks have emerged as a promising means to mitigate safety hazards in modern transportation systems. On highways, emergency situations associated with vehicles necessitate a reliable Media Access Control (MAC) protocol that can provide timely warnings of possible vehicle collisions.

An Edge-Assisted Cluster-Based MAC Protocol (ECMAC) is a vehicular MAC protocol for reliable and fast packet dissemination in software-defined vehicular networks [ECMAC]. To reduce the control messaging overhead for clustering, ECMAC separates the cluster control plane (i.e., managing cluster formation) from the data plane (i.e., actual data transmission and forwarding) by using a software-defined network controller in a cellular network edge server as illustrated in Figure 3.

For transmitting packets effectively and efficiently, ECMAC tries to channel interference minimization among adjacent clusters by using a joint optimization of channel assignment and a time slot scheduling. The joint optimization consists of two phases such as the channel assignment phase and the time slot allocation phase. In the first phase for the channel assignment, ECMAC allocates different wireless channels to the adjacent channels by minimizing the total inter-cluster interference by reusing the available channels. In the second phase for the time slot allocation, ECMAC uses a time-division multiple access (TDMA) schedule algorithm to guarantee a high reliability and a low latency. The TDMA schedule in ECMAC is determined by a joint optimization process in the cellular edge, which is formulated as a binary integer linear programming problem and solved by a heuristic approach based on the divide-and-conquer paradigm. This joint optimization process minimizes the signal interference by jointly considering channel assignment and time slot allocation, thereby ensuring reliable communication. Through extensive simulations, the effectiveness of ECMAC is demonstrated a higher delivery ratio of emergency packets than the legacy data delivery approaches.

```
                      Vehicular Cloud
              *******************************
            *            Edge Cloud           *
           *    +--------------------------+    *
          *     |       Edge-Server1       |     *
          *     |  +--------------------+  |     *
          *     |  |Cluster Formation,  |  |     *
          *     |  |Channel Assignment, |  |     *
          *     |  |Time Slot Allocation|  |     *
           *    |  +--------------------+  |    *
            *   +--------------------------+   *
              ********************************
                     ^                       ^
                     |                       |
                     V                       V
               +---------+               +---------+
               |   gNB1  |<-----...----->|   gNBm  |
               +---------+               +---------+
                    ^                         ^
                    :                         :
       +-------------------+     +-----------------------------+
       |            : V2I  |     |            : V2I            |
       |            v      |     |            v                |
       |+----+V2V +----+   |     |+----+ V2V+----+V2V +----+   |
       || CM1|<-->| CH1|==>|     || CM1|<-->| CHn|<-->| CM2|==>|
       |+----+==> +----+   |     |+----+==> +----+==> +----+   |
       |            ^      | ... |             ^               |
       |            | V2V  |     |             | V2V           |
       |            v      |     |             v               |
       |          +----+   |     |          +----+             |
       |          | CM2|==>|     |          | CM3|==>          |
       |          +----+   |     |          +----+             |
       +-------------------+     +-----------------------------+
              Cluster1                      Clustern

       <----> Wired Link   <....> Wireless Link   ===> Moving Direction
```

Figure 3: The Illustration of Edge-Assisted Clusterer-Based MAC Protocol

In ECMAC, the cellular network edge server can be implemented as a service instance in the CATS infrastructure. In the same way with CNP, service instances need to efficiently perform the context migration (e.g., mobility information and cluster membership) of vehicles so that they can continue to form clusters of vehicles, allocate wireless channels to the vehicles, and assign time slots to the vehicles over time.

4.3. Self-Adaptive Interactive Navigation Tool for Cloud-Based Navigation

Efficient navigation services are important in Intelligent Transportation Systems because they allow vehicles to move towards destinations quickly. For this efficient navigation, vehicles need to interact with a central cloud or an edge cloud in real time.

Self-Adaptive Interactive Navigation Tool (SAINT) is a cloud-based navigation guidance system for vehicular traffic optimization in road networks [SAINT]. The legacy navigation systems guide vehicles to take their navigation paths with real-time traffic statistics in road maps without considering the navigation paths of other vehicles. This uncoordinated navigation planning may incur traffic congestion in certain areas in the road networks.

```
                         Vehicular Cloud
              *************************************
            *              Edge Cloud               *
           *  +-----------------------------------+  *
          *   |   Edge Server    Edge Emergency   |   *
          *   |                       Center      |   *
          *   | +-------------+ +---------------+ |   *
          *   | |Road Traffic | |Road Emergency | |   *
          *   | |Information  | |Notification   | |   *
           *  | +-------------+ +---------------+ |  *
            * +-----------------------------------+ *
              *************************************
                ^                  ^              ^
                |                  |              |
                V                  V              V
           +---------+       +---------+        +---------+
           |   gNB1  |<----->|  gNB2   |<------>|  gNB3   |
           +---------+       +---------+        +---------+
                ^                   ^                   ^
                :                   :                   :
                :                   :                   :
          +-----------------+ +-----------------+ +-----------------+
          |     : V2I       | |     : V2I       | |     : V2I       |
          |     v           | |     v           | |     v           |
          |   +--------+    | |   +--------+    | |   +--------+    |
          |   |  SDV1  |===>| |   |  SDV2  |===>| |   |  SDV3  |===>|
          |   +--------+<........>+--------+    | |   +--------+    |
          |      ^        V2V        ^          | |        ^        |
          |      : V2V      | |       : V2V     | |        : V2V    |
          |      v          | |       v         | |        v        |
          |  +--------+     | |   +--------+    | |    +--------+   |
          |  |  SDV4  |===> | |   |  SDV5  |===>| |    |  SDV6  |==>|
          |  +--------+     | |   +--------+    | |    +--------+   |
          +-----------------+ +-----------------+ +-----------------+
                Subnet1             Subnet2             Subnet3
               (Prefix1)           (Prefix2)           (Prefix3)

       <----> Wired Link   <....> Wireless Link   ===> Moving Direction
```

     Figure 4: The Illustration of Self-Adaptive Interactive Navigation

On the other hand, SAINT uses a virtual metric called congestion contribution that estimates traffic congestion in each road segment in the current time and near-future time by considering the planned navigation paths of the vehices in the target road network. SAINT guides each vehicle to have a certain-level detour in order to make the whole road network have spread vehicular traffic and lessen possible traffic congestion in certain road segments or intersections.

For this cooperative navigation in SAINT, while vehicles are moving along the roadways, they need to send their periodic navigation queries and their mobility information to appropriate service instances in a central cloud or an edge cloud in the CATS infrastructure. The service instances need to process their navigation queries and reply to them with good navigation paths, considersing the road-wide traffic optimization as depicted in Figure 4. Due to the movement of the vehicles, the switching from a service instance to another service instance should be performed efficiently, considering the network delay between the service instance and each vehicle and the computing resources of the service instance.

SAINT can support the efficient delivery of emergency vehicles such as ambulance and fire engine to a road accident spot by the management of a congestion contribution matrix in a target road network [SAINTplus]. It can not only guide vehicles within the accident spot, but also can detour vehicles approaching the accident spot. This version of SAINT is called SAINT+.

4.4. Cloud-Based Drone Navigation (CBDN) for Efficient Battery Charging in Drone Networks

The growing popularity of Unmanned Aerial Vehicles (UAV) comes with a need to charge their battery at Quick Battery-Charging Machines (QCMs) due to their limited battery capacity. Without drone coordination, a drone’s choice for its QCM may lead to congestion resulting from multiple drones selecting the same QCM, thus increasing the drones’ battery-charging delay due to the queueing day at the QCM. This battery-charging delay leads to a long travel delay for each drone at the QCM. A Cloud-Based Drone Navigation (CBDN) efficiently determines drone routes to minimize the overall QCM congestion level for all QCMs in a target drone network [CBDN]. It finds globally coordinated drone routes that minimize the total travel delay in a drone network by reducing the overall QCM congestion level.

```
                        Vehicular Cloud
               *******************************
             *            Edge Cloud          *
            *    +--------------------------+    *
           *     |       Edge-Server1       |     *
           *     |  +--------------------+  |     *
           *     |  |Drone Registration, |  |     *
           *     |  |Drone Route Finding,|  |     *
           *     |  |QCM-Selection Scheme|  |     *
            *    |  +--------------------+  |    *
             *   +--------------------------+   *
               ********************************
                 ^            ^               ^
                 |            |               |
                 V            V               V
           +---------+     +---------+     +---------+
           |   gNB1  |<--->|   gNB2  |<--->|   gNB3  |
           +---------+     +---------+     +---------+
                 ^                ^                 ^
                 :                :                 :
       +-----------------+ +-----------------+ +-----------------+
       |         : V2I   | |      : V2I      | |    : V2I       |
       |         v       | |      v          | |    v           |
       |   +--------+    | |   +--------+    | |   +--------+    |
       |   |  UAV1  |===>| |   |  UAV2  |===>| |   |  UAV3  |===>|
       |   +--------+<........>+--------+    | |   +--------+    |
       |      ^         V2V       ^         | |        ^        |
       |      : V2V     | |       : V2I    | |        : V2I    |
       |      v         | |       v        | |        v        |
       |  +--------+     | |   +--------+    | |   +--------+    |
       |  |  UAV4  |===> | |   |  QCM1  |    | |   |  QCM2  |    |
       |  +--------+     | |   +--------+    | |   +--------+    |
       +-----------------+ +-----------------+ +-----------------+
             Subnet1             Subnet2             Subnet3
            (Prefix1)           (Prefix2)           (Prefix3)

       <----> Wired Link   <....> Wireless Link   ===> Moving Direction
```

Figure 5: The Illustration of Cloud-Based Drone Navigation

An edge cloud in the CATS infrastructure with computing and storage resources need to compute the trajectories of the drones (i.e., drone routes), along with their average speeds, source positions, and destination positions, as well as the battery charing loads at the QCMs. The wireless communications between drones and infrastrucure nodes (e.g., edge server) can be either 5G and beyond 5G or wireless LAN, as illustated in Figure 5. Drones interact with the edge server to compute navigation paths regarding the drone network-wide traffic optimization of all drones in the drone network. To decrease battery consumption, the drones only once report their mobility information (i.e., current position, destination, direction, and speed) to the edge computing decice to acquire their navigation paths.

Upon the commencement of the drone service, each drone reports its mobility information to the edge server. A drone’s QCM reservation for battery charging acquires the most efficient shortest path regarding the drone-network-wide traffic optimization of all the drones in the drone network. For this drone-network-wide traffic optimization, a drone sends its mobility information to the edge server before its departure, and the edge server computes an optimal navigation path to the drone and notifies the drone of the path in run time.

5. Requirements

This section specifies the requirements for the applicability of CATS to ITS use cases in Section 4.

* R1: Dynamic mapping between a required service and a service instance. Both network delay and computing delay are considered over time.

* R2: Run-time context migration of vehicles between edge servers (i.e., service instances). Each vehicle’s context (e.g., mobility information, communications parameters (e.g., channel, time slot)) is transferred to an appropriate service instance along with its movement over time.

* R3: Proactive load balancing among service instances considering the required Quality of Service (QoS) and Quality of Experience (QoE) for vehicles. The trajectories of vehicles are considered for such load balancing.

* R4: Dynamic clustering of geographically adjacent vehicles. Clusters of vehicles are dynamically reconstructed over time.

* R5: Dynamic network configuration for vehicles and network forwarding entities (e.g., base stations and switches/routers). In wireless networks, network resources (e.g., channel and time slot) per vehicle are dynamically configured by base stations. In wired networks, a network slice from a base station to a service instance are dynamically adjusted for each vehicle.

* R6: Differentiated packet scheduling for service types. Packets of real-time services (e.g., autonomous driving) and packets of non-real-time services (e.g., infotainment) are handled differently.

6. IANA Considerations

This document does not require any IANA actions.

7. Security Considerations

The same security considerations for Computing-Aware Traffic Steering (CATS) are applicable to the use cases for the Computing-Aware ITS [I-D.ietf-cats-usecases-requirements] [I-D.ietf-cats-framework].

8. References

8.1. Normative References

[RFC9365] Jeong, J., Ed., "IPv6 Wireless Access in Vehicular Environments (IPWAVE): Problem Statement and Use Cases", RFC 9365, DOI 10.17487/RFC9365, March 2023, <https://www.rfc-editor.org/info/rfc9365>.

8.2. Informative References

[I-D.ietf-cats-usecases-requirements]
Yao, K., Contreras, L. M., Shi, H., Zhang, S., and Q. An, "Computing-Aware Traffic Steering (CATS) Problem Statement, Use Cases, and Requirements", Work in Progress, Internet-Draft, draft-ietf-cats-usecases-requirements-09, 19 November 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-cats-usecases-requirements-09>.

[I-D.ietf-cats-framework]
Li, C., Du, Z., Boucadair, M., Contreras, L. M., and J. Drake, "A Framework for Computing-Aware Traffic Steering (CATS)", Work in Progress, Internet-Draft, draft-ietf-cats-framework-19, 20 November 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-cats-framework-19>.

[AUTOSAR] "AUTOSAR Adaptive Platform", Available: https://www.autosar.org/standards/adaptive-platform, March 2024.

[Eclipse-SDV]
"Eclipse Software Defined Vehicle Working Group Charter", Available: https://www.eclipse.org/org/workinggroups/sdv-charter.php, March 2024.

[COVESA] "Connected Vehicle Systems Alliance", Available: https://covesa.global/, March 2024.

[CNP-Vehicle]
Mugabarigira, B., Shen, Y., Jeong, J., Oh, T., and H. Jeong, "Context-Aware Navigation Protocol for Safe Driving in Vehicular Cyber-Physical Systems", IEEE Transactions on Intelligent Transportation Systems, Volume 24, Issue 1, Available: https://ieeexplore.ieee.org/document/9921182, January 2023.

[CNP-UAV] Mugabarigira, B. and J. Jeong, "Context-Aware Navigation Protocol for Safe Flying of Unmanned Aerial Vehicles", KICS Winter Conference, Available: http://iotlab.skku.edu/publications/international-journal/CNP-TITS-2023.pdf, January 2024.

[ECMAC] Shen, Y., Jeong, J., Jun, J., Oh, T., and Y. Baek, "ECMAC: Edge-Assisted Cluster-Based MAC Protocol in Software-Defined Vehicular Networks", IEEE Transactions on Vehicular Technology, Volume 73, Issue 9, Available: https://ieeexplore.ieee.org/document/10505005, September 2024.

[SAINT] Jeong, J., Jeong, H., Lee, E., Oh, T., and D. Du, "SAINT: Self-Adaptive Interactive Navigation Tool for Cloud-Based Vehicular Traffic Optimization", IEEE Transactions on Vehicular Technology, Volume 65, Issue 6, Available: https://ieeexplore.ieee.org/document/7243355, June 2016.

[SAINTplus]
Shen, Y., Lee, J., Jeong, H., Jeong, J., Lee, E., and D. Du, "SAINT+: Self-Adaptive Interactive Navigation Tool+ for Emergency Service Delivery Optimization", IEEE Transactions on Intelligent Transportation Systems, Volume 19, Issue 4, Available: https://ieeexplore.ieee.org/document/7953571, April 2018.

[CBDN] Kim, J., Kim, S., Jeong, J., Kim, H., Park, J., and T. Kim, "CBDN: Cloud-Based Drone Navigation for Efficient Battery Charging in Drone Networks", IEEE Transactions on

Intelligent Transportation Systems, Volume 20, Issue 11, Available: https://ieeexplore.ieee.org/document/8574043, November 2019.

Appendix A. Changes from draft-jeong-cats-its-use-cases-04

The following changes are made from draft-jeong-cats-its-use-cases-04:

* This version updates the figures in this draft by changing the IP-RSUs to gNBs to suite the 5G and beyong communication architecture.

Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea Ministry of Science and ICT (MSIT) (No. RS-2024-00398199 and RS-2022-II221015).

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government, Ministry of Science and ICT (MSIT) (No. 2023R1A2C2002990).

Contributors

This document is made by the group effort of CATS WG, greatly benefiting from inputs and texts by Peng Liu (China Mobile), Yong-Geun Hong (Daejeon University), and Joosang Youn (Dong-Eui University). The authors sincerely appreciate their contributions.

The following are coauthors of this document:

Juwon Hong
Department of Computer Science & Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon
Gyeonggi-Do
16419
Republic of Korea
Phone: +82 31 299 4106
Email: hongju2024@skku.edu
URI: http://iotlab.skku.edu/people-Joo-Won-Hong.php

Yiwen Shen
Department of Computer Science & Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon
Gyeonggi-Do
16419
Republic of Korea
Phone: +82 31 299 4106
Email: chrisshen@skku.edu
URI: https://chrisshen.github.io/

Authors' Addresses

Jaehoon Paul Jeong (editor)
Department of Computer Science & Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon
Gyeonggi-Do
16419
Republic of Korea
Phone: +82 31 299 4957
Email: pauljeong@skku.edu
URI: http://iotlab.skku.edu/people-jaehoon-jeong.php

Bien Aime Mugabarigira
Department of Electrical & Computer Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon
Gyeonggi-Do
16419
Republic of Korea
Phone: +82 10 5964 8794
Email: bienaime@skku.edu
URI: http://iotlab.skku.edu/people-Bien-Aime.php