2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Combined Path and Trajectory Planning for Energy-Efficient Coordination of Automated Vehicles in Confined Areas

Stefan Kojchev[1], Jonas Hellgren[2], Robert Hult[3] and Jonas Fredriksson[4]

***Abstract*— In this paper, we present a framework for combined path and motion trajectory planning for the purpose of coordinating fully automated vehicles in confined sites. The path planning component utilizes a Monte-Carlo tree search approach for computing the vehicle paths and the motion trajectory component utilizes a two-stage optimization-based algorithm that optimizes the state and input trajectories for all vehicles while avoiding inter-vehicle conflicts. The motion trajectories are tracked by a low-level controller and both the path and motion trajectories are recomputed based on the feedback signals. The performance of the framework is validated through numerical simulations and results show both improved energy efficiency and productivity.**

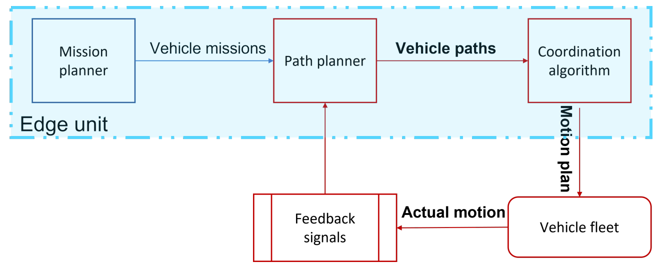


Fig. 1. Architecture of the fleet motion planning system. This paper focuses on the components in red.

I. INTRODUCTION

Ports, logistic centers, mines, etc., are examples of confined sites where a known amount of vehicles and/or machines are required to accomplish a site-specific goal, for example, extracting the desired amount of ore per day or unloading ship containers as fast and energy efficient as possible. To reduce CO2 emissions, the site operators request to replace internal combustion engine-driven vehicles with electric-driven vehicles. However, to retain productivity, more electric vehicles would be required, due to limitations such as range and towing capacity. Furthermore, to reduce cost and the fact that more vehicles are sharing the same space, it is beneficial to automate the operation of the vehicles with the goal of improving site productivity. Due to the enclosed nature of confined sites and the possibility of having no unsupervised actors on the site, the deployment of fully automated vehicles is eased in comparison to public roads [1].

The challenge within confined spaces revolves around efficiently utilizing machinery, vehicles, and road networks to meet productivity requirements. This entails making various crucial decisions, including task allocation and scheduling (determining which vehicle performs which task and the order of task execution), route planning (assigning specific routes to each vehicle), and motion planning (how the vehicles should move). Attempting to solve all tasks in one problem would lead to an intractable large-scale computational problem. A common approach to deal with such problems is to decompose them into multiple tractable sub-problems. Figure 1 shows a proposed decomposition and system architecture. The *Mission planner* component is responsible for assigning each available vehicle a transport mission. A transport mission provides a high-level description of the objectives to be achieved and the required control points to be visited, such as loading/unloading zones and for electric vehicles also charging stations. For example, Vehicle 1 might be tasked with loading a specific amount of materials from point A, while Vehicle 2 should recharge at charging station C. The *Path planner* then determines the optimal routes or paths for the vehicles, ensuring that they adhere to the specified control points and road network. Using these routes, the *Coordination algorithm* calculates the state and input trajectories for all vehicles, with the goal of avoiding conflicts between vehicles and ensuring efficient use of the control points. The motion trajectories of all vehicles are combined into a motion plan, which is communicated to each vehicle in the *Vehicle fleet*. The vehicles try to follow the plan and provide feedback on its execution.

This paper tackles the combined closed-loop path planning and vehicle coordination problem. Typically literature covers one of these problems. For example, [2] proposed a two-level algorithm for optimal pathfinding, where the high-level component performs a tree search based on conflicts between individual agents which impose constraints on the motion of the agents. The low-level component then performs a search to find the optimal path satisfying the constraints. The authors in [3] utilize reinforcement learning for cooperative path planning for multi-vehicle systems. However, this method makes the path planning decisions without relying on a model, which can be detrimental to decisions such as charging. In this paper, we utilize a Monte-Carlo Tree Search

*This work is partially funded by Sweden's innovation agency Vinnova, project number: 2018-02708.

[1]Stefan Kojchev is with Volvo Autonomous Solutions and the Mechatronics Group, Systems and Control, Chalmers University of Technology `stefan.kojchev@volvo.com; kojchev@chalmers.se`
[2]Jonas Hellgren is with Research Institutes of Sweden (RISE), 41258, Göteborg, Sweden `jonas.hellgren@ri.se`
[3]Robert Hult is with Volvo Autonomous Solutions, 41873, Göteborg, Sweden `robert.hult@volvo.com`
[4]Jonas Fredriksson is with the Mechatronics Group, Systems and Control, Chalmers University of Technology, 41296, Göteborg, Sweden `jonas.fredriksson@chalmers.se`

(MCTS) algorithm for centrally computing the vehicle paths such that the mission plan is respected while aiming at minimizing standstill time and avoiding undesirable vehicle behavior such as a critical state of charge. This means that the algorithm is tasked with computing the paths and making decisions about which vehicle utilizes which loading/unloading/charging zone in the confined site.

The coordination problem on the other hand has received significant attention in public road scenarios, such as the coordination of vehicles in intersections [4], [5]. Common approaches leverage Model Predictive Control (MPC) [6], [7], consensus-based methods [8], [9], direct optimal control [10], [11], and trajectory optimization methods [12], [13], for solving the problem. The confined site coordination problem has been addressed as part of our previous work in [14], [15], [16], [17]. In this paper, we leverage the method developed in [16] which uses a two-stage optimization-based approach for computing the motion profiles of the vehicles such that inter-vehicle conflicts are avoided while minimizing the mission time which is related to productivity and minimizing the consumed energy. The optimization-based control algorithm is adapted to the choice of a path-planning algorithm and integrated into the closed-loop system.

In this paper, we proposed a framework that combines path planning and vehicle coordination. The motion plan from the coordination component is followed by a low-level controller that provides actuation commands for all vehicles and represents the motion of the vehicle fleet. The components receive feedback from the fleet and are capable of re-planning accordingly. The authors in [18] propose an approach for the combined path and coordination planning at intersections and the authors in [19] focus on the routing problem with consideration of interactions between the AVs and their movements. The path planning and coordination algorithms in this paper differ from [18] and [19], as [18] relies on a specific choice of vehicle model for the coordination of the AVs, while [19] estimated travel speeds inversely proportional to the number of AVs on the road. Furthermore, the algorithms proposed in this paper are adapted to the confined site case, which also explores other problems such as loading and unloading the vehicles and the charging process, which to the best of our knowledge has not been addressed in the literature to date. The main benefit of the approaches presented in this paper is that they are not bounded by a specific choice of vehicle model and are easily extendable to different confined sites. Moreover, the approaches are capable of optimizing for the goal of improving energy efficiency and productivity of the vehicles, which are two vital factors in the deployment of automated vehicles in confined sites.

The framework is evaluated with respect to an approach with a simpler planning strategy in a simulation scenario of a representable confined site. The simulation demonstrates how the proposed approach can improve performance when dealing with a typical mission within confined sites.

II. PROBLEM STATEMENT

In this paper, we consider a fully confined area, meaning that non-controlled traffic participants such as pedestrians, manually operated vehicles, bicycles, etc., are absent. Furthermore, we assume that the site mission plan, which determines the site production goal and how many vehicles are required, has been computed and communicated to the Path planner component. Consequently, we have N_a fully automated vehicles operating in the site on paths that include cross-intersections, path merges, path splits, narrow roads, and dwelling zones to achieve the total production goal. In addition, we assume that overtakes are prohibited and that no vehicle reverses.

A. Path Planning

The path planner refines the abstract site mission into explicit paths for every vehicle while observing the current state of the traffic environment and the vehicle fleet. From the observation, the path planner gives commands, primarily the next destinations, to the vehicles. In essence, the path planner is required to primarily respect the control points (positions that a vehicle must pass) that are communicated by the mission planner components. In order to improve performance, the path planner is further tasked to minimize waiting time and ensure that the vehicles do not form a deadlock. A deadlock can occur when a vehicle requires access to an area that is already occupied by another vehicle.

The above-stated requirements and constraints can be summarized as

***Problem** 1:* (Optimal path planning problem) Obtain the vehicle paths that respect the given mission goal by solving

$$\begin{aligned} \min_{t} \quad & \sum_{i=1}^{N_a} J_{\text{vehicle},i}(t) + \epsilon \quad (€/s) && \text{(1a)}\\ \text{s.t.} \quad & \text{deadlock constraints}, \ \forall i && \text{(1b)}\\ & \text{state of charge}, \ \forall i, && \text{(1c)} \end{aligned}$$

where $J_{\text{vehicle},i}$ is the individual cost of each vehicle. The term ϵ is a parameter related to the constraint violation that becomes a large positive number if any of the stated constraints are violated, otherwise, it is equal to zero. The constraints ensure the algorithm does not produce vehicle paths that are undesired and/or unsafe.

B. Coordination of Automated Vehicles

Utilizing the computed vehicle paths we can form the vehicle coordination problem into an Optimal Control Problem with the task of obtaining the optimal state and input trajectories such that conflicts between the vehicles are avoided. The problem can be stated as:

***Problem** 2:* (Optimal coordination problem) Obtain the optimal state and control trajectories $\mathcal{X}^* = \{x_1^*, ..., x_{N_a}^*\}$, $\mathcal{U}^* = \{u_1^*, ..., u_{N_a}^*\}$, given the initial state $\mathcal{X}_0 =$

$\{x_{1,0}, ..., x_{N_a,0}\}$, by solving the optimization problem

$$\min_{x_i, u_i, \mathcal{O}^{\text{MUTEX}}} \sum_{i=1}^{N_a} J_{\text{ca},i}\left(x_i, u_i\right) \tag{2a}$$

$$\text{s.t} \quad \text{initial states} \;\; x_{i,0} = \hat{x}_{i,0}, \forall i \tag{2b}$$

$$\text{system dynamics} \;\; \forall i \tag{2c}$$

$$\text{state and input constraints} \;\; \forall i \tag{2d}$$

$$\text{safety constraints} \;\; \forall i \tag{2e}$$

where $J_{\text{ca},i}(x_i, u_i)$ is the cost function for the coordination algorithm, $\mathcal{O}^{\text{MUTEX}}$, groups the sets of occupancy orders in which the vehicles enter the MUTually EXclusive zones (MUTEX) zones and will be further explained in this section.

The problem is formulated in the spatial domain as it is beneficial to optimize the trajectories of the vehicles over their full paths. The rationale for using spatial dynamics is that the time it takes for the vehicle to traverse a path is not known *a-priori*. Thus, it is inappropriate to plan the vehicle's motion with time as the independent variable.

1) System dynamics and state and input constraints: The system dynamics for vehicle $i \in 1, ..., N_a$ in the spatial domain can be formed using that $\frac{\mathrm{d}p_i}{\mathrm{d}t} = v_i(t)$ and $\mathrm{d}t = \mathrm{d}p_i/v_i(t)$ and stated as

$$\frac{\mathrm{d}t_i}{\mathrm{d}p_i} = \frac{1}{v_i(p_i)} \tag{3}$$

$$\frac{\mathrm{d}x_i}{\mathrm{d}p_i} = \frac{1}{v_i(p_i)} f_i(p_i, x_i(p_i), u_i(p_i)) \tag{4}$$

$$0 \leq h(p_i, x_i, u_i). \tag{5}$$

where $p_i \in \mathbb{R}$ is the vehicle's position, $x_i \in \mathbb{R}^n$ is the vehicle state, $u_i \in \mathbb{R}^m$ is the control input, with $i \in \{1, \ldots, N_a\}$. The state is subdivided as $x_i = (v_i, z_i)$, with the speed along the path $v_i \in \mathbb{R}$ and $z_i \in \mathbb{R}^{n-1}$ collecting possible other states.Note that what the remaining state variables (z_i) are, depends on what model is used for the system dynamics. We assume that the functions f_i and h_i, which describe the vehicle system's dynamics and constraints, are smooth.

2) Safety constraints: Conflicts between vehicles can occur in MUTEX zones, such as intersections, narrow roads, merge-splits, and dwelling zones as each zone has limited simultaneous vehicle occupancy capacity. Failure to properly account for the capacity of these zones leads to a loss of optimality or conflict.

Intersections and narrow roads: In the intersection and narrow road MUTEX zones, meeting oncoming vehicles is not possible. From a safety perspective, this translates to "reserving" the zone for one or more vehicles coming from the same direction. The vehicles coming from the opposite direction are not allowed to occupy the zone until it is vacated.

Merge-splits: In the merge-split case, two vehicles coming from different roads, but moving in the same direction of travel, join together on a common patch of road. After some distance, the roads separate. For efficiency, it is desirable to have several vehicles in the zone at the same time, instead of blocking the whole zone. This requires imposing rear-end collision constraints once the vehicles have entered the merge-split zone.

Dwelling zones: The dwelling zones consist of a road patch that leads to either a charger or a loading or unloading station and a road patch after the station until a merge point with the remainder of the road. When a vehicle visits a dwelling zone, it is required to make a full stop at the station and after some time, the vehicle leaves the station with either an increased state of charge or increased/decreased vehicle mass. The mission planner assigns the amount of time that each vehicle is supposed to be at the station which corresponds to a certain increase in state of charge or mass change. However, in reality, depending on external factors the vehicles would not always charge or load/unload the exact specified amount. For example, the mission planner could assign that the vehicle should charge for two minutes and thus increase its state of charge by ten percent. However, due to factors such as grid power, the vehicle might not charge exactly ten percent for that time interval. We capture this stochasticity in the dwelling zones in the Vehicle fleet component by adding bounded random noise to the charging, loading, and unloading amount.

Each MUTEX zone corresponds to a specific occupancy order that is required to be computed. The order for all the zones is grouped in the set $\mathcal{O}^{\text{MUTEX}}$. The mathematical details of these constraints are given in [16].

III. METHODS

This section describes the methods that are used for solving the path planning problem, Problem 1, and the vehicle coordination problem, Problem 2.

A. Path Planning Algorithm

The path planner models a site as a directed graph, $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ are the vertex (nodes) and edge sets, respectively. A node can have itself as a destination if the node is a dwelling zone, i.e., a charging, loading/unloading zone. An example of a directed graph that is investigated in this paper is depicted in Figure 2. A primary property of the nodes in the path planner is that the time distance in between is constant, where the time between two nodes is in the order of seconds. Relatively long time steps between the nodes heavily speed up the path planning analysis, as the time horizon is typically required to be in the order of minutes. This is a necessary trade-off between improved performance due to a longer horizon and tractable computation corresponding to the time steps between the nodes.

Using the site graph, the path planner is required for each vehicle to compute a path such that a desired performance objective is achieved considering the capabilities of the vehicles. In this paper, the performance objective aims at minimizing the standstill time meaning that we can define the individual vehicle cost $J_{\text{vehicle},i}$ as

$$J_{\text{vehicle},i}(t) = \begin{cases} 0 & \text{(is absorbed)} \\ \frac{1}{t_{h,i}}(c_{\text{still}} t_{\text{still},i} + c_{\text{ch}} t_{\text{ch},i}) & \text{(else)} \end{cases} \tag{6}$$

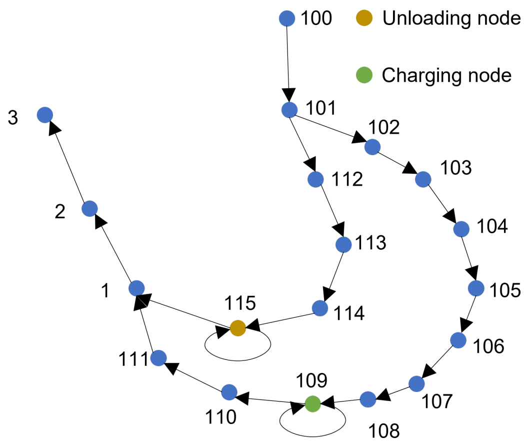


Fig. 2. Directed graph example including two dwelling zones.

where $t_{\text{still},i}$, $t_{\text{ch},i}$, and $t_{h,i}$ denote the standstill time, the charging time for vehicle i, and the search horizon in seconds, respectively. The terms c_{still} and c_{ch} are related to the cost of standing still and the charging cost and are valued in euros per second. Note that the cost per vehicle is zero in the case when the vehicle is in a dwelling zone, i.e., the vehicle is absorbed in a node. The intuition behind this choice is that the routing problem would not penalize necessary stops at these zones.

The path planner requires that the vehicle states in the controlled fleet are the present and the destination nodes, the state of charge, and the current absorption time. The present and destination nodes inform the planner of the current and the intended positions, the state of charge represents the present battery energy level and the current absorption time is correlated to the dwelling zones. Specifically, the absorption time only increases when the vehicle is in one of the dwelling zones and is necessary for the planner to determine when the vehicle would need to leave the dwelling zone. Consequently, the absorption time is equal to zero outside the dwelling zones.

The evolution of these states, can, for example, be modeled according to a longitudinal vehicle dynamics model. However, the approach is not specifically restricted by the choice of model.

1) Monte-Carlo tree search: An approach that is capable of handling path planning problems of the above-defined form is the Monte-Carlo Tree Search (MCTS) [20], [21]. MCTS is a widely adopted decision-making algorithm in the field of artificial intelligence and game theory and originates from the field of computer board games, such as chess and Go. MCTS is a heuristic search algorithm designed to explore large decision spaces efficiently and is characterized by its ability to balance exploration and exploitation, making it suitable for path-planning problems.

The MCTS consists of four major steps, namely, selection, expansion, simulation, and backpropagation. Based on a selection strategy, the algorithm starting from the root node iteratively selects child nodes. In the next step, the selected node is expanded to create one or more child nodes representing possible actions or states. Following the expansion, a simulation is performed from one of the newly expanded nodes, often following a randomized policy or heuristics. Finally, the results of the simulation are backpropagated through the tree to update node statistics, such as visit counts and estimated values. In this paper, we utilize an adapted tree-step expansion and a selfish default policy. The adapted tree-step expansion method only expands with more than one child node to the selected leaf node if a vehicle is located on a split node or if a vehicle is entering a conflict zone where a deadlock could occur. The restricted expansion significantly reduces the tree branching. A selfish default policy is a policy that minimizes the waiting time for the individual vehicle with respect to its constraints, i.e., it is the locally optimal solution for the vehicle without considering the effect on other vehicles.

Adapting MCTS to path planning for autonomous vehicles involves representing the problem as a search tree, where a tree node represents a set of vehicle states and every edge in the tree reflects a cost, as defined in eq. (1a). As the MCTS steps are repeated the search tree grows. Once a termination condition is met, for example after a certain amount of time, the path that minimizes the cost can be extracted and forwarded to the Coordination algorithm component. The tree size, and hence the computational cost, increases with the branching of the tree.

B. Decomposition Strategy for the Coordination of AVs

For the coordination of AVs in confined sites we employ the framework presented in [16].

The safety constraints in Problem 2 correspond to a specific MUTEX occupancy order that is required to be computed. Thus, the optimal vehicle coordination problem can be stated as a Mixed Integer Nonlinear Program (MINLP), where the crossing order corresponds to the “integer part” and the state and control trajectories corresponds to the “NLP part”. However, finding a solution to MINLP problems is known to be difficult, especially when the constraints or the objective function are non-convex [22]. Therefore, a common procedure is to apply an approach where the integer part of the solution is obtained first using a heuristic, and the continuous part of the solution thereafter is obtained by solving the NLP that results from fixing the integers to the values found with the heuristic.

The approach presented in [16] approximates the integer part of the solution of the coordination problem by solving a Mixed Integer Quadratic Problem (MIQP) resulting in an optimal crossing order for the approximated problem. The MIQP is formed as a quadratic approximation of the coordination problem, similarly to how QP sub-problems are formed in SQP methods [23]. Following the MIQP, we solve a “fixed-order coordination” NLP, for obtaining the state and control trajectories. Fig. 3 depicts the control structure of the heuristic. The proposed MIQP has an exponential complexity in the decision variables (i.e., amount of MUTEX zones) and the NLP has a cubic complexity in the number of stages, i.e.,

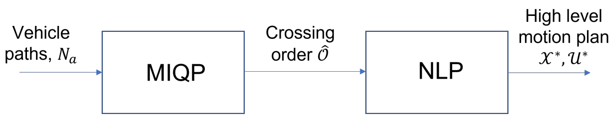


Fig. 3. Optimal coordination of vehicles heuristic structure

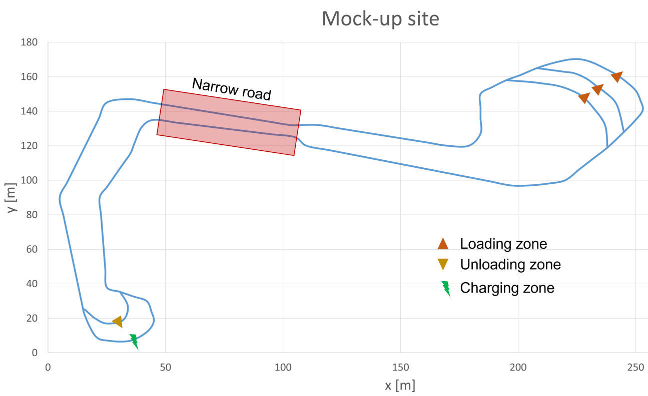


Fig. 4. Mock-up confined site.

the sum of vehicles and discretization points per vehicle. The specifics of the decomposition strategy are given in [16].

IV. SIMULATIONS

In this section, we present a simulation example where the combined path planning and vehicle coordination is demonstrated and evaluated with respect to a baseline solution.

A. Simulation Setup

For the simulation scenario, we consider a mock-up confined site with a layout shown in Figure 4. The site has three loading stations, one unloading station, and one station where the vehicles can unload while simultaneously charging. Furthermore, there is one narrow road on the site and multiple merge-split zones. The loading process requires 30 seconds for loading one tonne of goods, while the unloading process requires 10 seconds for unloading one tonne of goods. The vehicles are charged for 120 seconds, while they unload one tonne of goods and get a 10 % increase in their state of charge. We investigate the case when there are in total five vehicles operating on the site and the mission for the site is to continuously transfer load between the loading and unloading stations. The vehicles that are used in this simulation represent the Volvo TARA machine (TA15). To represent the fully electric machine, we use a model derived using Newton's second law of motion which can be stated as:

$$\dot{p}_i(t) = v_i(t), \tag{7a}$$

$$\dot{v}_i(t) = \frac{1}{m_i}\left(F_{M,i}(t) - F_{d,i}(v_i,t) - F_{rg,i}(p_i,t)\right) \tag{7b}$$

$$\dot{SOC}_i(t) = \frac{-P_{b,i}\left(v_i, F_{M,i}\right)}{E_{b,\max,i}}, \tag{7c}$$

where p_i, v_i, m_i and $E_{b,\max,i}$ are vehicle i's position, velocity, mass, and battery energy, respectively. The forces are the motor force $F_{M,i}$, the force generated by the friction brakes $F_{b,i}$, the aerodynamic drag $F_{d,i}$, and the rolling resistance and gravitational force $F_{rg,i}$. The aerodynamic drag and the rolling resistance and gravitational force can be described as

TABLE I

SIMULATION PARAMETERS IN SI UNITS.

Parameter	N_a	m	ρ	A	c_a
Value	5	20000	1.18	10	1
Parameter	c_r	$\overline{P}_b$	$\underline{P}_b$	$R_{\text{int},i}$	$N_{\text{cells},i}$
Value	0.01	240 (kW)	240 (kW)	0.001	200
Parameter	$K_{t,i}$	$E_{b,\max}$	r_w	M_f	$\underline{v}$
Value	5	13.6 (kWh)	0.536	49.16	0.1
Parameter	$\overline{v}$	$\underline{a}_{\text{lon}}$	$\overline{a}_{\text{lon}}$	$\overline{a}_{\text{lat}}$	T_{a_x}
Value	6.94	-2	2	2	0.5
Parameter	$\Delta\underline{v}$	$\Delta\overline{v}$	$\Delta\underline{p}$	$\Delta\underline{p}$	$\Delta\underline{SOC}$
Value	-0.1	0.1	-0.5	0.5	-0.01
Parameter	$\Delta\overline{SOC}$	$\Delta\underline{m}$	$\Delta\overline{m}$	t_h	
value	0.01	-100	100	30	

$$F_{d,i}(v_i,t) = \frac{1}{2}\rho A_i c_{a,i} v_i(t)^2 \tag{8a}$$

$$F_{rg,i}(p_i,t) = m_i g(sin(\theta(p_i(t))) + c_{r,i}cos(\theta(p_i(t)))), \tag{8b}$$

where ρ is the air density, A_i is the frontal area of the vehicle, $c_{a,i}$ is the aerodynamic drag coefficient, $c_{r,i}$ is the rolling resistance coefficient and θ is the road gradient.

We model the battery as an ideal voltage source with internal resistance. We thus define battery power as the difference between the product of velocity and motor force and the internal battery losses due to the resistance as

$$P_{b,i}\left(v_i, F_{M,i}\right) = F_{M,i}(t)v_i(t) - P_{\text{loss}}(t) \tag{9}$$

$$\underline{P}_{b,i} \leq P_{b,i} \leq \overline{P}_{b,i}, \tag{10}$$

with $\underline{P}_{b,i}, \overline{P}_{b,i}$ representing the power limits, and

$$P_{\text{loss}}(t) = \frac{R_{\text{int,i}} N_{\text{cells,i}}}{K_{t,i}^2} T_{M,i}^2(t), \tag{11}$$

where $R_{\text{int,i}}$ is the internal resistance of the battery, $N_{\text{cells,i}}$ is the number of cells of the battery, $K_{t,i}$ represents the electric machine's torque constant and $T_{M,i}$ is the motor torque defined as

$$T_{M,i}(t) = \frac{r_{w,i} F_{M,i}(t)}{M_{f,i}}, \tag{12}$$

where $r_{w,i}$ is the wheel radius and $M_{f,i}$ is the transmission's final gear ratio.

The Coordination Algorithm is defined in space and thus this model is converted into the spatial domain for its use in the component, as done in [16]. Along with the defined model, the coordination algorithm bounds the velocity and acceleration of the vehicles as

$$\underline{v}_i \leq v_i(p_i) \leq \overline{v}_i \tag{13}$$

$$\underline{a}_{i,\text{lon}} \leq a_i(p_i) \leq \overline{a}_{i,\text{lon}}, \tag{14}$$

$$\left(\frac{a_i(p_i)}{\overline{a}_{i,\text{lon}}}\right)^2 + \left(\frac{\kappa_i(p_i)v_i(p_i)^2}{\overline{a}_{i,lat}}\right)^2 \leq 1, \tag{15}$$

where the vehicle position, p_i is now the independent variable, a_i represents the vehicle's acceleration and the constraint in eq. (15) is resulting from the curvature of the road.

The objective function for this component consists of minimizing the power of the battery, the squares of the acceleration, and the end time. The first term is related to energy-efficient driving while the second term is related to driving comfort and component wear. The last term "motivates" the vehicles to arrive at their end destination as fast as possible, i.e., it affects to mission end time. In this paper, the objective weight parameters related to the minimization of the power of the battery and the acceleration are equal to one and the objective weight parameter related to the minimization of the end time is equal to ten.

The low-level controller in the Vehicle Fleet is set up as a Model Predictive Control (MPC), [24], in the time domain with a prediction and control horizon of five seconds that tracks the position and velocity reference computed in the Coordination Algorithm and uses the model as defined in (7). The control signal from the MPC is sent to a Vehicle Fleet model that deviates from (7) by having an additional lag element, denoted as $T_{a_x,i}$, in the acceleration term to represent the dynamic powertrain time constant. In addition, the position and velocity states of the vehicle fleet are subject to a random noise from a bounded interval, i.e., $\Delta p_i \in [\Delta \underline{p}_i, \Delta \overline{p}_i]$ and, $\Delta v_i \in [\Delta \underline{v}_i, \Delta \overline{v}_i]$. Furthermore, in the dwelling zones, there is additional noise that influences how much the vehicle is charged or the loading/unloading amount, which can be denoted as $\Delta SOC_i \in [\Delta \underline{SOC}_i, \Delta \overline{SOC}_i]$, and $\Delta m_i \in [\Delta \underline{m}_i, \Delta \overline{m}_i]$. In this way, we attempt to bridge the simulation closer to reality, where the position and velocity noise is related to sensor and measurement inaccuracies, while the noise related to the charging process is related to non-linear charging, and the noise in the loading/unloading amount is related to the inaccuracies while loading and unloading. The additive noise and model uncertainty parameters along with the rest of the simulation parameters are summarized in Table I.

The exchange of information between the components is done through a local server from which the components are able to post and get information. This communication is set up via a REST protocol [25]. The Path planner posts the paths of all vehicles to the Coordination algorithm and gets from the Vehicle Fleet the current states and if the vehicle is in an absorption node the dwelling time. The Coordination Algorithm also gets the current states from the Vehicle Fleet. The path plan is recomputed every second, while the motion plan is recomputed either every eight seconds or when there is a significant difference in the tracking of the plan or if the recomputed path plan goes to a different dwelling zone. Note that by recomputing the path plan, the "dynamic" cases such as when a new vehicle enters the site operations or when a path is blocked can be effectively managed by the Path planning component and consequently by the Coordination algorithm. The Path planner is implemented in Java and the Vehicle Fleet and the Coordination algorithm

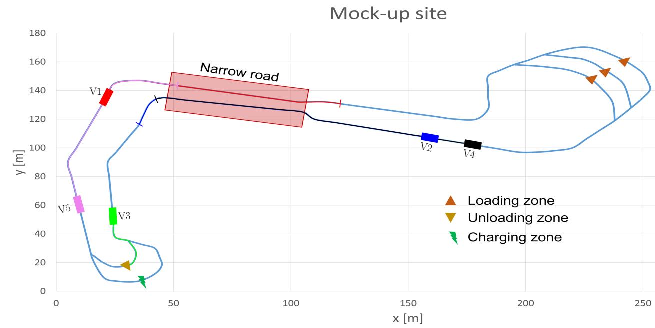


Fig. 5. One instance of the vehicle paths.

are implemented in MATLAB and utilize Gurobi for solving the MIQP and IPOPT [26] for solving the NLPs.

B. Baseline

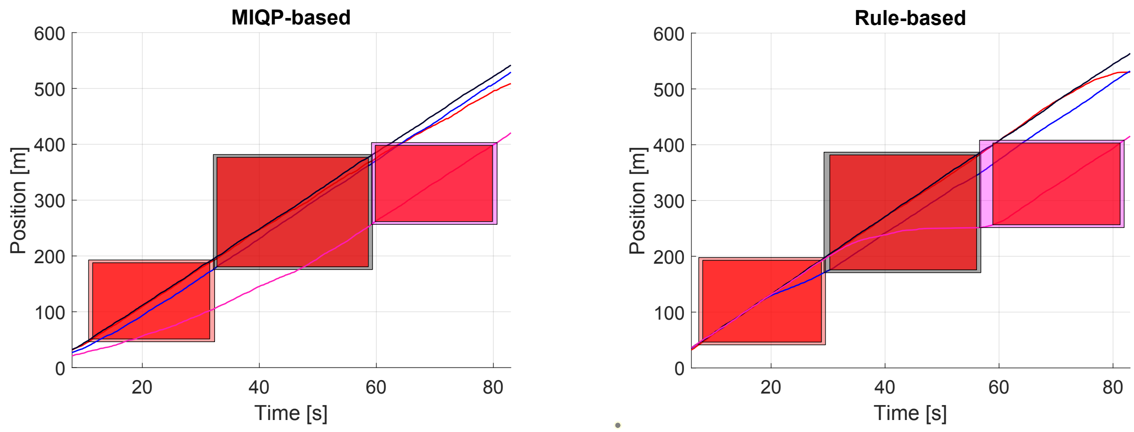
We compare the proposed approach with a rule-based motion planning strategy that has the same path planner component as stated in this paper. This allows us to focus on the benefit of the MIQP-based coordination algorithm.

In practical applications, rule-based strategies are frequently employed because they can be formulated without the need for optimization. While these approaches are relatively straightforward to implement, they encounter challenges related to optimality, scalability, and adaptability. Essentially, to enhance performance, the rules need to be adjusted for each unique use case, such as different confined site scenarios.

In this paper, the rule-based approach is structured in a way that involves progressing all vehicles through time, and specific rules for MUTEX zone occupancy are developed. These rules grant access to the zone to the first vehicle entering an area surrounding the zone. The area around the zone includes an additional 20-meter margin from the entry point to allow later-arriving vehicles to stop if necessary. To control each vehicle's behavior over time, we employ an MPC scheme. The MPC is structured such that the cost function is the same as the one used in the Coordination algorithm and uses the same dynamics and state and input constraints, but formulated in the time domain. For the MUTEX zone occupancy constraints, the MPC is also subject to position constraints that are imposed at specific time steps. The MPC is created with a ten-second prediction and control horizon. The MPC prediction horizon and the additional margin added to the zone are sufficient for a vehicle to make a full stop from $\overline{v}_i$.

C. Results

To demonstrate the benefit of the MIQP-based heuristic we investigate the closed-loop motion profiles for a specific scenario when multiple vehicles are approaching the narrow road zone. The specific scenario is depicted in Figure 5. Furthermore, we investigate the performance benefit in terms of total consumed energy and mission end time for when the vehicles are required to complete twenty laps of the



(a) Zone occupancy using the MIQP-based heuristic

(b) Zone occupancy using the rule-based approach

Fig. 6. Zone occupancy of the narrow road with the different heuristics. The red zones depict the conflict zone occupancy for the vehicles. A collision occurs if the red zones intersect.

site (corresponding to each vehicle transporting 20 tons of goods).

Figure 6 depicts the occupancy of the narrow road zone for the interacting vehicles using the MIQP-based heuristic and the rule-based baseline approach. The motion profiles are recorded from the Vehicle fleet and thus include the measurement noise and model mismatch. To account for the noise the safety constraints related to the zone occupancy are tightened by extending the zone which is a common approach, [11]. In this paper, we extend the zones by an additional five meters. The red rectangles are obtained from the entry and exit times of the vehicles in the zone, and these rectangles must not intersect with each other, otherwise, the vehicles have entered the zone while another vehicle has still not left the zone. Both approaches manage to avoid occupancy conflicts, however, as it can be noticed in Figure 7 there is a significant difference in how that is accomplished. The MIQP-based heuristic leverages the full available path plan to make adequate long-term changes. The effect of the full path planning is particularly visible for the fifth vehicle, where the MIQP-based heuristic avoids stopping and queuing the vehicle. Note that Vehicle 2 and Vehicle 4 are on the same road, i.e., a merge-split zone, and are subject to rear-end collision constraints that are also not violated. We omit depicting the speed profile for Vehicle 3 as that vehicle is not passing through the narrow road for the investigated scenario.

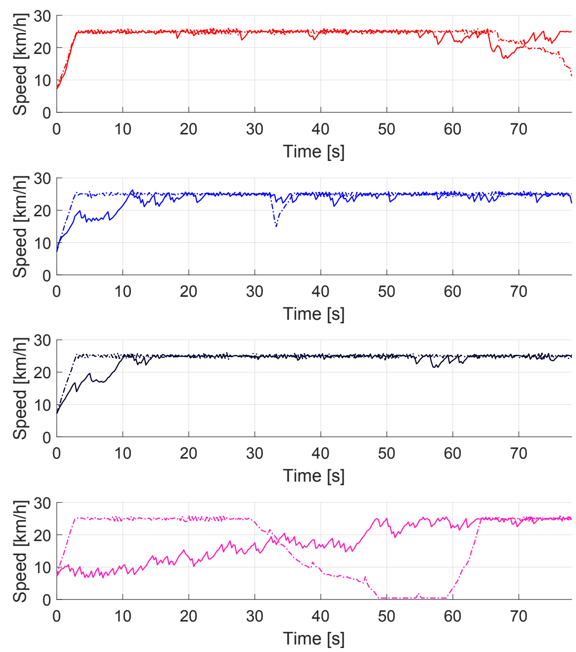


Fig. 7. Speed profiles for the narrow road crossing resulting from the MIQP-based heuristic (solid lines) and the rule-based approach (dashed lines). The colors in the figure correspond to the vehicles indicated in Figure 5. In order to satisfy the safety constraints the rule-based approach has to queue the fifth vehicle.

Utilizing this capability and overall determining the occupancy order in the zones based on an approximation of the complete problem, the MIQP-based heuristic also demonstrates improved energy consumption. Specifically, for the investigated site mission when every vehicle is required to transport 20 tonnes of material, the MIQP-based heuristic results in a 5.41 % improvement of the total energy consumption with respect to the rule-based approach. The mission end time for the MIQP-based heuristic is 6885 seconds, whereas, for the rule-based heuristic, all vehicles have completed their missions after 6911 seconds. Note that different sites will lead to different values of these metrics.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a framework for combined path and motion planning for automated vehicles in confined sites. The approaches do not depend on a specific choice of motion model, constraints, and objective, which allows for the framework to be easily adapted to different sites with different objectives and operating vehicles. The path planning component is tasked to reduce stand still times which are related to productivity, while to motion planning component is tasked with obtaining energy-efficient motion profiles for the vehicles that can be followed by the vehicle fleet. The performance of the framework was analyzed through closed-loop simulations and compared with respect to an alternative strategy, where a benefit in terms of energy efficiency is demonstrated.

One improvement to the path planning approach is to include a long-term memory which can be obtained using a learning-based method to assist in making charging decisions. Experimental validations of the approach also present a natural extension of the presented framework alongside refining the modeling of the battery within the path planning component.

REFERENCES

[1] Fagnant, Daniel J., and Kara Kockelman. "Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations." Transportation Research Part A: Policy and Practice 77 (2015): 167-181.

[2] Sharon, Guni, Roni Stern, Ariel Felner, and Nathan R. Sturtevant. "Conflict-based search for optimal multi-agent pathfinding." Artificial Intelligence 219 (2015): 40-66.

[3] Wang, Qichen, and Chris Phillips. "Cooperative path-planning for multi-vehicle systems." electronics 3, no. 4 (2014): 636-660.

[4] Rios-Torres, Jackeline, and Andreas A. Malikopoulos. "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps." IEEE Transactions on Intelligent Transportation Systems 18, no. 5 (2016): 1066-1077.

[5] Mariani, Stefano, Giacomo Cabri, and Franco Zambonelli. "Coordination of autonomous vehicles: taxonomy and survey." ACM Computing Surveys (CSUR) 54, no. 1 (2021): 1-33.

[6] Kneissl, Maximilian, Adam Molin, Hasan Esen, and Sandra Hirche. "A feasible MPC-based negotiation algorithm for automated intersection crossing." In 2018 european control conference (ecc), pp. 1282-1288. IEEE, 2018.

[7] Riegger, Lea, Markus Carlander, Niklas Lidander, Nikolce Murgovski, and Jonas Sjöberg. "Centralized MPC for autonomous intersection crossing." In 2016 IEEE 19th international conference on intelligent transportation systems (ITSC), pp. 1372-1377. IEEE, 2016.

[8] Mirheli, Amir, Mehrdad Tajalli, Leila Hajibabai, and Ali Hajbabaie. "A consensus-based distributed trajectory control in a signal-free intersection." Transportation research part C: emerging technologies 100 (2019): 161-176.

[9] Wuthishuwong, Chairit, and Ansgar Traechtler. "Consensus-based local information coordination for the networked control of the autonomous intersection management." Complex & Intelligent Systems 3, no. 1 (2017): 17-32.

[10] Hult, Robert, Mario Zanon, Sébastien Gros, Henk Wymeersch, and Paolo Falcone. "Optimisation-based coordination of connected, automated vehicles at intersections." Vehicle System Dynamics 58, no. 5 (2020): 726-747.

[11] Hult, Robert, Mario Zanon, Sebastien Gros, and Paolo Falcone. "Optimal coordination of automated vehicles at intersections: Theory and experiments." IEEE Transactions on Control Systems Technology 27, no. 6 (2018): 2510-2525.

[12] Yu, Chunhui, Yiheng Feng, Henry X. Liu, Wanjing Ma, and Xiaoguang Yang. "Corridor level cooperative trajectory optimization with connected and automated vehicles." Transportation Research Part C: Emerging Technologies 105 (2019): 405-421.

[13] Hu, Xiangwang, and Jian Sun. "Trajectory optimization of connected and autonomous vehicles at a multilane freeway merging area." Transportation Research Part C: Emerging Technologies 101 (2019): 111-125.

[14] Kojchev, Stefan, Robert Hult, and Jonas Fredriksson. "Optimization based coordination of autonomous vehicles in confined areas." In 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), pp. 1957-1963. IEEE, 2022.

[15] Kojchev, Stefan, Robert Hult, and Jonas Fredriksson. "Quadratic approximation based heuristic for optimization-based coordination of automated vehicles in confined areas." In 2022 IEEE 61st Conference on Decision and Control (CDC), pp. 6156-6162. IEEE, 2022.

[16] Kojchev, Stefan, Robert Hult, Jonas Fredriksson, and Maximilian Kneissl. "A Two-Stage MIQP-Based Optimization Approach for Coordinating Automated Electric Vehicles in Confined Sites." IEEE Transactions on Intelligent Transportation Systems (2023).

[17] Kojchev, Stefan, Robert Hult, and Jonas Fredriksson. "Energy Efficient Optimization-Based Coordination of Electric Automated Vehicles in Confined Areas." In 2023 62nd IEEE Conference on Decision and Control (CDC), pp. 3433-3440. IEEE, 2023.

[18] Bang, Heeseung, and Andreas A. Malikopoulos. "Re-Routing Strategy of Connected and Automated Vehicles Considering Coordination at Intersections." In 2023 American Control Conference (ACC), pp. 4419-4424. IEEE, 2023.

[19] Mostafizi, Alireza, Charles Koll, and Haizhong Wang. "A decentralized and coordinated routing algorithm for connected and autonomous vehicles." IEEE Transactions on Intelligent Transportation Systems 23, no. 8 (2021): 11505-11517.

[20] Kocsis, Levente, and Csaba Szepesvári. "Bandit based monte-carlo planning." In European conference on machine learning, pp. 282-293. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[21] C. B. Browne et al., "A Survey of Monte Carlo Tree Search Methods," in IEEE Transactions on Computational Intelligence and AI in Games, vol. 4, no. 1, pp. 1-43, March 2012, doi: 10.1109/TCIAIG.2012.2186810.

[22] Richards, Arthur, and Jonathan How. "Mixed-integer programming for control." In Proceedings of the 2005, American Control Conference, 2005., pp. 2676-2683. IEEE, 2005.

[23] Nocedal, Jorge, and Stephen Wright. Numerical optimization. Springer Science & Business Media, 2006.

[24] Rawlings James Blake David Q Mayne and Moritz Diehl. 2017. Model Predictive Control: Theory Computation and Design. 2nd ed. Madison Wisconsin: Nob Hill Publishing.

[25] Fielding, Roy Thomas. Architectural styles and the design of network-based software architectures. University of California, Irvine, 2000.

[26] Wächter, Andreas, and Lorenz T. Biegler. "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming." Mathematical programming 106, no. 1 (2006): 25-57.