Intermittent Cooperation in Path Planning for Selfish Agents

Noa Agmon
Bar-Ilan University
Ramat-Gan, Israel
agmon@cs.biu.ac.il

Itay Shedlezki
Bar-Ilan University
Ramat-Gan, Israel
itay.shedlezki@biu.ac.il

ABSTRACT

Multi-robot systems often involve autonomous agents navigating shared environments toward individual goals. In such environments, the spatial and temporal structure of the task may create opportunities for intermittent, localized cooperation, such as jointly opening a heavy gate, synchronizing to pass a traffic light as a convoy, or temporarily forming a vehicle platoon to reduce traversal time. Such cooperation is often opportunistic and fragile: robots remain self-interested, coordination is local and temporary, and deviation is always possible. We study this setting for two agents through the *Intermittent Cooperation-Based Two-Agent Path Planning* (IC2PP) problem, a shortest-path game on graphs in which agents navigate toward individual targets while optionally cooperating at specific nodes to reduce their own travel times. Although such cooperation can strictly benefit both agents, it is strategically fragile: agents may deviate at any point along their paths. While a joint strategy in which each agent follows its naive shortest path without considering cooperation may not necessarily form a Pure Nash Equilibrium (PNE), we characterize the structure of PNE joint strategies in IC2PP, and show that stable cooperation must follow a highly constrained form. We further prove that at least one PNE exists in every instance of IC2PP, and present a polynomial-time algorithm for enumerating all relevant PNEs. When multiple equilibria arise, we study coordination mechanisms based on bargaining-theoretic selection concepts and empirically compare equilibrium outcomes in terms of individual travel times and social welfare.

KEYWORDS

Self-Interested Agents, Two-Player Games, Graph Navigation, Non-cooperative games, Motion and path planning

ACM Reference Format:
Noa Agmon and Itay Shedlezki. 2026. Intermittent Cooperation in Path Planning for Selfish Agents. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 25 pages.

1 INTRODUCTION

Consider the following real-world scenario: two autonomous robots navigate from their respective start locations to separate task destinations. Each can independently follow an individually shortest path, yet under certain circumstances they may benefit from coordinating their routes. For instance, jointly opening a heavy gate, synchronizing to pass a traffic light as a convoy, or temporarily forming a vehicle platoon may reduce their traversal times, incentivizing deviation from individually optimal paths in favor of cooperation. Such coordination, however, is inherently precarious and limited to specific locations and times. It is feasible and beneficial only if both agents arrive sufficiently close in time to enable synchronization, and neither agent may gain by deviating (e.g. by initiating cooperation earlier or later, or by leaving prematurely once its individual benefit diminishes). As a result, cooperation that appears advantageous in principle may fail in practice. This creates a fundamental tension: cooperation can strictly improve individual outcomes, but only if it is precisely timed and strategically stable. Although cooperation opportunities are local and time-dependent, agents commit to complete paths in advance, and deviations at any point along a path can undermine coordination.

In this work, we show that despite this apparent fragility, cooperation at equilibrium is possible, and precisely because of it, such cooperation must follow a highly structured form. We study this phenomenon through the **Intermittent Cooperation-Based Two-Agent Path Planning (IC2PP)** problem, a two-player game on a graph modeling navigation in a shared environment in which self-interested agents navigate from individual start nodes to individual targets, seeking to minimize their arrival times while cooperation opportunities in the graph can be leveraged to improve their outcomes. These cooperation opportunities may undermine strategic stability, as they allow agents to improve their path times, and consequently agents following their shortest independent paths do not necessarily form a Nash equilibrium, nor is this outcome necessarily advantageous. We fully characterize the Pure Nash Equilibria (PNE) of this game and show that any cooperative equilibrium must follow a rigid structure: a single contiguous cooperation segment, preceded and followed by non-cooperative paths, with cooperation beginning and ending only at nodes *jointly stable* against unilateral deviation. This structural characterization has two key implications. First, a PNE always exists. Second, equilibrium computation becomes tractable: for each cooperation node, there exists at most one non-dominated PNE ending cooperation at that node, meaning that no other PNE yields strictly lower travel times for both agents. Leveraging these insights, we present a polynomial-time algorithm that enumerates all non-dominated equilibrium joint strategies.

Finally, when multiple equilibria arise, they reflect different trade-offs between the agents' individual benefits. We examine how agents may select among such equilibria using standard bargaining-based solution concepts for decentralized coordination, and empirically evaluate how cooperation opportunities and path alignment influence equilibrium efficiency.

2 RELATED WORK

IC2PP is related to a diverse set of two-player games, as well as multi-robot and multi-agent problems that arise across markedly

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus.

different domains, each with distinct assumptions, objectives and interaction models.

Several games on graphs study strategic interactions in spatial settings, including Stackelberg security games [13, 18, 49] and assignment-based models such as the dinner party problem [5, 8, 11, 16], and topological distance games [14, 20]. However, they typically focus on static assignments or partitions and do not address path-based strategies, where decisions consist of sequences of nodes subject to spatial constraints and hence may involve intermittent dynamics in practice.

Coalition formation research [4, 6, 23] studies how agents form cooperative groups to pursue mutual benefits, typically focusing on stable partitions or maximizing global utility. Dynamic coalition formation [2, 31, 38, 48] extends this line of work to settings in which collaboration can evolve over time, but it overlooks trajectory coordination along paths or the planning of sequences of short-term, intermittent cooperation.

Task allocation problems [33] address efficient execution of tasks by agents focusing on optimizing a global utility. For example, in Coalition Formation with Spatial and Temporal Constraints (CF-STP) [42] agents are assigned to time-critical tasks across locations, yet this framework assumes fully cooperative agents maximizing collective utility, rather than self-interested agents forming temporary coalitions for individual objectives. Ad-Hoc Teamwork [45] refers to the problem of enabling agents to collaborate without prior coordination, focusing on adaptive behaviors and adjustment skills developed through training and experience. However, this line of work also focuses on fully cooperative settings.

Multi-Agent Path Finding (MAPF), Autonomous Intersection Management (AIM), and congestion games typically model shared locations as detrimental phenomena to be avoided. MAPF focuses on collision-free planning, usually under centralized or fully cooperative assumptions [9, 30, 44], and extensions with task dependencies or cooperative requirements likewise assume fully cooperative agents [12]. Congestion games capture competition over shared resources whose costs increase with usage [24, 25, 29, 47]. AIM plans vehicle trajectories to avoid conflicts at intersections [15, 21, 28], including distributed methods [15], multi-intersection settings [28], and vehicle platooning [7, 34, 36], where coordinated arrivals improve efficiency. However, these typically assume fully cooperative behavior. In contrast, our work models nodes as opportunities for strategic coordination among self-interested agents, introducing a fundamentally different game-theoretic structure.

Cost-sharing games study how agents split the cost of jointly used network resources, focusing on equilibrium existence and efficiency under fixed cost-allocation mechanisms [1, 27]. While this line of work captures strategic incentives in network design, it assumes static participation in shared resources and does not consider path-based strategies in which agents endogenously decide when to cooperate and when to act independently along their trajectories, as in our setting.

Research on urban mobility and shared transportation [10, 17, 19, 26, 37, 43, 50] primarily focuses on coordination mechanisms for ride-sharing and fleet management, often aiming to balance operational efficiency with social welfare. These models typically abstract away from agent-level strategic path choices and do not consider cooperation as a decision made by self-interested agents. While existing work, such as [22, 32], considers cooperation switching with one or multiple hops, they focus on matching drivers and riders under centralized or algorithmic coordination, rather than modeling the strategic decision-making process of self-interested agents from a game-theoretic perspective.

3 IC2PP: PROBLEM DEFINITION

We formalize the **Intermittent Cooperation-Based Two-Agent Path Planning (IC2PP) problem** as follows:

> We consider two self-interested agents simultaneously navigating a shared graph-based environment from their respective source to target nodes, where cooperation at specific interaction points may reduce the delays they incur. Each agent seeks to find a path minimizing its individual travel time. Their strategic interaction is modeled as a two-player game, and the goal is to identify, characterize, and compare resulting Pure Nash Equilibrium (PNE) outcomes.

Consider an environment with two self-interested agents, a_1 and a_2, simultaneously navigating an undirected graph $G = (V, E)$ representing a shared physical environment. Each agent a_i starts at its initial node $s_i \in V$, aiming to reach its target node $g_i \in V$ as early as possible. Each edge $(v, u) \in E$ has an associated travel time $\tau_{vu} \geq 0$. Each node $v \in V$ may impose a travel delay on an agent visiting it, representing, for example, the time required to perform a local task (e.g., opening a gate) before proceeding. The travel delay incurred by a single agent passing alone at node v is τ_v^1, whereas cooperation may reduce it to τ_v^2 ($\tau_v^1 \geq \tau_v^2 \geq 0$). The set $V_C = \{v \in V \mid \tau_v^2 < \tau_v^1\}$ includes nodes in which cooperation strictly reduces delay (referred to as *cooperation nodes*).

If agent a_1 (w.l.o.g.) arrives at a cooperation node $c \in V_C$ at time t_1 before agent a_2 arrives at time t_2, cooperation requires the earlier agent to wait until the other arrives. As a result, if a_1 waits for a_2, its total delay at c is $(t_2 - t_1) + \tau_c^2$. In this case, agent a_2 incurs a total delay of τ_c^2. Otherwise, both agents incur the non-cooperative delay τ_c^1. Consequently, an agent's actual delay at a node depends on both its own arrival time and that of the other agent.

A path π is a sequence of nodes connected by edges. For any two nodes v, u along π, let $\pi_{v,u}$ denote the subpath from v to u. The *path time* $T(\pi^1 \mid \pi^2)$, formally defined in A.1, denotes the total travel time of an agent along π^1, given that the other agent follows π^2. This includes edge traversal times and vertex delays induced by synchronization and cooperation, excluding delays at the first and last nodes. $T(\pi)$ denotes the independent travel time of an agent along π, assuming the agent traverses all nodes alone, without cooperation.

The *shortest independent path* ($SIP_{v,u}$) denotes the shortest path between nodes v and u, assuming the traversing agent does not coordinate with the other agent at any node along the path. The *shortest cooperation path* ($SCP_{v,u}$) denotes the shortest path assuming the traversing agent cooperates at all cooperation nodes without waiting for the other agent, incurring delay τ_w^2 at each node $w \in SCP_{v,u}$.

A strategy for an agent is defined as the full path from its starting node to its target. We consider a strategic setting in which each agent selects a full-path strategy while being aware of the other agent's start and goal nodes, aiming to minimize its individual path time. A *joint strategy* is a pair of paths (π^1, π^2) specifying the strategies chosen by agents a_1 and a_2, respectively. A joint strategy is a Pure Nash Equilibrium (PNE) if neither agent can unilaterally deviate so as to strictly improve its individual travel time.

4 PNE PROPERTIES IN IC2PP

We seek to identify all joint strategies constituting a PNE: joint-paths where no agent has an incentive to unilaterally deviate. When considering a joint strategy in which neither agent's path includes any cooperation nodes, verifying whether the strategy constitutes a PNE is straightforward: it suffices to check that each agent follows its shortest independent path, which can be done using a standard shortest-path algorithm. However, when cooperation nodes are part of the path, the situation becomes more complex, as these cooperation opportunities may incentivize the agents to deviate from their paths in order to improve their arrival times. If there are $m > 0$ cooperation nodes, there may exist up to 2^m distinct cooperation patterns. Consequently, determining whether a joint strategy that involves cooperation opportunities forms a PNE is nontrivial. This raises several structural questions: *when should cooperation begin, what form should it take, and when should it end?* We show that cooperation-involving equilibrium strategies exhibit strong structural properties that sharply restrict these possibilities.

We divide our analysis into two classes of joint strategies: those that involve cooperation and those that do not. For each class, we characterize the structural conditions required for a joint strategy to constitute a PNE. We begin with joint strategies that involve cooperation. These can be partitioned into three distinct segments:
(1) Joining Segment: Each agent independently approaches the *cooperation starting node* c_s, where cooperation begins.
(2) Cooperation Segment: The agents intermittently cooperate at a subset of cooperation nodes from c_s to a later node c_d, referred to as the *cooperation departure node*.
(3) Departure Segment: From c_d onward, each agent follows an independent path to its target node.

In the following sections, we analyze each of these segments and formulate a set of structural conditions that must hold to prevent unilateral deviations[1] and ensure the joint strategy constitutes a PNE.

4.1 Joining segment

In this segment, each agent follows an independent path, involving no cooperation, toward the cooperation starting node c_s. In Appendix B.1, we prove the following Lemma:

LEMMA 1 (EARLY COOPERATION). *Each agent prefers to initiate cooperation at the earliest possible cooperation node along the other agent's path.*

Following Lemma 1, one might expect both agents to prefer initiating cooperation at the earliest cooperation node that is mutually reachable. However, doing so may enable one agent to exploit the other by leaving the cooperation prematurely, thereby imposing an inferior path time on the remaining agent. Anticipating such behavior, the other agent may instead prefer to avoid cooperation nodes that admit exploitative deviations, even if this delays the onset of cooperation. Figure 1 illustrates such a scenario.

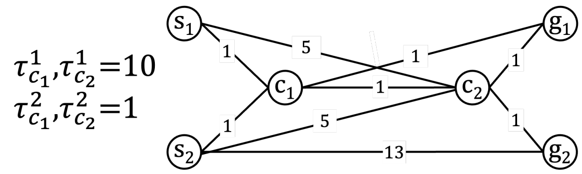


Figure 1: Although c_1 is the earliest cooperation node reachable by both agents, cooperation there allows agent a_1 to deviate and leave agent a_2 with an inferior outcome, leading a_2 to avoid c_1.

To ensure that neither agent has an incentive to deviate from its intended path during the joining segment, we explicitly consider two types of potential deviations:
(1) **Cooperation deviation:** If agent a_1's (w.l.o.g.) path includes a cooperation node reachable by agent a_2 within a relevant time for cooperation, then by the Early Cooperation Lemma, agent a_2 would prefer to deviate from its intended in order to initiate cooperation at that earlier node.
(2) **Arrival time deviation:** If the last-arriving agent at c_s can adjust its path to arrive earlier (thereby enabling cooperation to begin sooner) this adjustment benefits both agents.

DEFINITION 1 (NON-COOPERATIVE PARTIAL PATH). *A partial path $\pi_{s_i,v}$ from agent a_i's starting node s_i to a node $v \in V$ is defined as a* Non-Cooperative (NC) Partial Path *if the first cooperation node along the path that the other agent, a_{-i}, can reach from its starting node s_{-i} within relevant time for cooperation is v. Formally, for every cooperation node $c \in \pi_{s_i,v} \setminus \{v\}$, $T(SIP_{s_{-i},c}) > T(\pi_{s_i,c}) + \tau^1_c - \tau^2_c$.*

We denote the set of all Non-Cooperative Partial Paths from the starting node s_i to node v as $\Pi^{NC}_{s_i,v}$, and the *shortest* Non-Cooperative Partial Path from s_i to node v as $SIP^{NC}_{s_i,v}$. The shortest NC partial paths from a start node s_i to all nodes can be efficiently computed using a slight adaptation of Dijkstra's algorithm (Algorithm 1). If a cooperation node $c \in V_C$ is reachable by the other agent at a time suitable for cooperation, it is removed from the graph once its shortest non-cooperative partial path is identified, preventing further expansions.

Algorithm 1 SHORTEST NC PARTIAL PATHS$(G, V_C, s_i, SIP_{s_{-i}})$

$T_{s_i,v} \leftarrow \infty, \pi_{s_i,v} \leftarrow \emptyset$ for all $v \in V$
$T_{s_i,s_i} \leftarrow 0, \pi_{s_i,s_i} \leftarrow s_i$
$Q \leftarrow V$
while Q has a node v s.t. $T_{s_i,v} < \infty$ **do**
  $v \leftarrow$ node in Q with smallest $T_{s_i,v}$
  Remove v from Q
  if $v \notin V_C$ or $SIP_{s_{-i},v} > T_{s_i,v} + \tau^1_v - \tau^2_v$ **then**
    for each neighbor u of v **do**
      if $Ts_i, v + \tau^1_v + \tau_{v,u} < Ts_i, u$ **then**
        $Ts_i, u \leftarrow Ts_i, v + \tau^1_v + \tau_{v,u}$
        $\pi_{s_i,u} \leftarrow \pi_{s_i,v} \circ u$
Return a dictionary from each $v \in V$ to $\pi_{s_i,v}$

[1]Although described in dynamic terms for clarity, deviations in our model correspond to ex ante selection of an alternative full-path.

Appendix B.2 includes a line-by-line walkthrough of the algorithm.

Definition 2 (Mutually-Robust Non-Cooperative Partial Paths). *Two non-cooperative partial paths, $\pi^1_{s_1,v}$ and $\pi^2_{s_2,v}$, are defined as* Mutually-Robust Non-Cooperative Partial Paths *if neither agent can improve the cooperation starting time at v by unilaterally changing its path toward it. Formally, one of the following conditions holds: (1) Both agents arrive at node v simultaneously: $T(\pi^1_{s_1,v}) = T(\pi^2_{s_2,v})$. (2) If agent a_1 (w.l.o.g.) reaches v first, then a_2 arrives at v via its shortest independent path, i.e., $T(\pi^2_{s_2,v}) = T(SIP_{s_2,v})$.*

Following Definitions 1 and 2, to eliminate unilateral deviations within the joining segment, the agents' *joining segments* must form Mutually-Robust NC Partial Paths.

4.2 Cooperation segment

In this segment, an agent may deviate by seeking additional cooperation opportunities or by avoiding cooperation at certain nodes. In Appendix C.1, we prove the following:

Lemma 2 (Cooperation Continuity). *Consider two cooperation nodes. If both agents cooperate between them, then neither agent can strictly improve its path time by deviating to an individual path and rejoining later.*

Following Lemma 2, to prevent unilateral deviations along the cooperation segment, both agents must follow a shared continuous path from the cooperation starting node to the cooperation departure node. However, note that this does *not* mean that both agents should simply take the shortest cooperative path to the cooperation departure node c_d (SCP_{c_s,c_d}), since the agents differ in their individual objectives, thus their preferred departure nodes may not coincide. As a result, an agent may have an incentive to deviate by exiting cooperation prematurely and proceeding independently toward its target. If an agent benefits from leaving cooperation before c_d, such a deviation violates the Nash equilibrium conditions. Moreover, anticipating this, the other agent, may decide to depart even earlier to optimize its own departure node. This process could repeat multiple times, potentially resulting in suboptimal outcomes for both agents (see example in Figure 2). Alternatively, an agent may enforce a detour within the cooperation segment to avoid nodes at which the other agent could deviate prematurely, thereby preventing outcomes that would result in an inferior path time. Figure 3 illustrates such a scenario. Therefore we make the following vital definition:

Definition 3 (Stable Cooperation Partial Path). *A cooperation partial path π_{c_s,c_d} between cooperation nodes $c_s, c_d \in V_C$ is a* stable cooperation partial path *if the departure node c_d is the optimal departure node for both agents along the path. Formally, let $v^i_{d^*}(\pi_{c_s,c_d})$ denote the optimal departure node for agent a_i along $\pi_{c_s,c_d} \in \Pi_{c_s,c_d}$, defined as*

$$v^i_{d^*}(\pi_{c_s,c_d}) = \arg\min_{c \in \pi_{c_s,c_d}} T\left(\pi_{c_s,c} \circ SIP_{c,g_i} \mid \pi_{c_s,c_d} \circ SIP_{c_d,g_{-i}}\right)$$

then, $c_d = v^1_{d^*}(\pi_{c_s,c_d}) = v^2_{d^*}(\pi_{c_s,c_d})$.

This definition implies that once cooperation begins at node c_s, both agents would prefer to continue cooperating along π_{c_s,c_d} rather than deviating earlier towards their respective target nodes. The set of all stable cooperation partial paths between two nodes v and u is denoted by $\Pi^S_{v,u}$, and the *shortest* stable cooperation partial path between these nodes is denoted by $SCP^S_{v,u}$.

The stability of a partial path between two cooperation nodes c_1 and c_2 depends solely on the path times from c_1 onward and is independent of the path segment leading to c_1. Leveraging this property, the shortest stable cooperation partial paths ending at a cooperation node $c \in V_C$ can be computed efficiently via a backward variant of Dijkstra's algorithm (Algorithm 2) which explores the graph backward from c while excluding nodes that violate stability. Appendix C.2 includes a line-by-line walkthrough of the algorithm.

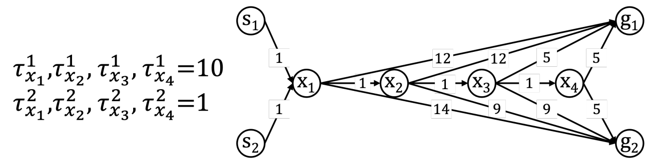


Figure 2: The optimal departure node for a_2 along the path x_1, x_2, x_3, x_4 is x_4, while for a_1, it is x_3. Along the path x_1, x_2, x_3, the optimal departure node for a_2 is x_2, and for the path x_1, x_2, the optimal departure node for a_1 is x_1. Thus, the only PNE in this scenario is the path containing only x_1.

Algorithm 2 Shortest Stable Paths($G, V_C, c_d, SIP_{g_1}, SIP_{g_2}$)

$T_{v,c_d} \leftarrow \infty, \pi_{v,c_d} \leftarrow \emptyset$ for all $v \in V$
$T_{c_d,c_d} \leftarrow 0, \pi_{c_d,c_d} \leftarrow c_d$
$Q \leftarrow V$
while Q has a node v s.t. $T_{v,c_d} < \infty$ **do**
  $v \leftarrow$ node in Q with smallest T_{v,c_d}
  Remove v from Q
  for each neighbor u of v s.t. $u \in Q$ and $\tau_{u,v} + \tau^2_v + T_{v,c_d} + SIP_{c_d,g_1} \le SIP_{u,g_1}$ and $\tau_{u,v} + \tau^2_v + T_{v,c_d} + SIP_{c_d,g_2} \le SIP_{u,g_2}$ **do**
    if $Tv, c_d + \tau^2_v + \tau_{u,v} < Tu, c_d$ **then**
      $Tu, c_d \leftarrow Tv, c_d + \tau^2_v + \tau_{u,v}$
      $\pi_{u,c_d} \leftarrow u \circ \pi_{v,c_d}$
Return a dictionary from each $v \in V$ to π_{v,c_d}

Following Definition 3, to eliminate unilateral deviations within the cooperation segment, this segment must form Stable Cooperation Partial Path.

4.3 Departure segment

In this segment, since cooperation ceases and both agents aim to reach their respective targets as quickly as possible, each agent will necessarily follow its shortest independent path towards its respective target: SIP_{c_d,g_1} and SIP_{c_d,g_2}. However, if the shortest independent path of one agent, (w.l.o.g) a_1, includes a cooperation node c'_d that serves as a better departure point for a_2 than c_d, then a_2 would prefer to continue cooperating along SIP_{c_d,c'_d} instead of taking its independent shortest path directly to its target. This violates the Nash equilibrium conditions. Thus, the cooperation departure node

c_d must also be the optimal exit node for both agents along the departure segment $(SIP_{c_d,g_1}, SIP_{c_d,g_2})$.

4.4 Full Path Equilibrium Cooperation

To conclude our analysis of the cooperation involving joint strategies, we identify five conditions that a cooperative joint strategy must satisfy, proven to be both necessary and sufficient for it to constitute a Nash equilibrium (full proof appears in Appendix E.1). Note that this holds for *cooperation* joint strategies, hence the shortest independent paths joint strategy may also be a PNE.

THEOREM 1. *A cooperation joint strategy (π^1, π^2) constitutes a PNE if and only if all the following conditions hold:*

(1) *The agents cooperate along exactly one cooperation segment, $\pi^S_{c_s,c_d} \in \Pi^S_{c_s,c_d}$, which is stable.*
(2) *Both agents reach the cooperation start node c_s via Mutually-Robust non-cooperative partial paths $\pi^{1(NC)}_{s_1,c_s} \in \Pi^{NC}_{s_1,c_s}$ and $\pi^{2(NC)}_{s_2,c_s} \in \Pi^{NC}_{s_2,c_s}$.*
(3) *The cooperation end node c_d is jointly optimal for departure: $c_d = v^1_{d^*}(\pi^2_{c_s,g_2}) = v^2_{d^*}(\pi^1_{c_s,g_1})$.*
(4) *From c_d onward, both agents follow their shortest independent paths to their targets, SIP_{c_d,g_1} and SIP_{c_d,g_2}.*
(5) *Cooperation weakly dominates the shortest independent path for both agents: $T(\pi^1 \mid \pi^2) \leq T(SIP_{s_1,g_1})$, $T(\pi^2 \mid \pi^1) \leq T(SIP_{s_2,g_2})$.*

Consequently, (π^1, π^2) can be formally expressed as follows:

$$\pi^1 = \pi^{1(NC)}_{s_1,c_s} \circ \pi^S_{c_s,c_d} \circ SIP_{c_d,g_1}, \quad \pi^2 = \pi^{2(NC)}_{s_2,c_s} \circ \pi^S_{c_s,c_d} \circ SIP_{c_d,g_2}$$

Condition 1 ensures the validity of the Cooperation segment, while Condition 2 guarantees the validity of the Joining segment. Conditions 3 and 4 verify the correctness of the Departure segment. Finally, Condition 5 ensures that the entire path is advantageous for both agents, confirming the overall benefit of the cooperation.

DEFINITION 4 (EQUILIBRIUM COOPERATION JOINT STRATEGY). *We refer to a joint strategy satisfying the above conditions as an* Equilibrium Cooperation Joint Strategy *(ECJS), and denote the set of all such strategies by $\mathbb{ECJS}$.*

Further analysis of the ECJS structure yields the insight:

Pareto Sub-optimality: The optimal ECJS in $\mathbb{ECJS}^{c_s,c_d}$ is not necessarily Pareto optimal. Specifically, since $T(SIP^{NC}_{s_i,c_s}) \geq T(SIP_{s_i,c_s})$ and $T(SCP^S_{c_s,c_d} \mid SCP^S_{c_s,c_d}) \geq T(SCP_{c_s,c_d} \mid SCP_{c_s,c_d})$, the full-path joint strategy $\left(SIP^{NC}_{s_1,c_s} \circ SCP^S_{c_s,c_d} \circ SIP_{c_d,g_1}, SIP^{NC}_{s_2,c_s} \circ SCP^S_{c_s,c_d} \circ SIP_{c_d,g_2}\right)$ may be suboptimal compared to $(SIP_{s_1,c_s} \circ SCP_{c_s,c_d} \circ SIP_{c_d,g_1},\ SIP_{s_2,c_s} \circ SCP_{c_s,c_d} \circ SIP_{c_d,g_2})$, indicating that the optimal joint strategy in $\mathbb{ECJS}^{c_d}$ is not necessarily Pareto optimal. Figure 3 illustrates such a scenario. In this example, the game presents three PNE: $((s_1, g_1), (s_2, g_2))$, $((s_1, c_1, c_2, g_1), (s_2, c_1, c_2, g_2))$, and $((s_1, c_1, a, c_3, g_1), (s_2, c_1, a, c_3, g_2))$. Among the three PNE, the joint strategy $((s_1, c_1, a, c_3, g_1), (s_2, c_1, a, c_3, g_2))$ minimizes the path time of a_2, though it remains suboptimal compared to the joint strategy where both agents follow the shortest path between v_{c_1} and v_{c_3}: $((s_1, c_1, c_2, c_3, g_1), (s_2, c_1, c_2, c_3, g_2))$.

4.5 Equilibrium in Independent Joint Strategies

A joint strategy that involves no cooperation may constitute a PNE only if both agents follow their shortest independent paths and no profitable cooperation opportunities exist along either path. A naive validation approach would require checking all possible cooperation opportunities along these paths. However, leveraging Lemmas 1 and 2, it follows that the best response of an agent, a_1 (w.l.o.g), to a fixed strategy π of the other agent, a_2, is either the shortest independent path SIP_{s_1,g_1}, or a cooperative path that initiates cooperation at the earliest reachable cooperation node along π and maintains cooperation until the optimal departure node c_d. Algorithm 3, computes this best-response strategy in polynomial time.

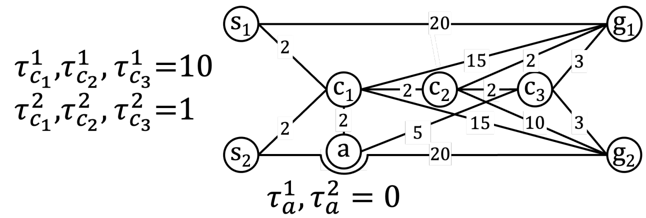


Figure 3: A PNE exists from c_1 to c_3, but not when both agents follow SCP_{c_1,c_3}, since a_1 prefers to deviate at c_2. Equilibrium is obtained when both instead follow c_1, a, c_3.

Algorithm 3 BEST RESPONSE PATH $(G, V_C, s_1, g_1, s_2, g_2, \pi)$

$SIP_{s_1} \leftarrow$ ALL SHORTEST PATHS FROM(G, τ^1, s_1)
$SIP_{g_1} \leftarrow$ ALL SHORTEST PATHS TO(G, τ^1, g_1)
$c^*_1, d \leftarrow None$
for all $c_i \in \pi$ by its order of occurrence along π **do**
  if $SIP_{s_1,c_i} \leq \pi_{s_2,c_i} + \tau^1_{c_1} - \tau^2_{c_i}$ **then**
    $c^*_1 \leftarrow c_i$
    break
if $c^*_1 = None$ **then**
  return SIP_{s_1,g_1}
$optimalPathTime \leftarrow \infty$
$cooperationPathTime \leftarrow 0$
for all $v \in \pi_{c^*_1,g_2}$ by its order of occurrence along π **do**
  $cooperationPathTime \leftarrow cooperationPathTime + \tau^2_v$
  if $cooperationPathTime + SIP_{v,g_1} \leq optimalPathTime$ **then**
    $optimalPathTime \leftarrow cooperationPathTime + SIP_{v,g_1}$
    $d \leftarrow v$
  $v_{next} \leftarrow$ next node in π
  $cooperationPathTime \leftarrow cooperationPathTime + \tau_{v}, v_{next}$
if $SIP_{s_1,c^*_1} + optimalPathTime \leq SIP_{s_1,g_1}$ **then**
  return $SIP_{s_1,c^*_1} \circ \pi_{c^*_1,d} \circ SIP_{d,g_1}$
else
  return SIP_{s_1,g_1}

A full explanation of the algorithm, including a line-by-line walkthrough and a complexity analysis, is provided in Appendix D.1. Applying this algorithm to both agents and comparing the resulting strategies with the shortest independent paths allows us to efficiently verify whether $(SIP_{s_1,g_1},, SIP_{s_2,g_2})$ is a PNE.

[1] In the context of the pseudocode, $SP_{v,u}$ denotes both the shortest path between nodes v and u and the associated path time, depending on the context.

4.6 PNE Joint Strategy Set

We show that the number of ECJS that must be considered is at most linear in $m = |V_C|$, and that this set is guaranteed to be non-empty.

Let $\mathbb{ECJS}^{c_d}$ denote the set of ECJS that end cooperation at $c_d \in V_C$, and let $\mathbb{ECJS}^{c_s,c_d}$ denote those that start cooperation at $c_s \in V_C$ and conclude it at c_d. Since all joint strategies in $\mathbb{ECJS}^{c_d}$ involve the simultaneous departure of the agents from c_d, and their path times from c_d to their respective target nodes remain constant, there exists one (or a few equivalent) joint strategy in $\mathbb{ECJS}^{c_d}$ that arrives earliest at c_d and dominates all others. Accordingly, our approach seeks to identify this optimal joint strategy for each cooperation node $c_d \in V_C$, thereby reducing the number of relevant cooperation paths to be linear in m. This task reduces to identifying the optimal cooperation starting node c_{s^*} and the joint strategy in $\mathbb{ECJS}^{c_{s^*},c_d}$ that minimizes the arrival time at c_d. For a given cooperation starting node c_s, the optimal joint strategy in $\mathbb{ECJS}^{c_s,c_d}$ is obtained when both agents follow their shortest non-cooperative partial paths to c_s ($SIP^{NC}_{s_1,c_s}$, $SIP^{NC}_{s_2,c_s}$), with arrival times synchronized to ensure mutual robustness, and then jointly follow the shortest stable cooperation partial path $SCP^S_{c_s,c_d}$.

Consequently, if $\mathbb{ECJS}^{c_s,c_d} \neq \emptyset$, its optimal element (π^1, π^2) satisfies

$$\pi^i = SIP^{NC}_{s_i,c_s} \circ SCP^S_{c_s,c_d} \circ SIP_{c_d,g_i}, \qquad i \in \{1,2\} \tag{1}$$

and finding the optimal joint strategy in $\mathbb{ECJS}^{c_d}$ reduces to finding the optimal cooperation starting node c^*_s minimizing the arrival time at c_d:

$$c^*_s = \arg\min_{c_s \in V_C} \Big\{ \max\big(T(SIP^{NC}_{s_1,c_s}), T(SIP^{NC}_{s_2,c_s})\big) + \tau^2_{c_s} + T(SCP^S_{c_s,c_d} \mid SCP^S_{c_s,c_d})\Big\} \tag{2}$$

In Section 5, we use this characterization to introduce an efficient algorithm to map all ECJS an agent must consider when determining its strategy by examining all cooperation nodes $c \in V_C$, treating each as a potential cooperation **ending** node and identifying the optimal ECJS concluding cooperation at that node. As shown in Appendix E.2, not all cooperation nodes need to be considered in this process. Specifically, concatenation of two stable paths is stable, and the longer stable path necessarily dominates the shorter. Therefore, partial cooperation paths can be ignored.

The following theorem, proven in Appendix E.3, concludes the insights on the size of the ECJS set:

THEOREM 2. *IC2PP always admits a Pure Nash Equilibrium*

5 MAPPING THE SET OF RELEVANT ECJS

The structural results presented above restrict the equilibrium search to at most one candidate per cooperation departure node. Based on this observation, Algorithm 4 computes the optimal stable joint strategy ending cooperation at a given node. Algorithm 5 enumerates all non-dominated cooperative PNEs by iterating over all possible departure nodes.

5.1 Joint Paths To Cooperation Departure Node

Given a cooperation departure node c_d, Algorithm 4 computes the dominating stable cooperation joint strategy ending cooperation at that node. Appendix F includes a line-by-line walkthrough of the algorithm and its complexity analysis.

Algorithm 4 OPTIMAL STABLE JOINT STRATEGY ENDING COOPERATION AT A GIVEN NODE($G, V_C, c_d, s_1, s_2, g_1, g_2$)

```
SIP_{s1} ← SHORTEST PATHS FROM(G, τ^1, s1)
SIP_{s2} ← SHORTEST PATHS FROM(G, τ^1, s2)
SIP_{g1} ← SHORTEST PATHS TO(G, τ^1, g1)
SIP_{g2} ← SHORTEST PATHS TO(G, τ^1, g2)
SCP^S_{cd} ← SHORTEST STABLE PATHS(G, V_C, c_d, SIP_{g1}, SIP_{g2})
SIP^NC_{s1} ← SHORTEST NC PATHS(G, V_C, s1, SIP_{s2})
SIP^NC_{s2} ← SHORTEST NC PATHS(G, V_C, s2, SIP_{s1})
optNE ← ∞,    c_{s*} ← None
for all c_s ∈ V_C do
    if max(SIP^NC_{s1,cs}, SIP^NC_{s2,cs}) + τ^2_{cs} + SCP^S_{cs,cd} ≤ optNE then
        optNE ← max(SIP^NC_{s1,cs}, SIP^NC_{s2,cs}) + τ^2_{cs} + SCP^S_{cs,cd}
        c_{s*} ← c_s
if optNE + τ^2_{cd} + SIP_{cd,g1} ≤ SIP_{s1,g1} and optNE + τ^2_{cd} + SIP_{cd,g2} ≤ SIP_{s2,g2} then
    return (SIP^NC_{s1,cs*} ∘ SCP^S_{cs*,cd} ∘ SIP_{cd,g1}, SIP^NC_{s2,cs*} ∘ SCP^S_{cs*,cd} ∘ SIP_{cd,g2})
return None
```

LEMMA 3. *Algorithm 4 computes, in polynomial time, the optimal stable joint strategy ending cooperation at c_d.*

The correctness of Lemma 3 follows from the observation that if $\mathbb{ECJS}^{c_d} \neq \emptyset$, the joint strategy presented in Eq. 1 where c_s is computed as in Eq. 2, is dominant in $\mathbb{ECJS}^{c_d}$. As established in Appendix F, the algorithm's time complexity is $O(|E| + |V|\log|V|)$.

Important note. If the departure segment admits further beneficial cooperation for either agent, the resulting joint strategy is not an ECJS. However, as shown in Appendix E.2, such strategies are disregarded by Algorithm 5 when evaluating all cooperation nodes as potential departure nodes.

5.2 Optimal ECJS Computation

To map all relevant joint strategies in $\mathbb{ECJS}$, we evaluate each cooperation node as a potential cooperation departure node, retaining only the dominating ECJS and discarding dominated concatenations of stable paths. Theorem 3, proven in Appendix G, establishes that Algorithm 5 identifies all non-dominated (or equivalent) ECJS.

THEOREM 3. *Algorithm 5 returns a set $ECJS_{map}$ of joint strategies satisfying the following properties:*
(1) ***PNE Guarantee:*** *Every element in $ECJS_{map}$ is a PNE.*
(2) ***Dominance:*** *Every ECJS not included in $ECJS_{map}$ is dominated by at least one strategy in $ECJS_{map}$.*

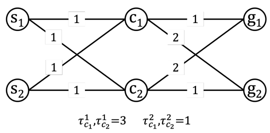
Complexity. Algorithm 5 iterates over all m cooperation nodes, invoking SHORTEST STABLE PATHS and Algorithm 4 for each, yielding complexity of $O(m \cdot (|E| + |V|\log|V|))$.

6 ECJS SELECTION

When multiple non-dominated ECJSs exist, trade-offs arise: one agent's travel time improves only if the other's worsens (see example

Algorithm 5 OPTIMAL ECJS$(G, V_C, s_1, s_2, g_1, g_2)$

$ECJSmap \leftarrow \{(SIP_{s_1,g_1}, SIP_{s_2,g_2})\}, \quad dominated \leftarrow \emptyset$
for all $c_d \in V_C$ **do**
  if $c_d \notin dominated$ **then**
    $ECJSmap[c_d] \leftarrow$ ALGORITHM $4(G, V_C, c_d, \dots)$
    $dominated \leftarrow dominated \cup$
    SHORTEST STABLE PATHS$(G, V_C, c_d, SIP_{g_1}, SIP_{g_2})$
$S_1 \leftarrow$ BR$(G, V_C, s_1, g_1, s_2, g_2, SIP_{s_2,g_2})$
$S_2 \leftarrow$ BR$(G, V_C, s_2, g_2, s_1, g_1, SIP_{s_1,g_1})$
if $S_1 \neq SIP_{s_1,g_1}$ or $S_2 \neq SIP_{s_2,g_2}$ **then**
  $ECJSmap \leftarrow ECJSmap \setminus \{(SIP_{s_1,g_1}, SIP_{s_2,g_2})\}$
return $ECJSmap \setminus dominated$



	$s_2 \to c_1 \to g_2$	$s_2 \to c_2 \to g_2$
$s_1 \to c_1 \to g_1$	$(3, 4)$	$(5, 5)$
$s_1 \to c_2 \to g_1$	$(6, 6)$	$(4, 3)$

Figure 4: Two non-dominated ECJSs: cooperation at c_1 benefits a_1, while cooperation at c_2 benefits a_2; both outperform no cooperation.

in Figure 4). However, since every ECJS improves upon independent shortest paths, agents share an interest in agreeing on a strategy.

This coordination problem can be modeled as a *correlation game* [3], where agreement on a joint strategy is essential to avoid suboptimal outcomes. Coordination can be achieved either through *conventions* [40, 46] (e.g., minimizing the maximum path time, maximizing minimum utility relative to the *SIP*, or maximizing social welfare) or through the concept of a correlated equilibrium[3]. In this setting, the agents agree on a shared probability distribution over the set of non-dominated ECJSs and use it to jointly select a strategy (e.g. via an external correlation device). This mechanism ensures consistent execution and prevents mismatched decisions, enabling the agents to maximize the expected value of a shared objective function, such as fairness or social welfare.

Since the joint strategies in $\mathbb{ECJS}$ inherently reflect conflicting preferences, reaching a mutually acceptable agreement can be challenging. To address this, we model the problem as a bargaining game [41], aiming to identify a correlated equilibrium guided by standard bargaining solution concepts. Specifically, we seek a solution that satisfies the following properties:

(1) **Pareto Optimality**: No agent's outcome can be improved without worsening the outcome of the other.
(2) **Symmetry**: Identical agents with symmetric options should receive identical outcomes.
(3) **Fairness**: The outcome should be impartial and just, avoiding favoritism or discrimination between agents.

While *Pareto optimality* and *symmetry* are well defined, *fairness* is more nuanced and context dependent. We therefore operationalize fairness by comparing four common bargaining solutions: Nash solution [41], Kalai-Smorodinsky solution [35], and the egalitarian and utilitarian solutions.

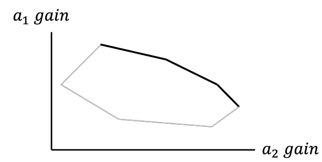


Figure 5: An illustration of the set S of all expected gain pairs

Formally, we denote the set of non-dominated joint strategies in $\mathbb{ECJS}$ by $\mathbb{ECJS}^{ND}$, and represent the selection of a probability distribution over the joint strategies in $\mathbb{ECJS}^{ND}$ as a bargaining game (S, d). The utility of each agent is defined by the time saved compared to its *SIP*. That is, for a given joint strategy (π^1, π^2) the utility of agent a_i is defined as $u_i(\pi^1, \pi^2) = T(SIP_{s_i,g_i}) - T(\pi^i|\pi^{-i})$. We define the disagreement point d as:

$$d = \left(u_1(SIP_{s_1,g_1}, SIP_{s_2,g_2}),\ u_2(SIP_{s_2,g_2}, SIP_{s_1,g_1})\right)$$

The set S of all possible outcomes in the bargaining game is defined as the set of expected utility pairs achievable by randomizing over joint strategies in $\mathbb{ECJS}^{ND}$. Let $\mathbb{ECJS}^{ND} = \{(\pi^1_1, \pi^2_1), \dots, (\pi^1_n, \pi^2_n)\}$. Then

$$S = \left\{ \sum_{i=1}^{n} \alpha_i \cdot \left(u_1(\pi^1_i, \pi^2_i),\ u_2(\pi^2_i, \pi^1_i)\right) \;\middle|\; \alpha_i \geq 0,\ \sum_{i=1}^{n} \alpha_i = 1 \right\}$$

where each α_i denotes the probability assigned to the i-th joint strategy in $\mathbb{ECJS}^{ND}$, and the resulting pair represents the expected utilities of the two agents. We determine the parameters $\alpha_1, \dots, \alpha_n$ for each solution concept as follows.

(1) **Nash Solution**, maximizes the product of the agents' utilities relative to the disagreement point, capturing mutual benefit under rational cooperation.
(2) **Kalai-Smorodinsky Solution**, ensures proportional fairness by preserving each agent's utility ratio relative to its maximum attainable utility.
(3) **Egalitarian Solution**, seeks to equalize outcomes by maximizing the minimum utility across agents.
(4) **Utilitarian Solution**, maximizes the overall social welfare.

Since we seek a Pareto-optimal solution, we restrict our attention to the Pareto frontier of the set S, where no agent's expected utility can be improved without worsening the other's. As a result, the solution corresponds to a probability distribution supported on at most *two* joint strategies. The specific strategies involved, however, may vary depending on the chosen bargaining solution concept.

7 IMPLEMENTATION AND RESULTS

To provide a comprehensive perspective of IC2PP, understand how different selection methods affect individual and social outcomes, and examine factors influencing cooperation incentives, we fully implemented all algorithms[2] and evaluated IC2PP algorithms on randomly generated graphs with diverse topologies, travel times, and node delays. To broaden the experimental evaluation, we also utilize the MAPF Benchmark Set [44], which provides a variety of grid-based maps. To adapt these maps to IC2PP, each grid cell is

[2]The code repository is publicly available at https://github.com/shedlezki/Intermittent-Cooperation-Simulation.

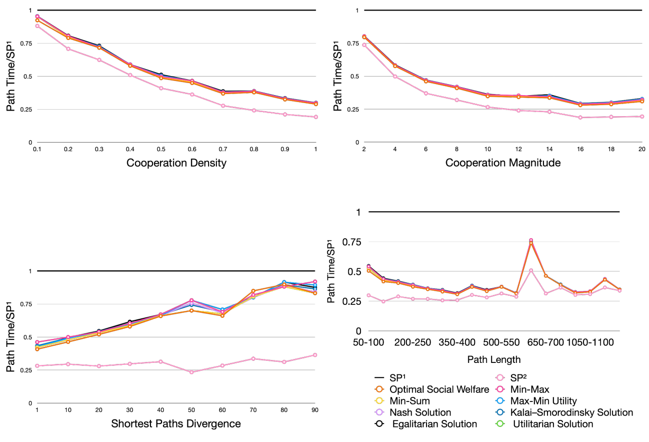


Figure 6: Effects of topology attributes on individual path times.

modeled as a node with two delay values, τ^1 for single-agent and τ^2 for cooperative execution, while edges between adjacent cells have a travel time of 1 time unit.

To capture the factors that influence cooperation, we varied the following attributes of an IC2PP instance:

- **Cooperation Magnitude** — The average ratio between the task execution time of a single agent and that of two agents cooperating at a cooperation node.
- **Cooperation Density** — the ratio of cooperation nodes to all nodes in the graph.
- **Path Lengths** — the average path time between the start and target nodes of the agents.
- **Shortest Paths Divergence (SPD)** — The minimal path time separating the agents if following their individual shortest paths from their respective start to target nodes, reflecting how aligned these paths are.

We compare methods for joint strategy selection as follows. First, we compute the optimal social welfare for each scenario using a polynomial time algorithm (explained in details in Appendix H). Then, using Algorithm 5, we identify all non-dominated ECJS and evaluate the *Price of Anarchy* (PoA) [39] and *Price of Stability* (PoS) [1], defined respectively as the ratios of the worst and best ECJS to the optimal social welfare. Finally, we compare individual path times and social welfare across ECJS selection methods under varying *Cooperation Magnitude, Density, Path Length*, and *SPD*. We conducted four separate experiments, each evaluating the effect of a different cooperation factor (Magnitude, Density, Path Length, and SPD) across a range of values. For each configuration, the algorithms were tested on 50 randomly sampled scenarios from the MAPF Benchmark Set. To control Path Length and SPD, start and target nodes were selected based on predefined path lengths.

Figure 6 summarizes cooperation parameters' impact on path times via improvement ratios relative to independent shortest paths. Decreasing shortest paths divergence or increasing cooperation density/magnitude improves individual and total path times across selection methods. Conversely, longer paths minimally affect individual or social benefits.

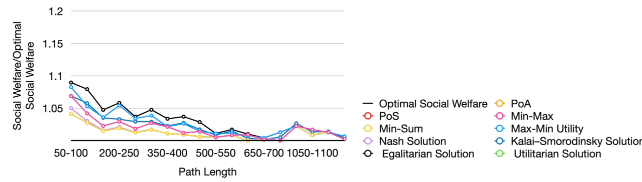


Figure 7: Social Welfare vs. Path Lengths

Figure 7 shows that increasing path length improves the ratio of achieved social welfare to the optimal social welfare, with the Utilitarian solution and Min-Sum convention consistently attaining the highest ratios and approach the optimum as path length increases (other factors appear in Appendix I).

Overall, since selection methods produce comparable outcomes for individuals, agents can adopt shared social objectives, such as makespan minimization or social welfare maximization, without compromising individual performance.

8 CONCLUSIONS

In this paper, we introduced the IC2PP framework for analyzing cooperation among two self-interested agents whose incentives to cooperate depend on temporal and spatial context. We characterized the structure of Pure Nash Equilibria (PNE) and developed an efficient algorithm to enumerate all non-dominated PNE, proving existence in all instances. We compared coordination mechanisms for two agents and examined how different factors affect cooperation outcomes in general.

These findings establish a foundation for understanding and guiding rational cooperation in autonomous multi-agent systems operating in real-world settings, where cooperation incentives are context-dependent and shaped by temporal and spatial synchronization. Therefore, this work opens the door to a broad line of future research, including extensions to $k > 2$ agents, online planning, and scenarios with limited information about other agents.

REFERENCES

[1] Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Éva Tardos, Tom Wexler, and Tim Roughgarden. 2008. The price of stability for network design with fair cost allocation. *SIAM J. Comput.* 38, 4 (2008).

[2] Tone Arnold and Ulrich Schwalbe. 2002. Dynamic coalition formation and the core. *Journal of Economic Behavior & Organization* 49, 3 (2002).

[3] Robert J Aumann. 1974. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics* 1, 1 (1974).

[4] Haris Aziz, Felix Brandt, and Hans Georg Seedig. 2013. Computing desirable partitions in additively separable hedonic games. *Artificial Intelligence* 195 (2013).

[5] Haris Aziz, Grzegorz Lisowski, Mashbat Suzuki, and Jeremy Vollen. 2024. Neighborhood stability in assignments on graphs. *arXiv preprint arXiv:2407.05240* (2024).

[6] Haris Aziz and Rahul Savani. 2016. Hedonic Games. In *Handbook of Computational Social Choice*. Cambridge University Press.

[7] Masoud Bashiri and Cody H. Fleming. 2017. A platoon-based intersection management system for autonomous vehicles. In *2017 IEEE Intelligent Vehicles Symposium (IV)*. 667–672. https://doi.org/10.1109/IVS.2017.7995794

[8] Damien Berriaud, Andrei Constantinescu, and Roger Wattenhofer. 2023. Stable dinner party seating arrangements. In *Proc. of WINE*.

[9] Andrea Bertolini. 2022. *Decentralized algorithms for multi-agent pathfinding*. Ph.D. Dissertation. Politecnico di Torino.

[10] Filippo Bistaffa, Alessandro Farinelli, and Sarvapali Ramchurn. 2015. Sharing rides with friends: A coalition formation algorithm for ridesharing. In *Proc. of AAAI*.

[11] Hans L Bodlaender, Tesshu Hanaka, Lars Jaffke, Hirotaka Ono, Yota Otachi, and Tom C van der Zanden. 2020. Hedonic seat arrangement problems. *arXiv preprint arXiv:2002.10898* (2020).

[12] Caroline Bonhomme, Christophe Grand, Charles Lesire, Jean-Louis Dufour, and Christophe Guettier. 2024. Multi-Agent Path Finding with Task Assignment and Supporting Constraints. In *Proc. of ECAI*.

[13] Víctor Bucarey, Carlos Casorrán, Martine Labbé, Fernando Ordoñez, and Oscar Figueroa. 2021. Coordinating resources in stackelberg security games. *European Journal of Operational Research* 291, 3 (2021).

[14] Martin Bullinger and Warut Suksompong. 2024. Topological distance games. *Theoretical Computer Science* 981 (2024).

[15] Matteo Cederle, Marco Fabris, and Gian Antonio Susto. 2024. A Distributed Approach to Autonomous Intersection Management via Multi-Agent Reinforcement Learning. https://arxiv.org/abs/2405.08655

[16] Esra Ceylan, Jiehua Chen, and Sanjukta Roy. 2023. Optimal seat arrangement: what are the hard and easy cases? *arXiv preprint arXiv:2305.10381* (2023).

[17] Mengjing Chen, Weiran Shen, Pingzhong Tang, and Song Zuo. 2019. Dispatching Through Pricing: Modeling Ride-Sharing and Designing Dynamic Prices.. In *Proc. of IJCAI*.

[18] Julio B Clempner and Alexander S Poznyak. 2015. Stackelberg security games: Computing the shortest-path equilibrium. *Expert Systems with Applications* 42, 8 (2015).

[19] Catherine Cleophas, Caitlin Cottrill, Jan Fabian Ehmke, and Kevin Tierney. 2019. Collaborative urban transportation: Recent advances in theory and practice. *European Journal of Operational Research* 273, 3 (2019).

[20] Argyrios Deligkas, Eduard Eiben, Dušan Knop, and Šimon Schierreich. 2024. Individual rationality in topological distance games is surprisingly hard. *arXiv preprint arXiv:2404.14128* (2024).

[21] Kurt Dresner and Peter Stone. 2008. A multiagent approach to autonomous intersection management. *Journal of artificial intelligence research* 31 (2008).

[22] Florian Drews and Dennis Luxen. 2021. Multi-Hop Ride Sharing. *Proceedings of the International Symposium on Combinatorial Search* 4, 1 (Aug. 2021), 71–79. https://doi.org/10.1609/socs.v4i1.18279

[23] Jacques H Dreze and Joseph Greenberg. 1980. Hedonic coalitions: Optimality and stability. *Econometrica: Journal of the Econometric Society* (1980).

[24] Martin Gairing, Kostas Kollias, and Grammateia Kotsialou. 2020. Existence and efficiency of equilibria for cost-sharing in generalized weighted congestion games. *ACM Transactions on Economics and Computation (TEAC)* 8, 2 (2020).

[25] Laurent Gourves, Jérôme Monnot, Stefano Moretti, and Nguyen Kim Thang. 2015. Congestion games with capacitated resources. *Theory of Computing Systems* 57, 3 (2015).

[26] Mario Guajardo, Mikael Rönnqvist, Patrik Flisberg, and Mikael Frisk. 2018. Collaborative transportation with overlapping coalitions. *European Journal of Operational Research* 271, 1 (2018).

[27] Anupam Gupta, Aravind Srinivasan, and Éva Tardos. 2008. Cost-sharing mechanisms for network design. *Algorithmica* 50, 1 (2008), 98–119.

[28] Matthew Hausknecht, Tsz-Chiu Au, and Peter Stone. 2011. Autonomous Intersection Management: Multi-intersection optimization. In *Proc. of IROS*.

[29] Ara Hayrapetyan, Éva Tardos, and Tom Wexler. 2006. The effect of collusion in congestion games. In *Proc. of ACM STOC*.

[30] Florence Ho, Rúben Geraldes, Artur Gonçalves, Bastien Rigault, Benjamin Sportich, Daisuke Kubo, Marc Cavazza, and Helmut Prendinger. 2020. Decentralized multi-agent path finding for UAV traffic management. *IEEE Transactions on Intelligent Transportation Systems* 23, 2 (2020).

[31] Martin Hoefer, Daniel Vaz, and Lisa Wagner. 2018. Dynamics in matching and coalition formation games with structural constraints. *Artificial Intelligence* 262 (2018).

[32] Yunfei Hou, Xu Li, and Chunming Qiao. 2012. TicTac: From transfer-incapable carpooling to transfer-allowed carpooling. In *Proc. of GLOBECOM*.

[33] Yichuan Jiang. 2015. A survey of task allocation and load balancing in distributed systems. *IEEE Transactions on Parallel and Distributed Systems* 27, 2 (2015).

[34] Jozef Juraško, Lukáš Lovás, Rastislav Bencel, and Peter Trúchly. 2025. Toward 6G-V2X: Edge-Assisted Platoon Coordination for Cooperative Intersection Control. In *2025 International Symposium ELMAR*. 21–24. https://doi.org/10.1109/ELMAR66948.2025.11194002

[35] Ehud Kalai and Meir Smorodinsky. 1975. Other Solutions to Nash's Bargaining Problem. *Econometrica* (1975), 513–518.

[36] Jiwon Kang, Yongjun Choi, and Keemin Sohn. 2025. Simultaneous Optimization of Traffic Signal Control and Vehicle Platooning Scheme Based on Connected and Automated Vehicle (CAV) Technology. *IEEE Access* (2025).

[37] Alf Kimms and Igor Kozeletskyi. 2017. Consideration of multiple objectives in horizontal cooperation with an application to transportation planning. *IISE Transactions* 49, 12 (2017).

[38] M. Klusch and A. Gerber. 2002. Dynamic coalition formation among rational agents. *IEEE Intelligent Systems* 17, 3 (2002).

[39] Elias Koutsoupias and Christos Papadimitriou. 1999. Worst-case equilibria. In *Proc. of STACS*.

[40] David Lewis. 2008. *Convention: A philosophical study*. John Wiley & Sons.

[41] John F et al. Nash. 1950. The bargaining problem. *Econometrica* 18, 2 (1950).

[42] Sarvapali D. Ramchurn, Maria Polukarov, Alessandro Farinelli, Cuong Truong, and Nicholas R. Jennings. 2010. Coalition formation with spatial and temporal constraints. In *Proc. of AAMAS*.

[43] Duncan Rheingans-Yoo, Scott Duke Kominers, Hongyao Ma, and David C Parkes. 2019. Ridesharing with driver location preferences. *arXiv preprint arXiv:1905.13191* (2019).

[44] Roni Stern, Nathan Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, TK Kumar, et al. 2019. Multi-agent pathfinding: Definitions, variants, and benchmarks. In *Proc. of SoCS*.

[45] Peter Stone, Gal Kaminka, Sarit Kraus, and Jeffrey Rosenschein. 2010. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proc. of AAAI*.

[46] Peter Vanderschraaf. 1995. Convention as correlated equilibrium. *Erkenntnis* 42, 1 (1995).

[47] Berthold Vöcking and R Aachen. 2006. Congestion Games: Optimization in Competition.. In *ACiD*.

[48] Dayong Ye, Minjie Zhang, and Danny Sutanto. 2013. Self-adaptation-based dynamic coalition formation in a distributed agent network: A mechanism and a brief survey. *IEEE Transactions on Parallel and Distributed Systems* 24, 5 (2013).

[49] Yiqun Zhang, Pengfei Zhang, Xiaodong Wang, Feng Song, Chaoyong Li, and Junhong Hao. 2022. An open loop Stackelberg solution to optimal strategy for UAV pursuit-evasion game. *Aerospace Science and Technology* 129 (2022).

[50] Zishuo Zhao, Xi Chen, Xuefeng Zhang, and Yuan Zhou. 2022. Dynamic car dispatching and pricing: Revenue and fairness for ridesharing platforms. *arXiv preprint arXiv:2207.06318* (2022).