Panorama: LLM-Guided Heuristics for Digital Twin-Empowered Vehicular Edge Computing

Ziyao Huang*, Kui Wu†, Weiwei Wu‡, Xiangtong Qi§, Jianping Wang*, Jen-Ming Wu¶
*City University of Hong Kong, Hong Kong, China †University of Victoria, Victoria, Canada
‡Southeast University, Nanjing, China §Hong Kong University of Science and Technology, Hong Kong, China
¶Hon Hai (Foxconn) Research Institute, Taiwan, China
Email: zhuang88@cityu.edu.hk, wkui@uvic.ca, weiweiwu@seu.edu.cn,
ieemqi@ust.hk, jianwang@cityu.edu.hk, jen-ming.wu@foxconn.com

***Abstract*—Digital Twin-empowered Vehicular Edge Computing (DTVEC) integrates digital twin models with edge computing in vehicular networks to enable real-time, high-fidelity services such as autonomous driving and traffic management. These services depend on timely and temporally aligned data from multiple sources. To quantify this requirement, we introduce the Age of Fusible Information (AoFI), a novel metric that captures both data freshness and inter-source temporal alignment. Minimizing AoFI through distributed subchannel allocation across base stations (BSs) is a challenging NP-hard problem due to many-to-many relationships, inter-BS interference, and vehicle mobility. To address this, we propose a multi-agent automatic heuristic design (AHD) framework that leverages the interpretability and adaptability of large language models (LLMs) for dynamic, context-aware heuristic search. Unlike conventional learning-based methods, our approach discovers interpretable, resource-efficient heuristics guided by a centralized diagnostic signal called *panorama*. Extensive evaluations using real-world vehicle data show that LLM-generated heuristics outperform state-of-the-art learning-based and manually crafted baselines at minimizing AoFI and generalize well across diverse, unseen contexts.**

I. Introduction

Digital twin (DT) technology, which constructs high-fidelity virtual replicas of physical entities and systems, has gained traction in diverse applications such as healthcare monitoring [1] and autonomous driving [2]. Its rapid adoption stems from capabilities in real-time monitoring, predictive analytics, and informed decision-making through simulations and "what-if" analyses. Typically, as shown in Fig. 1, DT systems aggregate data from multiple information sources at edge servers via base stations (BSs) to create precise digital representations. Vehicular edge computing (VEC), closely related to DT, addresses computation-intensive and time-critical tasks, especially in intelligent transportation and autonomous driving. Integrating DT into VEC systems, known as DT-empowered VEC (DTVEC) [3], enhances responsiveness and decision accuracy in dynamic vehicular environments, improving service quality.

Efficient provisioning of DT services in DTVEC fundamentally relies on fresh data from multiple sources. Specifically, to ensure high-fidelity DT models, data from various sources must be temporally well-aligned, meaning the generation times of the data involved in fusion processes should lie within strict timing thresholds. For example, [4] shows that in vehicular data fusion, the quality can drop by over 50% when the misalignment threshold reaches 100 ms. Moreover, data from all sources must be uploaded frequently to maintain the continuous freshness of the information.

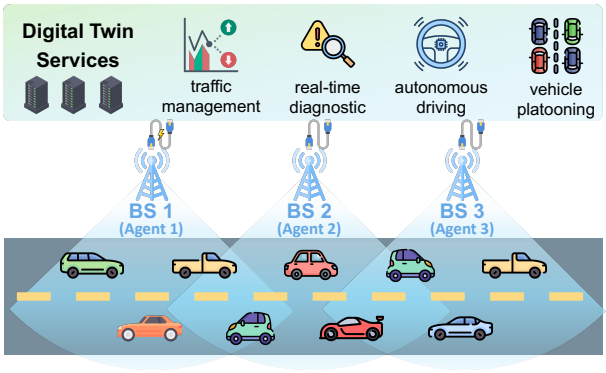


Fig. 1: System architecture of a typical DTVEC.

Ensuring precise temporal alignment and frequent data uploads requires effective subchannel allocation strategies at BSs to manage transmissions efficiently. However, designing such strategies is inherently challenging due to the complex many-to-many relationships among system components: a single information source may simultaneously serve multiple digital twins (DTs), each DT typically aggregates data from several sources, and individual BSs concurrently support multiple DTs and information sources. This many-to-many allocation challenge is a variant of the notorious 3-dimensional matching (3DM) that is shown to be NP-hard [5]. Further complicating the problem are real-world constraints such as distributed multi-BS coordination, vehicle mobility, and inter-BS interference. To address these difficulties, researchers have explored neural network (NN)-based approaches, particularly deep reinforcement learning (DRL), for subchannel allocation in multi-BS or even single-BS VEC environments [6], [7]. While promising, NN-based methods often suffer from limited interpretability and demand substantial computational resources (e.g., GPUs) to achieve satisfactory performance.

Jianping Wang is the corresponding author.
This work is supported by the Collaborative Research Fund under Project No. C1042-23GF, and the National Natural Science Foundation of China under Grant No. 62572120.

This paper approaches this challenge from a novel angle: leveraging the interpretability and strong code-generation capabilities of large language models (LLMs) to automatically discover effective heuristics. This paradigm, known as automatic heuristic design (AHD), has recently gained traction in works such as EoH [8] and AlphaEvolve [9], which combine evolutionary computation (EC) and LLMs to solve classical optimization problems, such as bin packing and routing. Compared to NN-based policies, LLM-generated heuristics[1] offer several distinct advantages: they produce human-readable decision rules that facilitate long-term maintenance and domain-specific customization, and they have been shown to discover heuristics that significantly outperform state-of-the-art (SOTA) solutions [10], [11].

Despite the pioneering advances of EoH and AlphaEvolve, adapting LLM-based AHD methods to our setting is far from straightforward, due to two key challenges. First, existing approaches are tailored for single-agent environments and do not address the inter-agent coordination complexities inherent in distributed multi-agent systems. Second, these methods are primarily designed for classical combinatorial optimization problems with relatively short and simple structures, where a single scalar objective can effectively guide the search process. In our case, the problem involves more intricate dependencies and dynamics, making it difficult to rely on a single score as meaningful feedback. We systematically tackle all the above challenges by developing a multi-agent LLM-based AHD framework. Our contributions include:

- We propose a new information freshness metric AoFI, which captures the stringent data alignment requirements in multi-source fusion for DTVEC. Unlike existing metrics, AoFI accounts for both inter-source correlation and tolerance to temporal misalignment, making it well-suited for fusion-centric applications.
- We develop a novel multi-agent LLM-based AHD framework that extends existing single-agent AHD methods to distributed, multi-agent settings. The framework introduces a global diagnostic signal, termed *panorama*, which captures system-wide dynamics to guide heuristic evolution. Our framework supports both (1) manually designed *panorama* to enhance AHD through domain expert knowledge and (2) automatic search of critics that yield *panorama* from existing environmental feedback. The proposed framework can substantially enhance existing methods, notably MCTS-AHD [10], which previously underperformed compared to learning-based baselines. It now discovers superior heuristics for the first time. We also utilize our framework to enhance CALM [11], which can achieve further improvements beyond its already competitive performance.
- Leveraging the developed multi-agent AHD framework, we discover a new SOTA heuristic for the distributed subchannel allocation problem in DTVEC. Evaluations on real-world data reveal that it consistently achieves the lowest AoFI, highest instance-wise winning ratio, and highest throughput across various settings. Notably, it generalizes well to unseen time periods and cities, maintaining superior performance across problem scales and spatiotemporal variations.

[1]Our LLM-based approach poses no latency concerns for real-time channel allocation, as heuristic generation and evaluation occur offline, prior to deployment.

II. Related Work

A. Information Freshness Metrics.

The mainstream information freshness metric is the age of information (AoI) [12], defined as the time elapsed since the generation time of the most recent data received by the information holder from the information source. To quantify the freshness of data fed to DTs, AoI is one of the most popular choices [13], [14]. Additionally, several new freshness metrics have been proposed. For instance, [15] developed the Ultra-low AoI (ULAoI) metric to capture the impact of extreme events where the instantaneous AoI exceeds some predefined thresholds. Overall, existing freshness metrics employed for DT's data are mostly defined between one information holder and one information source, which are inapplicable to our context, where a DT requires temporally aligned data from multiple sources.

Beyond DT-specific literature, several freshness metrics consider multiple sources per holder, such as the age of correlated information (AoCI) [16], age of services (AoS) [17], coage of information (CoI) [18], age of usage information (AoUI) [19], and age of collection (AoC) [20]. Nonetheless, these metrics have limitations in the context of DTVEC. For example, in AoS, CoI, and AoUI metrics, freshness can improve even if updates from some relevant sources are missing. Conversely, the AoCI only improves when all sources deliver data simultaneously. This metric does not account for misalignment tolerance and imposes an overly stringent requirement for real-time fusion in continuous-time domains. AoC, meanwhile, disregards packet generation times entirely, treating data as fresh once all packets have arrived.

Unlike all existing information freshness metrics, the AoFI metric proposed in this paper uniquely captures both the correlation among multiple sources and a tolerance for temporal misalignment. Its computation hinges on a joint, window-based matching of updates across sources, where the freshness value is reset only when the generation timestamps of packets received from all sources collectively fall within a specified tolerance window. This interdependence renders AoFI a novel metric that is particularly useful for multi-source, fusion-based DT systems.

B. LLM-based AHD

Existing LLM-based AHD methods [8], [9], [10], [11] operate through an evaluate–generate loop. In each iteration, information about the current elite heuristics, such as natural language descriptions, source code, and performance scores, is integrated with predefined prompt templates inspired by genetic operators to construct new prompts. The LLM then

generates new candidate heuristics in response to the prompts, which are evaluated based on their performance. The results are incorporated into subsequent prompts, and this cycle continues until the evaluation budget is exhausted. Notably, while most approaches use a frozen LLM throughout the process, [11] enhances search by fine-tuning the LLM during evolution.

These methods are predominantly designed for single-agent problems with relatively simple structures, such as classical combinatorial optimization tasks. As a result, they rely solely on scalar performance scores for feedback. This simplification limits their applicability to more complex settings, such as ours, which involves distributed BS coordination and intricate dependencies. In such environments, richer feedback signals are required for effective heuristic search.

While [21] represents the first attempt to apply LLM-based AHD to a networking problem, it does not explicitly tackle the challenges posed by multi-agent coordination or structural complexity. Another line of work in networking leverages LLMs as *direct* decision-makers [22], [23]. However, this approach is unsuitable for our scenario, as the latency associated with invoking either local or API-based LLMs does not align with the stringent requirements of high-frequency decision-making.

III. System Model

A. System Components

We consider a DTVEC system comprising I information sources (ISs), J base stations (BSs), and K digital twins (DTs). Correspondingly, we index these components in sets $\mathcal{I} = \{1, 2, \ldots, I\}$, $\mathcal{J} = \{1, 2, \ldots, J\}$, and $\mathcal{K} = \{1, 2, \ldots, K\}$. To deliver high-quality services, DTs rely on up-to-date data from ISs for refreshing their models [24] or for predictions [25]. A DT may rely on information from multiple ISs, and an IS may provide its data to multiple DTs. For instance, a traffic DT would require trajectory and motion data from multiple vehicles to predict road traffic [26], and a vehicle may provide its data to a traffic DT for road traffic prediction and to a vehicle DT for real-time vehicle diagnosis. We use a binary variable $x_{i,k}(t) \in \{0, 1\}$ to denote whether IS i is associated with DT j at time t, i.e., provides its data to DT j. Here, $x_{i,k}(t) = 1$ indicates that IS i is associated with DT j at time t.

Due to the considerable resource demands for deployment, these DT applications cannot be executed locally on ISs. Instead, they are hosted on BSs or in close vicinity to them. As a result, data from an IS $i \in \mathcal{I}$ must be first transmitted to a nearby BS $j \in \mathcal{J}$ through wireless communication. Subsequently, the data is forwarded to the host of the relevant DT using optical fibre connections. Given that multiple ISs may concurrently connect to the same BS, it is imperative that BSs prudently manage bandwidth allocation among connected ISs to ensure each IS can upload data at an adequate rate.

We denote by $y_{i,j}(t) \in \{0, 1\}$ whether IS i connects to BS j at time t, where $y_{i,j}(t) = 1$ means yes and otherwise not. The inclusion of a timestamp is necessary because ISs may move, leading to transitions between the coverage areas of different BSs. Let $\hat{I}_j(t) = \{i \in \mathcal{I} | y_{i,j} = 1\}$ denote the set of ISs connected to BS j at time t. Assume that a mobility management mechanism [27] has been implemented on BSs, such that BSs can track the mobility of an IS and seamlessly switch their connection to the ISs. Also, it is assumed that each IS is connected to one BS at any instant of time, that is, $\sum_{j \in \mathcal{J}} y_{i,j}(t) = 1, \forall i, t$.

B. Transmission Model

To provide necessary information to associated DTs, each IS transmits its data to its associated DTs via its connected BS. The data transmission model comprises the upload from the IS to the BS and the forwarding from the BS to the associated DTs. We assume a work-conserving mode, i.e., whenever there is data to transmit, the channel should not remain idle.

IS-to-BS Uploading. We consider the Orthogonal Frequency Division Multiple Access (OFDMA) mechanism, which is widely used in LTE and 5G networks. Under this mechanism, the available bandwidth resources of the BS are equally divided into M orthogonal subchannels/subcarriers of size b Hz. Formally, let $\mathcal{M} = \{1, 2, \cdots, M\}$ be the index set of the subchannels; let $n_{i,j,m}(t) \in \{0, 1\}$ indicate whether subchannel m is allocated to IS i at time t (where $n_{i,j,m}(t) = 1$) or not. Since the BS can allocate each subchannel to at most one connected IS, we have

$$\sum_{i \in \mathcal{I}: y_{i,j} = 1} n_{i,j,m}(t) \leq 1, \forall m, j. \tag{1}$$

Moreover, the BS can only allocate subchannels to ISs that are connected to it, i.e.,

$$n_{i,j,m}(t) = 0, \forall i \in \{i' \in \mathcal{I} | y_{i',j}(t) = 0\}. \tag{2}$$

Since the subchannels are orthogonal and ISs connected to the same BS will not share the same subchannel, the intra-BS interference is neglected [28]. Still, ISs connected to different BSs that are assigned to the same subchannel may suffer from inter-BS interference. The communication quality is measured by the signal-to-interference-plus-noise ratio (SINR). The SINR from IS i to its connected BS j on subchannel m at time t is as follows [29]:

$$\gamma_{i,j,m}(t) = \frac{p_{i,j,m} g_{i,j,m}(t)}{N_G + \sum\limits_{j \neq j'} \sum\limits_{i' \in \mathcal{I}} y_{i',j}(t) n_{i,j',m}(t) p_{i',j',m} g_{i',j',m}(t)}, \tag{3}$$

where N_G is the additive Gaussian white noise (AGWN), $p_{i,j,m}$ is the transmit power employed by IS i on subcarrier m for transmission to BS j, and $g_{i,j,m}(t)$ is the uplink channel gain between IS i and BS j on subchannel m. The channel gain is typically impacted by path loss, shadowing, and fading [30].

Consequently, the uploading data rate from an IS i to its connected BS can be measured by the Shannon-Hartley formula as follows:

$$r_i(t) = b \sum_{j \in \mathcal{J}} y_{i,j}(t) \sum_{m \in \mathcal{M}} n_{i,j,m}(t) \log\left(1 + \gamma_{i,j,m}(t)\right). \tag{4}$$

BS-to-DT Forwarding. If the DT is deployed at the BS, the data can be directly fed to the service program. Otherwise, the data will be further forwarded to the server hosting the DT. Assume that BSs and servers are interconnected via optical fibres, which is significantly faster than wireless channels. Thus, the time required to upload ISs' data to DTs is dominated by the wireless IS-to-BS transmissions, and we can treat the time spent in BS-to-DT forwarding as a negligible constant.

Overall Delay. Denoted the size of the data packet from IS i as s_i. If IS i starts to send a data packet to DT k at time t, the transmission duration for this data packet is

$$D_{i,k}(t) = \min_{t'>t} \left\{ s_i \leq \int_{\tau=t}^{t'} r_i(\tau) d\tau \right\} - t. \tag{5}$$

Equivalently, if a data packet arrives at DT k from IS i at time t, the starting time of its uploading is

$$G_{i,k}(t) = \max_{t'<t} \left\{ s_i \leq \int_{\tau=t'}^{t} r_i(\tau) d\tau \right\}. \tag{6}$$

More generally, the uploading start time of the most recent packet that has arrived at DT k from IS i till time t is

$$G^R_{i,k}(t) = \begin{cases} -C, \text{ if } L_{i,k}(t) = 0; \\ G_{i,k}\left(\max\left\{t' \leq t | A_{i,k}(t) = 1\right\}\right), \text{ otherwise.} \end{cases} \tag{7}$$

In the equation above, C is a large positive constant; $A_{i,k}(t) \in \{0, 1\}$ is a binary variable indicates whether a packet from IS i arrives at DT k at time t (where $A_{i,k}(t) = 1$) or not; $L_{i,k}(t) = \mathbf{1}\left\{\exists t' < t, \textit{ s.t. } A_{i,k}(t) = 1\right\}$ indicates whether or not at least one packet has arrived at DT k from IS i till time t, *s.t.* means "subject to".

Remark 1. *In the model above, we assume an IS provides the same type of data packet to all associated DTs; however, in practice, it may supply heterogeneous data packets to different DTs. In such cases, we can reformulate the problem by replacing the single IS with multiple dummy ISs, each providing a specific data packet type to a subset of DTs. The resulting bandwidth allocation can then be mapped back to the original scenario by equating the bandwidth of each dummy IS to the upload bandwidth of the corresponding data packet within the original IS. This approach is feasible because the upload bandwidth within a single IS, such as a vehicle, can typically be programmatically allocated without restriction. For example, Linux systems enable per-process bandwidth control by packet marking with iptables, combined with traffic shaping via Linux traffic control (tc).*

C. A New Metric for Data Freshness

Data freshness plays a pivotal role in the quality of the DT, which is commonly measured by the AoI metric. From the perspective of a DT, the instantaneous AoI of an IS is defined as the time elapsed since the generation time of the latest received data sent by the IS. The AoI metric is defined between one information source and one information holder, and it considers the information fresh upon receiving the recently generated data from a single IS. However, a DT may simultaneously utilize information from multiple ISs for fusion, and the data can be useful only when data received from all sources are synchronized[2]. This has motivated us to develop a new freshness metric named age of fusible information (AoFI).

Denote the set of ISs associated with DT $k \in \mathcal{K}$ as $\tilde{\mathcal{I}}_k(t) = \{i \in \mathcal{I} | x_{i,k}(t) = 1\}$. We assume that the generation times from these ISs need to be synchronized before fusion. Consequently, the AoFI of a DT k drops each time synchronously generated data packets from all associated ISs have arrived. Let $o_k > 0$ be the misalignment tolerance value for DT k. We assume the data packets are synchronously generated and therefore fusible, if the maximum difference in their generation times does not exceed o_k.

We consider the generate-at-will model [31], i.e., data at IS is continuously generated, and the data sampling time is neglected. Thus, the generation time of a data packet is the beginning time of its uploading, as shown in (6) and (7).

The AoFI Metric. Let

$$f_k(t, t_1, \cdots, t_I) = \mathbf{1}\left\{ \prod_{i \in \tilde{I}_k(t)} L_{i,k}(t) = 1 \text{ and } \left(\max_{i \in \tilde{I}_k(t)} - \min_{i \in \tilde{I}_k(t)} \right) G^R_{i,k}(t) \leq o_k \right\} \tag{8}$$

be the function that indicates whether or not data can be fused at DT k at time t from the data packets arrived at $t_1, \cdots, t_I$ from IS $1, \cdots, I$, where $\tilde{I}_k(t) = \{i \in I | x_{i,k}(t) = 1\}$ is the set of ISs that are serving DT k at time t. Subsequently, the earliest generation time of the packet used in the most recent fusion at DT k is

$$G^F_k(t) = \max_{\substack{t' \leq t \\ t_i \leq t', \forall i \in \mathcal{I}}} f(t', t_1, \cdots, t_I) \cdot \min_{i \in \tilde{I}_k(t)} G^R_{i,k}(t'). \tag{9}$$

Consequently, the AoFI value of DT k at time t is as follows:

$$h_k(t) = t - G^F_k(t). \tag{10}$$

Notably, when fusion has not been incurred, the indicator function f always returns 0, and $G^R_{i,k}$ yields a constant even when no packet from an IS has arrived at any DT (where $G^R_{i,k}(t) = -C$). Thus, the AoFI value increases from 0 when $t = 0$. This essentially assumes a fresh start for the system.

For each DT k, we assume a positive scalar w_k indicating its weight/popularity. Then, given the time horizon T, we measure the information freshness of the DTVEC system by the time-average weighted AoFI (TAWA):

$$H(T) = \sum_{k \in \mathcal{K}} w_k \cdot \frac{1}{T} \int_{t=0}^{T} h_k(t) dt. \tag{11}$$

[2]Data synchronization in DT is domain-specific. For instance, a traffic flow DT may treat packets generated and received within a 200 ms window as fresh and can fuse them for practical use. Hence, the term *synchronization* here does not necessarily imply strict time equivalence.

D. Problem Definition

We assume that the host of the DTVEC system either possesses the BSs, such as independently deployed roadside units, or engages in a collaborative arrangement with the network operator, for instance, the telecommunications service provider managing 5G gNBs, to administer the bandwidth allocated to ISs. To optimize the quality of all DTs in the DTVEC system, we investigate the problem of minimizing the TAWA by adapting the subchannels BSs distributedly allocated to the connected ISs over time.

Problem 1. ***TAWA-Optimizing Subchannel Allocation (TOSA) Problem*** *Assume that the following information for each BS $j \in \mathcal{J}$ at time t is given: the set of connected information sources (ISs) $\tilde{\mathcal{I}}_j(t)$ (including each IS's distance to the BS, transmission power, data packet size, and the set of DTs it serves), the set of DTs $\mathcal{K}$ (including each DT's weight and misalignment tolerance), the time horizon T, and the bandwidth of each subchannel b. The objective of the TOSA problem is for each BS $j \in \mathcal{J}$ to independently determine its subchannel allocation $\boldsymbol{n}_j(t) = (n_{i,j,m}(t))_{i\in\tilde{\mathcal{I}}_k(t),m\in\mathcal{M}}$, indicating which subchannels are assigned to each connected IS over time, in order to minimize the TAWA defined in (11), subject to the constraints in (1) and (2).*

We can reduce the NP-hard 3DM problem [5] to prove the hardness of TOSA. The detailed proof is omitted to save space.

Theorem 1. *The discrete-time TOSA problem is NP-hard.*

IV. Panorama: A Novel Approach to Solving TOSA

The combination of challenges—including stringent data alignment requirements, many-to-many resource allocation, inter-base station interference, high vehicle mobility, and system decentralization—makes it difficult to discover an effective heuristic solution. To navigate this complexity, we develop a new AHD approach by leveraging the strong code generation capabilities of LLMs [8], [11].

A. The Preliminary Evolutionary Search Framework

We follow the prevailing paradigm in LLM-based AHD to construct an evolutionary search framework as illustrated in Fig. 2. This framework accepts a user-provided textual task description and outputs the best-discovered heuristic. In each iteration, high-performing (elite) heuristics, stored in a base, are used to prompt the LLM to generate new heuristic candidates. These candidates are subsequently evaluated, and those demonstrating promise are added to the base. Key components of the framework are:

LLM. The LLM primarily serves as the heuristic generator. Given textual prompts, it produces structured outputs from which complete heuristic descriptions—typically including executable code—can be extracted. Besides, the LLM serves as a reflective agent, generating feedback based on prior search trajectories to guide subsequent generations.

Operator Base. This is a fixed collection of predefined operators, each representing a prompt-generation template tailored to specific tasks. Operators guide the heuristic creation process by shaping the prompt fed to the LLM. For instance:

- The *simplify* operator prompts the LLM to generate a simplified variant of an existing heuristic.
- The *mutate* operator asks the LLM to propose a substantially different alternative for the provided heuristic.
- The *crossover* operator encourages the synthesis of a new heuristic motivated by multiple existing ones.
- In the absence of prior heuristics, the *create* operator prompts the LLM to design a heuristic solely based on the task description.

Heuristic Base. This repository maintains both seed heuristics (if provided) and those generated during the evolutionary process. Each heuristic is characterized by multiple attributes, like its core idea, implementation code, and performance score. These attributes are embedded within the prompt when referencing a heuristic during LLM invocation.

Prompt Composer. This module governs the selection of operators and heuristics used to construct prompts for the LLM. It balances exploration and exploitation by choosing between diverse operators and prioritizing heuristics based on attributes such as performance or diversity.

Training Environment. This module evaluates the feasibility and performance of candidate heuristics based on their implementation. Infeasible heuristics are discarded. For valid candidates, the performance score is logged and used both as a reference in prompts and as a selection criterion. Higher-performing heuristics are more likely to be reused, increasing their influence on subsequent generations.

B. Panorama: Beyond Scores as the Centralized Lens

A significant advancement in LLM-based AHD was introduced by EoH [8]. Its critical innovation is an enriched heuristic representation, where heuristics are characterized not only by executable code but also by concise linguistic descriptions capturing their core ideas. This dual-mode representation substantially improves search efficiency—the number of LLM queries required to identify effective heuristics—and the quality of the discovered solutions. This richer representation is also employed by the most recent SOTA methods, such as MCTS-AHD [10] and CALM [11].

Nevertheless, existing LLM-based AHD frameworks typically rely solely on performance scores as the environmental feedback to inform heuristic generation. Although this minimalist approach proves adequate for classical combinatorial optimization problems, such as the travelling salesman or bin-packing tasks, it faces critical limitations when confronted with more complex, multi-agent problems like TOSA, where performance can be influenced by numerous intertwined factors that cannot be captured with a single final score.

We thus propose *Panorama*, a new AHD approach that uses high-dimensional environmental feedback collected directly from the centralized training environment during heuristic evaluation. The term *Panorama* emphasizes its ability to provide a holistic view of the multi-agent system dynamics.

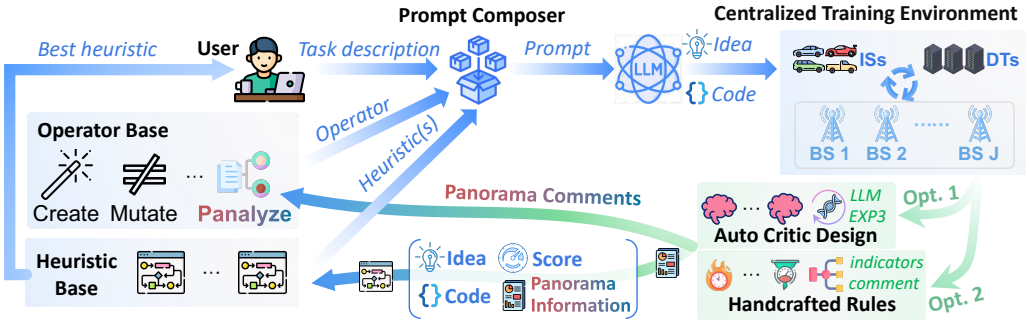


Fig. 2: Our multi-agent LLM-based AHD framework.

C. Evolution with Panorama

Our multi-agent AHD framework takes as input a single-agent LLM-based method, such as CALM [11] and MCTS-AHD [10], and empowers it by adding a crucial component: the *panorama*. This addition enables the framework to surpass simple scalar scores by incorporating system-level feedback gathered during centralized training. With this feedback, the LLM can better interpret the behavior of a heuristic in a multi-agent context and generate more targeted improvements during the search process. The *panorama* component encompasses two constituents: *panorama* information used to enrich the representation of heuristics and *panorama* comments used in a new operator named *Panalyze* that directly revises an existing heuristic. We disclose their details below.

Enriched Heuristic Representation. Every heuristic now carries, in addition to code, idea summary, and performance score, extra *panorama* information: a set of high-level feedback indicators, e.g., the average ratio of successfully fused packets to arrived packets across all vehicles. During prompt composition, the *panorama* information is displayed alongside other attributes, enabling the LLM to reason about where a heuristic excels or falters in the multi-agent setting. To prevent biasing the native operators of the underlying AHD method, the *panorama* information is presented in a neutral, concise format, serving as supplementary context for the LLM to ground its reasoning.

***Panorama*-Aware Operator: *Panalyze*.** To directly guide heuristic evolution using this holistic environmental feedback, we introduce a novel operator, *panalyze* (a coined term blending *panorama* and analyze). This operator explicitly prompts the LLM to refine an existing heuristic by "*panorama* comments". Specifically, *panorama* comments are explicit diagnostic insights derived from *panorama* information of this heuristic. For example, when the ratio of fused packets to received packets falls below a given threshold, this indicates significant temporal misalignment among data packets; as such, the system suggests improving the heuristic's packet alignment efficiency. This mechanism ensures that code revisions are driven by clear evidence from the system dynamics.

Remark 2. *Panorama is used only during the centralized heuristic search stage for heuristic revisions. It is not required during heuristic deployment, which remains fully decentralized. This follows the widely adopted centralized-training-decentralized-execution (CTDE) paradigm used in multi-agent deep reinforcement learning [32], ensuring compatibility with the distributed operational constraints of TOSA.*

D. Panorama with Prior Domain Knowledge

To fully leverage LLM's semantic understanding capability, the *Panorama* framework moves beyond traditional scalar "reward design" by incorporating interpretable linguistic feedback. This allows insights from domain experts to be injected directly into the AHD process. We first introduce the *panorama* generation based on handcrafted rules designed specifically for the TOSA problem.

The joint subchannel allocation decisions made by BSs fundamentally influence the system's TAWA. This impact propagates through a causal chain of interrelated factors: subchannel allocation $\rightarrow$ inter-BS interference $\rightarrow$ throughput $\rightarrow$ fusion success rate $\rightarrow$ AoFI values. To facilitate targeted diagnosis, we define corresponding *panorama* constituents for each stage. The final system-level *panorama* is formed by aggregating the information and comments generated across these stages.

Inter-BS Interference: We focus on identifying scenarios where vehicles are systematically subjected to high interference (defined as *victim* vehicles frequently assigned *hot* subchannels). The *panorama* information reports the proportion of simulated instances containing such victims. If this ratio is significant, the *panorama* comment explicitly suggests improving sub-channel popularity estimation and strengthening inter-BS coordination.

Throughput: To detect bandwidth allocation imbalances, we utilize Tukey's fence [33] to identify *starved* vehicles (low-side outliers in packet-arrival frequency). The *panorama* information signals the percentage of instances containing starved vehicles. Consequently, the *panorama* comment advises a proactive revision of the heuristic to balance bandwidth utilization and prevent such starvation.

Fusion Efficiency: We assess data usage efficiency via the ratio of fused packets to total arrived packets, flagging instances as *fusion-inefficient* if the mean ratio is below a threshold. The *panorama* information quantifies the frequency of such

instances. When detected, the *panorama* comment highlights that packets are arriving but failing to fuse, urging the heuristic to improve the temporal alignment of packets within the fusion window.

DT Freshness: Finally, we apply outlier detection to weighted AoFI values to identify *freshness imbalance*. The *panorama* information provides the percentage of imbalanced instances and the rank of the worst-served DT (to indicate if high-priority twins are neglected). The resulting *panorama* comment prompts the LLM to judiciously adjust resource allocation or fairness in scheduling to rectify these disparities.

E. Panorama without Prior Knowledge: Automatic Critic Design

While handcrafted rules effectively inject expert knowledge, they may not be available or optimal for every scenario. To enhance the applicability of our *Panorama* framework, we propose the Automatic Critic Design (ACD) method to adaptively generate *panorama* information and comments when manually-crafted feedback is unavailable.

A straightforward approach would be to utilize an LLM to directly criticize each heuristic. However, this strategy is computationally expensive, doubling the number of LLM queries and requiring additional iterations to optimize the critic prompt. As a remedy, we propose to shift the role of the LLM from a direct critic to a *critic designer*. Specifically, we prompt the LLM to synthesize executable critic functions that map the environmental feedback of an existing heuristic to a tuple of texts (comprising *panorama* information and comments). This interface allows a generated critic to be reused across multiple heuristics, significantly reducing the token overhead for *panorama* generation.

Managing the set of generated critics presents two key challenges: (1) selecting the most effective critic, and (2) improving the quality of critics over time.

Critic Selection as Adversarial MAB. We address the selection challenge by formulating it as an adversarial Multi-Armed Bandit (MAB) problem. In this context, we can only indirectly estimate a critic's quality by observing the performance of heuristics generated from its feedback. Since we must identify the best critics rapidly within a limited trial budget, we adopt the EXP3 algorithm [34].

In our formulation, each critic acts as an expert. We define the reward r_t for the *panorama* information/comment provided by a critic at the t-th heuristic generation as 1 if the heuristic revised by the critic's comment outperforms the original version, and 0 otherwise. Under EXP3, each critic is assigned a weight, and critics are sampled with a probability proportional to this weight. The weight is updated upon usage: it remains unchanged if the reward is 1 and decays exponentially if the reward is 0.

Critic Evolution. To address the second challenge, we leverage the success of evolutionary search in LLM-based AHD. Treating the critic functions as heuristics, we apply a genetic framework to evolve them. We employ the following operators: (1) *create*: Prompts the LLM to generate a new critic, enforcing diversity from existing ones; (2) *crossover*: Generates a new critic inspired by two parent critics; (3) *mutate_comment*: Revises a critic's comment logic without altering the information extraction logic; and (4) *mutate_panorama*: Creates a fully mutated version of an existing critic.

Overall ACD Workflow. The ACD method, detailed in Algorithm 1, combines EXP3-based selection with genetic evolution. In each round where *panorama* is required, a critic is sampled, and its weight is updated based on the heuristic generation reward r_t. To maintain population quality, we remove the critic with the lowest weight every U rounds and generate a new one via the evolutionary operators. Notably, we adapt the standard EXP3 update rule by replacing the global time step t with the critic's individual age. Standard EXP3 assumes a static set of experts; our adaptation ensures that dynamically created critics do not suffer disproportionate weight penalties due to failures in their early "life", thereby facilitating a fairer comparison in a dynamic population.

Algorithm 1: Automatic Critic Design (ACD).

Input: A LLM, The maximum population size $P_{\max}$ for critics, the update frequency U for the critic population.

$\mathcal{C} \leftarrow \emptyset, T_c \leftarrow 0, c' \leftarrow \text{null}, r' \leftarrow 1$;

When *Panorama information or comment is needed at t-th heuristic generation*

- $T_c \leftarrow T_c + 1$;
- **if** $T_c \bmod U = 0$ **then**
  - $\mathcal{C} \leftarrow \mathcal{C} \setminus \left\{ \arg\min_{c \in \mathcal{C}} c.\textit{weight} \right\}$;
- **if** $|\mathcal{C}| < P_{max}$ **then**
  - prompt $\leftarrow$ Draw an operator and existing critics (if required by the operator) and create a prompt for new critic generation;
  - $c_{new} \leftarrow \text{LLM}(\text{prompt})$;
  - $c_{new}.\textit{weight} \leftarrow 1$, $c_{new}.\textit{age} \leftarrow 1$, $c_{new}.\textit{loss} \leftarrow 0$;
  - $\mathcal{C} \leftarrow \mathcal{C} \cup \{c_{new}\}$;
- $w_{sum} = \sum_{c \in \mathcal{C}} c.\textit{weight}$;
- $c_t \leftarrow$ Draw a critic from $\mathcal{C}$ where $\Pr\{c_t = c\} = \frac{c.\textit{weight}}{w_{sum}}$;
- Apply *panorama* generated by c_t;
- Observe r_t by the heuristic generation result;
- $c_t.\textit{loss} \leftarrow c_t.\textit{loss} + \frac{1 - r_t}{c_t.\textit{weight}/w_{sum}}$;
- $c_t.\textit{weight} \leftarrow \exp\left\{ c_t.\textit{loss} \cdot \sqrt{\frac{2 \log P_{max}}{P_{max} c_t.\textit{age}}} \right\}$;
- **for** $c \in \mathcal{C}$ **do**
  - $c.\textit{age} \leftarrow c.\textit{age} + 1$;

V. Experimental Evaluation

A. Experimental Setup

Environmental Settings. We assume that the communication system has $M = 64$ subchannels, uniformly dividing a total bandwidth of 10 MHz. The path loss, shadowing, and fading follow the setup in [30]. The AGWN power is -114 dBm [35].

To obtain a vehicle's physical location at any time, we apply linear interpolation over the real trajectory data from the FIB-T dataset [36], which includes vehicle trajectories reconstructed from real video data for two Chinese metropolises: Shenzen and Jinan. Specifically, we construct a training set consisting of 20 instances. Each instance includes data within a 3-minute interval randomly selected from the forenoon hours (9:00 am

$\sim$ 12:00 pm) of November 4, 2020, in Shenzhen. For each instance, we randomly determine the number of DTs, selecting a value within the range of 10 to 30. For each DT, 2 to 15 vehicles are randomly chosen from the set of vehicles currently connected to a BS to serve as its data sources. The number of serving vehicles per DT is randomly chosen.

Next, we randomly determine the number I of BS in the map, choosing a value uniformly from the range $[2, 10]$. We then use the bisecting K-means clustering algorithm to partition the trajectory points into the I clusters. The center of each cluster is treated as the location of a BS[3].

A vehicle connects to a BS only if their Euclidean distance does not exceed 500 m [30]. When a vehicle is within the coverage range of multiple BSs, it connects to the BS currently serving the fewest vehicles to achieve load balancing. Each vehicle i has its transmission power p_i sampled uniformly from the interval $[21, 23]$ dBm. The packet size s_i is randomly selected within the range of 500 to 4000 bytes, thereby capturing a diverse range of vehicular applications. For example, in cooperative driving scenarios, packets conveying safety event reports and environmental perception data typically measure around 800 and 1200 bytes, respectively [39].

LLM-based AHD Settings. We evaluate four framework variants: **PanoCA-H**, **PanoMC-H**, **PanoCA-ACD**, and **PanoMC-ACD**. Here, **MC** and **CA** denote the MCTS-AHD [10] and CALM [11] backbones, respectively, both enhanced by the Panorama framework. Regarding the signal source, **H** employs the human-designed signals from Section IV-D, while **ACD** leverages the ACD mechanism introduced in Section IV-E[4].

Each heuristic generated by the LLM is designed to assign a score to each vehicle-subchannel pair from the perspective of a BS. If all scores are non-positive (a positive score means a suitable allocation), the subchannel is left unallocated for that BS. Otherwise, the vehicle with the highest score is granted access to the subchannel. The heuristic is invoked every 100 ms to update the allocation. Each LLM-based AHD method is executed three times, and unless otherwise specified, we report the performance of the heuristic that achieved the best average score on the training dataset.

Note that we also evaluated AlphaEvolve [9] on our task, using its open-source implementation, *OpenEvolve* [40]. However, the heuristics it discovered yielded only marginal improvements over the initial seed heuristic—no more than 15.43% across four runs. In comparison, MCTS-AHD and CALM achieve improvements over 40%, respectively. Due to this limited performance, we did not extend *OpenEvolve* to a multi-agent version or include it as a baseline.

Baselines. We consider three types of baselines: (1) **MCTS** [10] and **CALM** [11]: The current SOTA LLM-based AHD methods. (2) **MAPPO** [32]: A widely adopted multi-agent deep reinforcement learning (MADRL) algorithm. To ensure fair comparison, MAPPO is configured with the same observation and action spaces as the LLM-generated heuristics. To accommodate the dynamic number of vehicles connected to each BS, the model employs a stack of four LSTM layers to process variable-length input sequences. These are followed by a multilayer perceptron (MLP), which outputs the priority values for subchannel allocation. Each LSTM layer has a hidden size of 64, and the MLP has two hidden layers of sizes 256 and 128, respectively. These sizes were chosen to ensure that the per-BS decision time of the MAPPO-trained policy is comparable to that of the heuristics from our method, which is approximately 2 ms on average, as empirically verified. The reward is defined as the negative of TAWA. The policy network is trained on the same dataset. (3) **SFR-S** [41]: A state-of-the-art, manually designed heuristic for uplink scheduling in multi-cell OFDMA networks.

B. Overall Performance

To evaluate the overall performance, we sample 20 instances from the FIB-T dataset, each containing vehicle trajectories. For the best heuristic found by each method, we report the average TAWA and normalized throughput across instances, along with top-1 and top-3 winning ratios, defined as the fraction of instances where a method's TAWA ranks within the top-1 or top-3. The results are summarized in Fig. 3. The results demonstrate that:

1) *Our framework discovers the best-performing heuristic at most instances.* The heuristic discovered by our PanoMC-ACD achieves the lowest TAWA, the best top-1 winning ratios (60%), outperforming all the baselines. Moreover, in 90% of the test instances, the winner is discovered by a *Panorama*-based AHD framework.
2) *Our framework can discover heuristics with highly efficient data alignment.* PanoMC-H identifies a heuristic that, although exhibiting lower throughput than all baseline methods by 6.825% relative to CALM, 79.648% to MCTS, 74.912% to MAPPO, and 87.568% to SFR-S, attains substantially higher TAWA. The TAWA improvements are 12.453%, 79.648%, 16.245%, and 97.888% over CALM, MCTS, MAPPO, and SFR-S, respectively. These results indicate that packets received under the heuristic discovered by PanoMC-H are more effectively temporally aligned.
3) *Incorporating our panorama-based framework into single-agent LLM-based AHD methods enables them to discover heuristics that not only outperform their original versions but also surpass the learning-based MAPPO policy.*
   Specifically, when enhanced with our *Panorama* framework, PanoCA-H and PanoCA-ACD discover heuristics with TAWA values that are 16.226% and 20.755% better than the CALM-discovered heuristic, respectively. Furthermore, while the heuristic discovered by the original MCTS is 12.274% worse than MAPPO, the integration of our framework reverses this gap: PanoMC-H and

[3]Trajectory-aware coverage optimization is critical [37], and clustering-based methods are considered effective [38] for cellular BS deployment.

[4]For ACD, the critic population is set to half of the heuristic population with an evolution frequency of 20 generations. The environment provides per-instance DT AoFI values, average vehicle data rate, and packet arrival counts as inputs.

PanoMC-ACD outperform MAPPO by 16.245% and 25.632%, respectively.

Notably, SFR-S exhibits a significantly higher TAWA compared to other methods, likely due to its unawareness of vehicle mobility and data alignment requirements. Therefore, we exclude it from subsequent discussions.

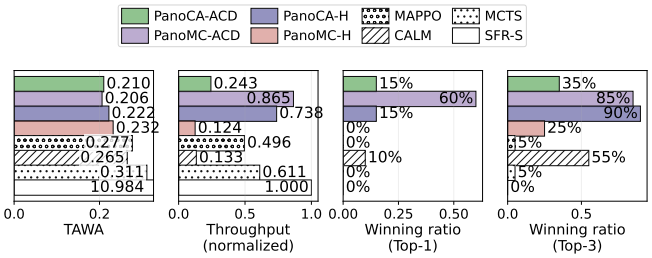


Fig. 3: Overall performance.

C. Discussions

Improved Searching Efficacy. The performance of the best heuristic identified during the search process across all LLM-based AHD methods is illustrated in Fig. 4, with results averaged over three runs. We have two key observations: (1) Compared to their single-agent counterparts without *panorama*, the multi-agent LLM-based methods (PanoCA-H, PanoCA-ACD, PanoMC-H, and PanoMC-ACD) consistently discover significantly higher-quality heuristics; (2) The performance gains do not come at the cost of slower convergence in early steps. Specifically, during the early stages of search (before approximately 300 steps), the quality of the best-discovered heuristic does not differ significantly across methods. However, beyond this point, the heuristics found by most multi-agent variants show markedly superior quality, indicating improved search efficacy.

***Panorama*-Inspired Components in the Best Heuristic**. The best-performing heuristic on the training instances was discovered by PanoCA-H. Due to the space limit, we present only its key code segments in Fig. 5. These segments reveal that the heuristic jointly considers multifaceted factors like transmission urgency, the weights of associated DTs, and mobility. Interestingly, *the LLM autonomously invents a novel feature termed "signal integrity", which does not correspond to any known heuristics*. As shown in Fig. 5, this feature combines last-round data rate, last-round allocation, transmission power, and vehicle–BS distance. The heuristic also implements an allocation mechanism that particularly favours vehicles whose mobility factors fall below the mean within the same BS. Conversely, such behaviors were absent in the best LLM-generated heuristics without *panorama*. This suggests that these advanced components have emerged as a response to *panorama*'s diagnostic feedback, particularly in addressing inter-BS interference and throughput imbalance.

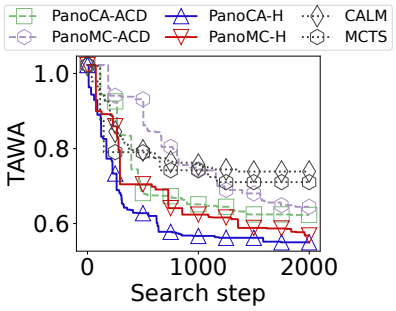


Fig. 4: Best heuristic performance during evolution (smaller TAWA values mean better performance).

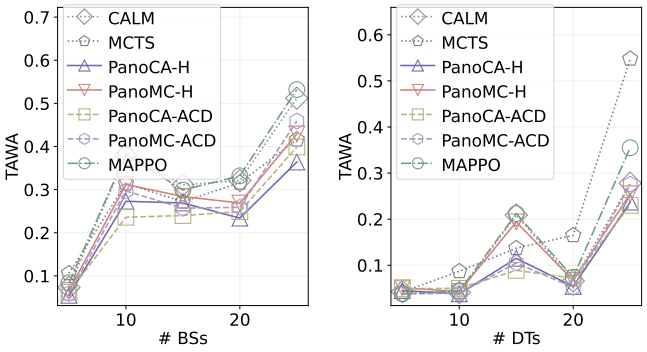
```python
import numpy as np

def heuristic( ... ):
    # Urgency Calculation
    urgency = remaining_packet_size / (time_since_last_packet_arrival +
1e-10)
    ...
    # Relevance Calculation
    if vehicle_serving_dt[vehicle_index, dt_index] == 1:
        relevance[vehicle_index] += DT_weights[dt_index] /
(time_since_last_packet_arrival[vehicle_index] + 1e-10)
    ...
    # Current Signal Integrity Analysis based on real-time conditions
    observed_interference = np.sum(last_data_rate * last_allocation[:,
vehilce_index]) / (np.count_nonzero(last_allocation[:, vehilce_index]) +
1e-10)
    signal_integrity[vehilce_index] = (transmission_power[vehilce_index]
- observed_interference) / distance_to_BS[vehilce_index]
    ...
    # Dynamic Interference Assessment based on Mobility
    mobility_factor[vehilce_index] = 1 + last_data_rate[vehilce_index] /
1000 / distance_to_BS[vehilce_index]
    ...
    # Allocation Metric Calculation refining based on real-time
observations
    allocation_metric = (normalized_urgency + normalized_relevance +
signal_integrity) / (1 + np.mean(mobility_factor) +
np.abs(signal_integrity))
    ...
    # Prioritization for Allocation
    for j in range(n_subchannels):
        vehicle_index = sorted_indices[j % n_vehicles]
        if j < n_vehicles:
            priorities[vehicle_index, j] = 1 +
allocation_metric[vehicle_index] / 2
    else:
        selected_vehicle = sorted_indices[j % n_vehicles]
        if mobility_factor[selected_vehicle] < np.mean(mobility_factor):
            priorities[selected_vehicle, j] = 1 +
allocation_metric[selected_vehicle] / 2
    ...
    return priorities
```

Fig. 5: Key code segments of the best-performing heuristic (discovered by PanoCA-H). The accompanying comments were also automatically generated during the search process.

Performance with Varying Number of BSs. We construct five datasets, each containing 20 instances, with the number of BSs fixed at 5, 10, 15, 20, and 25, respectively. As shown in Fig. 6a, the performance of all methods generally degrades as the number of BSs increases, likely due to the growing complexity of inter-BS coordination for interference mitigation. Nevertheless, *at every scale, the best performance is achieved by a heuristic discovered by our Panorama-based method, either PanoCA-H or PanoCA-ACD*. For instance, with 10 BSs, PanoCA-ACD and PanoCA-H improve the TAWA of the CALM-discovered heuristic by 37.90% and 28.16%, respectively. Similarly, with 5 BSs, PanoMC-ACD and PanoMC-H improve the TAWA of the MCTS-discovered heuristic by 44.340% and 37.736%, respectively. In the scenario with 25 BSs, the heuristics discovered by PanoCA-H and PanoMC-

ACD outperform MAPPO by 31.579% and 13.722%, respectively. These results validate the robustness of our solution against variations in the number of BSs.



(a) Impact of BS number. (b) Impact of DT number.

Fig. 6: Impact of the number of BSs and DTs.

Performance with Varying Number of DTs. We construct five datasets, each comprising 20 instances, with the number of DTs set to $5, 10, 15, 20$, and 25, respectively. Note that the unrealistically high numbers of DTs (e.g., 25) are intentionally used to stress test our framework. As illustrated in Fig. 6a, the performance of all methods generally declines as the number of DTs increases, likely due to the heightened demand for channel resources required to support additional DTs. *Despite this trend, the heuristics discovered by our proposed method consistently outperform all baseline methods across the majority of datasets*. For instance, with 10 DTs, PanoMC-ACD improves upon the MCTS-discovered heuristic by 55.172%. In the scenario with 15 DTs, PanoCA-ACD outperforms the CALM-discovered heuristic by 57.416%. Furthermore, with 25 DTs, heuristics discovered by *Panorama*-based methods surpass MAPPO by margins ranging from 22.254% (PanoMC-ACD) to 35.211% (PanoCA-ACD). Collectively, these results validate the robustness of our method against variations in the number of DTs.

Generalizability. Next, we demonstrate the generalizability of our method by testing whether the heuristics trained with data from one city perform well in a different city or at different times of the same city. Recall that when preparing the training dataset, we use vehicle trajectories from the forenoon hours of November 2020 in Shenzhen [36]. To test generalizability, we construct four test datasets with trajectories sampled during the forenoon or afternoon hours of Oct-2022 in Jinan and Apr-2021 in Shenzhen, allowing evaluation on unseen contexts.

Results in Fig. 7 show that *most trends observed in the overall evaluation still hold*. For instance, our *Panorama*-based methods consistently identify the best heuristics: PanoMC-ACD for all Jinan-based datasets and PanoCA-H for all Shenzhen-based datasets. Furthermore, integrating *panorama* empowers baselines like CALM and MCTS to discover superior heuristics. The improvement is particularly pronounced for MCTS. On the "AM Oct-2022, Jinan" dataset, the heuristic discovered by MCTS underperforms MAPPO by 233.333%. In contrast, after enhancement with our *Panorama* framework, the heuristic designed by PanoMC-ACD outperforms MAPPO by 26.316%. This suggests *strong generalization capability of the heuristic generated by our framework*.

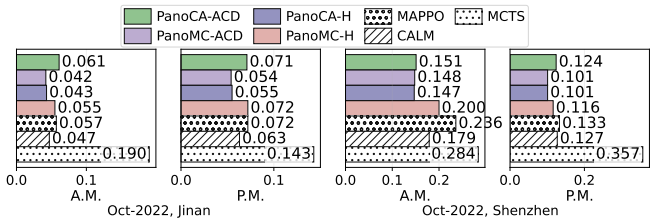


Fig. 7: Performance of methods on the unseen city and dates.

Human-Designed *Panorama* vs. ACD. Human-designed panorama (Section IV-D) *facilitates robust heuristic discovery*. As shown in Fig. 3, while PanoCA-H does not achieve the highest TAWA, it achieves the best top-3 winning ratio. ACD-generated *panorama* (Section IV-E) *automates design and yields efficient algorithms*. Two ACD-guided heuristics outperform all baselines (Fig. 3) and dominate in four out of five DT configurations (Fig. 6b). In all DT and BS settings, at least one ACD-guided heuristic ranks in the top two. However, PanoCA-ACD underperforms CALM in the Jinan scenario (Fig. 7), suggesting potential overfitting.

Notably, PanoMC-ACD exhibits a performance advantage emerging after approximately 1,000 search steps (Fig. 4). This "warm-up" phase is likely because PanoMC-ACD needs to accumulate sufficient environmental feedback to refine initial critics into highly effective ones.

VI. Conclusion

This paper introduces a novel LLM-driven multi-agent automatic heuristic design (AHD) framework that significantly advances the methodological foundation for solving the complex distributed subchannel allocation problem in Digital Twin-Empowered Vehicular Edge Computing (DTVEC) systems. At the core of our contribution is a new AHD paradigm that leverages the interpretability and adaptability of large language models (LLMs) to enable dynamic, context-aware heuristic evolution in multi-agent environments, a capability that existing methods lack. Our framework incorporates a comprehensive diagnostic signal mechanism tailored to this problem and introduces a novel Automatic Critic Design (ACD) method to enhance the efficiency and effectiveness of heuristic search. Comprehensive evaluation demonstrates that this framework achieves consistently superior performance compared to both learning-based and manually designed baselines. In addition to introducing a brand-new data freshness metric, Age of Fusible Information (AoFI), this work establishes a versatile and generalizable LLM-based AHD methodology that opens new opportunities for tackling a wide range of dynamic, distributed network optimization problems.

REFERENCES

[1] J. Chen, W. Wang, B. Fang, Y. Liu, K. Yu, V. C. Leung, and X. Hu, "Digital twin empowered wireless healthcare monitoring for smart home," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 11, pp. 3662–3676, 2023.

[2] Y. Hui, X. Ma, Z. Su, N. Cheng, Z. Yin, T. H. Luan, and Y. Chen, "Collaboration as a service: Digital-twin-enabled collaborative and distributed autonomous driving," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18607–18619, 2022.

[3] Y. Dai and Y. Zhang, "Adaptive digital twin for vehicular edge computing and networks," *Journal of Communications and Information Networks*, vol. 7, no. 1, pp. 48–59, 2022.

[4] D. Kuhse, N. Holscher, M. Gunzel, H. Teper, G. Von Der Bruggen, J.-J. Chen, and C.-C. Lin, "Sync or sink? the robustness of sensor fusion against temporal misalignment," in *2024 IEEE 30th Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pp. 122–134, IEEE, 2024.

[5] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. Springer Science & Business Media, 2012.

[6] H. Zhang, C. Lu, H. Tang, X. Wei, L. Liang, L. Cheng, W. Ding, and Z. Han, "Mean-field-aided multiagent reinforcement learning for resource allocation in vehicular networks," *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2667–2679, 2022.

[7] R. Rauch, Z. Becvar, P. Mach, and J. Gazda, "Cooperative multi-agent deep reinforcement learning for dynamic task execution and resource allocation in vehicular edge computing," *IEEE Transactions on Vehicular Technology*, 2024.

[8] F. Liu, T. Xialiang, M. Yuan, X. Lin, F. Luo, Z. Wang, Z. Lu, and Q. Zhang, "Evolution of heuristics: Towards efficient automatic algorithm design using large language model," in *International Conference on Machine Learning*, pp. 32201–32223, PMLR, 2024.

[9] A. Novikov, N. Vũ, M. Eisenberger, E. Dupont, P.-S. Huang, A. Z. Wagner, S. Shirobokov, B. Kozlovskii, F. J. R. Ruiz, A. Mehrabian, M. P. Kumar, A. See, S. Chaudhuri, G. Holland, A. Davies, S. Nowozin, P. Kohli, and M. Balog, "AlphaEvolve: A coding agent for scientific and algorithmic discovery," tech. rep., Google DeepMind, 05 2025.

[10] Z. Zheng, Z. Xie, Z. Wang, and B. Hooi, "Monte carlo tree search for comprehensive exploration in llm-based automatic heuristic design," in *International Conference on Machine Learning*, PMLR, 2025.

[11] Z. Huang, W. Wu, K. Wu, J. Wang, and W.-B. Lee, "Calm: Co-evolution of algorithms and language model for automatic heuristic design," *arXiv preprint arXiv:2505.12285*, 2025.

[12] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1183–1210, 2021.

[13] J. Li, S. Guo, W. Liang, J. Wu, Q. Chen, Z. Xu, W. Xu, and J. Wang, "Wait for fresh data? digital twin empowered iot services in edge computing," in *2023 IEEE 20th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, pp. 397–405, IEEE, 2023.

[14] S. Chandra, R. Arya, R. Sharma, K. Yadav, M. Gandor, *et al.*, "Freshness-aware device-to-device communication in digital twin network for disaster management," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2025.

[15] H. Liao, Z. Zhou, Z. Jia, Y. Shu, M. Tariq, J. Rodriguez, and V. Frascolla, "Ultra-low aoi digital twin-assisted resource allocation for multi-mode power iot in distribution grid energy management," *IEEE Journal on Selected Areas in Communications*, 2023.

[16] Q. He, G. Dan, and V. Fodor, "Minimizing age of correlated information for wireless camera networks," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 547–552, IEEE, 2018.

[17] Y.-H. Chiang, S. Wakisaka, C. Zhu, H. Lin, and Y. Ji, "Age-efficient concurrent information update scheduling in edge-native systems," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 893–897, 2022.

[18] Y.-H. Chiang, H. Lin, and Y. Ji, "Information cofreshness-aware grant assignment and transmission scheduling for internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 19, pp. 14435–14446, 2021.

[19] X. Xie and H. Wang, "Minimizing age of usage information for capturing freshness and usability of correlated data in edge computing enabled iot systems," *IEEE Transactions on Mobile Computing*, vol. 23, no. 5, pp. 5644–5659, 2023.

[20] J. Liang, T.-T. Chan, and H. Pan, "Minimizing age of collection for multiple access in wireless industrial internet of things," *IEEE Internet of Things Journal*, vol. 11, no. 2, pp. 2753–2766, 2023.

[21] Y. Wang, Y. Pei, and Y. Zhao, "Ts-eoh: An edge server task scheduling algorithm based on evolution of heuristic," in *2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA)*, pp. 693–700, IEEE, 2024.

[22] D. Wu, X. Wang, Y. Qiao, Z. Wang, J. Jiang, S. Cui, and F. Wang, "Netllm: Adapting large language models for networking," in *Proceedings of the ACM SIGCOMM 2024 Conference*, pp. 661–678, 2024.

[23] C. Liu, X. Xie, X. Zhang, and Y. Cui, "Large language models for networking: Workflow, advances and challenges," *IEEE Network*, 2024.

[24] X. Liang, W. Liang, Z. Xu, Y. Zhang, and X. Jia, "Multiple service model refreshments in digital twin-empowered edge computing," *IEEE Transactions on Services Computing*, 2023.

[25] M. K. C. Shisher and Y. Sun, "How does data freshness affect real-time supervised learning?," in *Proceedings of the Twenty-Third International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, pp. 31–40, 2022.

[26] M. Shaygan, C. Meese, W. Li, X. G. Zhao, and M. Nejad, "Traffic prediction using artificial intelligence: Review of recent advances and emerging opportunities," *Transportation Research Part C: Emerging Technologies*, vol. 145, p. 103921, 2022.

[27] Y. Ullah, M. B. Roslee, S. M. Mitani, S. A. Khan, and M. H. Jusoh, "A survey on handover and mobility management in 5g hetnets: current state, challenges, and future directions," *Sensors*, vol. 23, no. 11, p. 5081, 2023.

[28] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 856–868, 2018.

[29] H. Jiang, X. Dai, Z. Xiao, and A. Iyengar, "Joint task offloading and resource allocation for energy-constrained mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 4000–4015, 2022.

[30] L. Liang, S. Xie, G. Y. Li, Z. Ding, and X. Yu, "Graph-based resource sharing in vehicular communication," *IEEE Transactions on Wireless Communications*, vol. 17, no. 7, pp. 4579–4592, 2018.

[31] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492–7508, 2017.

[32] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of ppo in cooperative multi-agent games," *Advances in neural information processing systems*, vol. 35, pp. 24611–24624, 2022.

[33] J. W. Tukey *et al.*, *Exploratory data analysis*, vol. 2. Springer, 1977.

[34] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The non-stochastic multiarmed bandit problem," *SIAM journal on computing*, vol. 32, no. 1, pp. 48–77, 2002.

[35] X. Zhang, M. Peng, S. Yan, and Y. Sun, "Deep-reinforcement-learning-based mode selection and resource allocation for cellular v2x communications," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6380–6391, 2019.

[36] F. Yu, H. Yan, R. Chen, G. Zhang, Y. Liu, M. Chen, and Y. Li, "City-scale vehicle trajectory data from traffic camera videos," *Scientific data*, vol. 10, no. 1, p. 711, 2023.

[37] S. Mitra, S. Ranu, V. Kolar, A. Telang, A. Bhattacharya, R. Kokku, and S. Raghavan, "Trajectory aware macro-cell planning for mobile users," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, pp. 792–800, IEEE, 2015.

[38] S. Mitra, P. Saraf, R. Sharma, A. Bhattacharya, S. Ranu, and H. Bhandari, "Netclus: A scalable framework for locating top-k sites for placement of trajectory-aware services," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, pp. 87–90, IEEE, 2017.

[39] S. Maaloul, H. Aniss, L. Mendiboure, and M. Berbineau, "Performance analysis of existing its technologies: Evaluation and coexistence," *Sensors*, vol. 22, no. 24, p. 9570, 2022.

[40] A. Sharma, "Openevolve: an open-source evolutionary coding agent," 2025.

[41] Y. Özcan and C. Rosenberg, "Uplink scheduling in multi-cell ofdma networks: A comprehensive study," *IEEE Transactions on Mobile Computing*, vol. 20, no. 10, pp. 3081–3098, 2020.