Simulation and resolution of spatial conflicts between pedestrians based on social choice theory

Paper #3876

Abstract. Simulating pedestrians in public spaces is a challenge due to the complexity of human behavior and spatial conflict resolution mechanisms. Beyond models based on physical forces or individual-level planners, some interactions — especially in constrained environments — involve verbal or gestural negotiations to reach consensus. We propose a novel modeling framework based on social choice theory to model pedestrian preferences and cooperation in local negotiation mechanisms to resolve spatial conflicts. By proposing a simulation tool, we validate the relevance of our model by comparing it to SFM (Social Force Model) and CBS-MAPF (Multi-Agent Pathfinding) models. Experimental results show that the proposed approach allows reaching efficient and realistic behaviors, plus, being computed at the agent level, avoids the computational costs of global planning approaches.

1 Introduction

Complex environments, such as urban spaces, present high stakes when it comes to considering the flow of people. Some environments require careful design of spaces for crowd management, for example, to manage and optimize critical situations such as evacuations in dense urban spaces [1][13]. Simulation is a powerful tool for modeling these environments and testing complex scenarios. For this, it is necessary to establish realistic models of pedestrian behavior and movement.

The movement of pedestrians as a crowd or from an individual point of view has been extensively studied. In particular, a major study has been carried out through the social forces model [9] from a physics point of view, where each pedestrian is governed by a set of forces exerted by obstacles and his target. Other models have been proposed, such as models based on agent-based modeling (ABM) or stochastic games [3]. These models adopt a different viewpoint by modeling agent behavior at an individual level to observe emerging realistic behaviors.

This article focuses on the social behavior of pedestrians in the face of spatial conflicts. In [14][25], the authors show that game theory can be used to model their decisions. We draw on this work to analyze complex scenarios with obstacles, where pedestrians attempt to resolve potential collisions. The aim is not to predict crowd movements, but to propose a cognitive model of human decision-making, by modeling pedestrian interactions and conflict resolution in an interpretable and realistic way. Taking into account trajectories and spatial representations, we exploit cooperative game theory [11] for the pragmatic aspect of resolution. In addition, the aggregation of individual preferences—arising from communications among pedestrians—can be interpreted through the lens of social choice theory. We choose to draw on social choice theory or voting theory [18] [4] to study conflict resolution through cooperative negotiations between pedestrians. To the best of our knowledge, no other approach draws on social choice theory to model pedestrian behavior through negotiation. Although this theory is generally applied to economic or political models, it also sheds light on micro-interactions through implicit voting systems. In this way, we explore the social interactions of pedestrians from the angle of negotiation and collective choice.

Our study has links with the resolution of MAPF [22] (Multi-Agent Path Finding) problems, which involve computing compatible paths for several agents in a given environment. In contrast to this work, we aim to model realistic pedestrian behaviors, dependent on individual decisions and negotiations. Although such resolutions of MAPF are different approaches, we compare them to our model to assess the quality of solutions found by pedestrians.

This article is organized as follows: section 2 presents existing work, section 3 develops the proposal of our model based on social choice theory, and section 4 presents experimental results of the model comparing it to ad-hoc versions of SFM [9] and CBS-MAPF [19], before concluding the article in section 5.

2 Existing works

A number of physics-based models focus on pedestrian interactions in crowds, inspired by the study of particle dynamics. The Social Force Model (SFM) presented by Helbing and Molnar [9] yielded notable results in predicting crowd movements. Such a model made it possible to reproduce specific situations observable in real-life situations, such as pedestrians waiting on the side of a train platform to let passengers off. Several extensions have been made to SFM [27][10] to better adapt to real-life scenarios, notably to avoid blockages.

Pedestrian dynamics can also be modeled and simulated by motion planning at individual-level [3]. This approach enables a fine-grained algorithm to take into account complex scenarios with obstacles, with the emphasis on movement realism. These planners can thus be used in virtual reality [3][8] or in robots evolving among humans [21].

We recall the differences between physics models and those based on agent behavior (e.g., from game theory or standard simulations). The latter attempt to understand human behavior in decision-making processes, thus making it possible to simulate agents or robots with human behavior in, e.g., virtual environments. Studies also extend to the analysis of human cognitive processes [24] [5]. However, such models can be computationally more expensive and lend to subjective interpretation; scaling them up to a large group may reveal unrealistic features on local models. Conversely, physics models use mean-field game theory [12] or physics equations to approximate and study a crowd as a complex system. Both approaches have led to significant advances, but physics models focus on a global vision with the aim of predicting behavior rather than anticipating and modeling the individual [26]. Furthermore, these models are not interested in individual modeling, relevant in the emergence of realistic behaviors, but rather in the global movement of crowds [3].

Compared to the resolution of MAPF problems, with the calculation of compatible paths for a set of agents, we focus on pedestrian modeling with individual decisions made through negotiations. Our agents therefore follow a cognitive process rather than a global optimization like the CBS (Conflict-Based Search) algorithm for MAPF problem-solving [19], or Increasing Cost Tree Search [22]. Unlike the latter, which are offline planners, our approach is closer to an online planner [23] where pedestrian behaviors adapt to the situations encountered. In this respect, we emphasize the possible sub-optimality of certain solutions, which nevertheless lead to social agreements between pedestrians that achieve their objective. This is not the case with centralized offline planners such as CBS-MAPF [23], which do not adapt to the situations encountered.

While social choice in pedestrian conflict resolution through negotiation has not yet been studied in the literature, work on pedestrian negotiations nevertheless exists [14] [2]. These are mainly concerned with vehicle-pedestrian or traffic situations involving explicit interactions and gestures, e.g. with a hand. Such models aim to predict and/or detect pedestrian behaviors with game theory or statistical models [15]. In this study, we aim to simulate spatial conflicts resolved by negotiations only between pedestrians, whose behaviors are based on social choice and inspired by previous work.

3 Conflict Negotiation Model (CNM)

In the previous sections, we highlighted the prospects for research into the construction of cognitive models of pedestrians. Specifically, we focus on the decision-making aspect within a social behavior framework. In order to model a pedestrian and his interactions, we focus on the pedestrian's thinking process in relation to his or her movements in space. Obstacle-dense spaces are more prone to this need to optimize motions. We therefore focus on conflict resolution in these spaces, while the planning aspect will use the A^* [7] algorithm to compute optimal paths.

We now present our *Conflict Negotiation Model* (CNM) as a proposal for modeling pedestrians attempting to resolve conflicts (potential collisions) in the situations described above. In this model, each pedestrian is modeled with the aim of reaching a respective target position by finding a collision-free path, considering other pedestrians and obstacles as static.

Our approach, based on social choice theory, is inspired by the aggregation of preferences (here at the scale of an individual), interactions by communication between pedestrians, and finally the outcome of a choice accepted by neighboring pedestrians leading to a local consensus. It should be noted that collective decisions differ from classical social choice in that they do not exploit structured voting rules, the focus being on the resolution of spatial conflicts.

3.1 Pedestrian modeling

Each agent is denoted a_i, where i is the identifier. In the following, we consider that time is discrete and space is discretized in two dimensions (a grid). Before defining the negotiations, we introduce our agent modeling with the characteristics that define an agent. In particular, the preferences that characterize a pedestrian's own behavior. Let a_i be an agent defined by the following characteristics:

- **A position.** A coordinate (x, y) representing a cell in two-dimensional space, noted $pos(a_i)$.
- **A speed.** A positive real $v(a_i)$ such that $0 \leq v(a_i) \leq 2$.
- **A target.** A target cell noted $goal(a_i)$.
- **Preferences.** They characterize a_i by modeling its desires defined by its situation at a given moment. Below, we detail a_i's Motivation and Attention.

Motivation. The motivation of a_i denoted $mot(a_i)$ is an abstract positive real modeling, in this case, an agent's willingness to wait or give up a place. We pose $0 < mot(a_i) \leq 1$. This represents the bargaining strength relative to a_i's desires, communicated in costs when negotiating with others. It psychologically models a_i's motivation to wait or to change path in favor of others.

Attention. The attention of a_i, $att(a_i)$, is a positive real denoting its concentration. $0 < att(a_i) \leq 1$ which is similar to a percentage chance of detecting another agent in a_i's field of view. For example, a distracted pedestrian with a phone.

For the sake of clarity, Figure 1 illustrates a capture of an environment in simulation, here named *Map 6*[1]. The rest of the model is given in sections 3.5 and 4. Static obstacles are represented by red squares filling entire cells. Goals, and here intermediate goals, are green diamonds. Numbered agents are displayed as colored discs. The dotted arrows are the trajectories calculated with A^* and the green ones are the velocity vectors. Note that a velocity vector is calculated with the direction and sense given by a trajectory, the norm is the scalar $v(a_i)$ for an agent a_i. They are displayed here for information purposes.

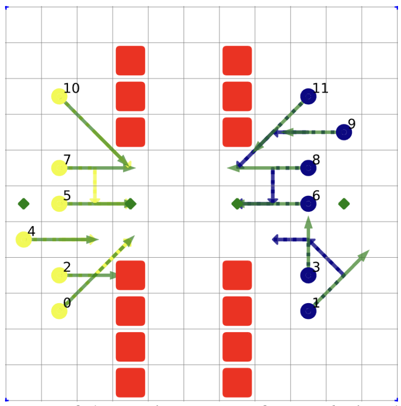

Figure 1: Capture of the environment of *Map 6*, simulating CNM.

Characteristics can be summarized in a *cost function* as the sum of an agent's preferences and capacities, communicated when negotiating with other agents. A proposed cost function is given below.

3.2 Cost function and behaviors

Here we set out the definitions and possible interactions on which our model is based.

3.2.1 Cost function

Def. Collision: A collision is defined at a given moment when two agents occupy the same cell.

[1] For environments with a multi-goal configuration, each agent is assigned the furthest goal.

Def. Trajectory: a trajectory or path T is a finite and contiguous sequence of cells defined as

$$T \triangleq [(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)]$$

for a trajectory of length $size(T) = n$ and such that

$$\forall i \in [1, n-1], |x_i - x_{i+1}| \leq 1 \wedge |y_i - y_{i+1}| \leq 1$$

Note that not all cells are necessarily distinct in pairs.

Def. Cost function : A cost function denoted C_T for a trajectory T aggregates the characteristics of an agent a_i and allows us to evaluate the "cost" or penalty associated with the choice T.

$$C_T : characteristics(a_i) \longrightarrow \mathbf{R}^+$$

C_T is used to guide the agent's decisions in a given environment, as it seeks to minimize its costs in the simulation. This cost function is directly inspired by an individual's welfare function, used collectively in classical social choice theory. We propose a first cost function :

$$\boxed{C_T(a_i) = \frac{K_1 \cdot size(T)}{d(pos(a_i), goal(a_i))} + K_2 \cdot v(a_i) + \frac{K_3}{mot(a_i)}} \quad (1)$$

With d the Euclidean distance in two dimensions[2] and K_i constant weight factors. The proposed cost function (1) highlights the following phenomena in particular:

- $\frac{size(T)}{d(pos(a_i),goal(a_i))} \searrow \implies C_T(a_i) \searrow$ reflecting the optimality of path T and suggesting room for maneuver in the adaptation of a_i on this path.
- $v(a_i) \nearrow \implies C_T(a_i) \nearrow$ translating a_i's high inertia and difficulty adapting to other trajectories.
- $mot(a_i) \searrow \implies C_T(a_i) \nearrow$ reflecting a_i's unwillingness to negotiate a change of path. This can be seen as a criterion of fatigue.
- $att(a_i)$ is not taken into account in the equation, but is considered beforehand, as explained in the perception step later.

The cost function provides an associated cost for each trajectory that a pedestrian considers when making a decision. In the following, we use $K_1 = K_2 = K_3 = 1$.

3.2.2 Influences

Def. Influence: an influence — in Ferber's sense [6] — given a grid fixed at a given time, is here a situation involving at least one agent *susceptible* to modify its own state (e.g., preferences) of the one of another agent. An influence may or may not modify an agent, and may occur internally (within the agent) or originate externally (from another agent or the environment). Here, we briefly define the two types of influence under consideration:

- **Internal influence** : An agent's cognitive processes alone enable it to modify its preferences according to its situation.
- **External influence** : An agent a_i is influenced by its environment, e.g., static obstacles that change its trajectory. Then, other agents can communicate with a_i, influencing its decision-making.

We adopt the classic approach to modeling a situated agent, using the loop: [*Perception* → *Decision* → *Action*].

[2] With this function d, the ratio with $K_1 = 1$ is close to 1. Without normalizing C_T, this term has little weight. A more refined version could be a difference or a more realistic heuristic of d.

Perception initiates the pedestrian's cognitive process, taking into account other agents present within a radius *RADIUS* set at 2 in the experiments. This part is characterized as an internal influence. The Decision is governed by external influences, i.e., communications with other agents, with the aim of establishing a local consensus to avoid collisions. Although the decision can be made by the pedestrian alone, our model includes *forced* waiting to resolve *potential* collisions, so the pedestrian will depend on the state of other agents. Finally, Action, which is not our main object of study, is directly related to decision-making: subject to a feasible movement, the decision taken by the pedestrian will be the one he makes.

The cognitive process of our pedestrians lies mainly in decision-making, through influences, communication and self-interested decision-making. We distinguish between two phases of decision-making: Analysis and Choice.

Analysis takes place through communication with other pedestrians, and is referred to as *the negotiation*. Then, the choice that the pedestrian will make for the next time step is obtained by what we will call *a vote* between pedestrians, respecting individual preferences as far as possible. This division into two phases underlines the importance of these two processes in the decision-making process: on the one hand, pedestrians consider alternative trajectories to resolve collisions, and on the other, a search for local consensus on collisions takes place in order to confirm decisions.

3.3 System dynamics

Algorithm 1 SMA

```
for a_i ∈ AGENTS_SCHEDULER do
    move(a_i)
    think(a_i)
end for
for a_i ∈ AGENTS_SCHEDULER do
    neighs ← filter(NOT_NEGOTIATING,
                neighbors(a_i, RADIUS) + {a_i})
    if len(neighs) > 1 then
        VOTE_AND_NEGOTIATIONS(sort_v(neighs))
                                    ▷ Nested scheduler
    end if
end for
```

The CNM proposal has an intrinsic dynamic that organizes voting and negotiations. First, agents behave individually with internal processes. Second, they vote and negotiate collectively in the manner of a centralized system that manages $O(n)$ local exchanges. Nevertheless, the second part on vote and negotiations has a distributed equivalent system with $O(K * n^2)$ mutual exchanges, with n agents and a vote in K iterations. There exists multiple possible implementations of CNM, we choose to implement the dynamic in two parts described before with the centralized approach for faster computations.

We propose a multi-agent system dynamic for CNM simulation in Algorithm 1 that orders the cognitive process and governs the agents' actions. At initialization, agents have uniformly randomly generated speed and preferences within the value bounds defined in Section 3, the rest is pre-configured. Dynamics are organized by a first loop on agents moving with $move(a_i)$. $think(a_i)$ is the function that chooses the next trajectory for a_i, such that a_i computes a set of possible trajectories and chooses among them a trajectory T of minimum cost. T will thus be the item of voting and negotiation. Then, the second agent loop resolves conflicts through votes between neighbors not already involved in a vote ($filter$ enables this selection within a given radius). Note that the order in which agents are selected is *random*. However, the speed $v(a_i)$ comes into play in the vote on line 9, where the $neighs$ agents are sorted in descending order of speed ($sort\_v$). This prioritizes collisions with the fastest agents, favoring the latter to negotiate before the others. This happens during the vote presented later in Algorithm 2.

In addition, the initial choice made by an agent is not always the one that will be realized. Assuming that this choice is physically possible, the vote will give rise to negotiations that may change the actions actually taken, as shown later in Algorithm 3.

3.4 Cognitive process

3.4.1 Perception and attention

We decide to give pedestrians a fixed perception radius (*RADIUS*), for the moment without considering static obstacles, enabling them to detect neighboring agents within this perimeter. This model is perfectible, for example, a ray casting [17] would provide a finer perception. This perception radius enables the agent to estimate potential future collisions and communicate with its neighbors to avoid them.

The first step (line 6, Alg. 1) is to determine when analysis comes into play in the agent's decision-making process. Here, we define a simple observation process that enables an agent to assess its situation. If no analysis is required, the agent simply maintains its initial trajectory. In this model, an agent a_1's attention plays a role in its ability to observe: there is a probability $att(a_1)$ that a pedestrian a_i will succeed in communicating, i.e. negotiating, with a_1. The agent a_1 will therefore not initiate its analysis and decision-making process in the event of failure, even if a_i is present in its field of vision.

3.4.2 Vote and Negotiations

At each iteration of the system, a systematic vote takes place between neighboring pedestrians. The vote and the initiation of the negotiations are shown in the Algorithm 2. The items of vote are the trajectories or paths of the pedestrians themselves, this is shown through a proposal of paths (line 6). The agents check the collisions between paths in order to establish *necessary* negotiations (line 7): the vote here, on the contrary to a classic social choice approach, is strictly pragmatic. The preferences of the agents in this vote can be considered as must-satisfy conditions to avoid collision (see **(HYP2)** defined later). Hence, the vote happens at each system iteration to assess collisions and results of negotiations.

Afterward, each collision becomes a negotiation between two agents (lines 14 and 21). The decision-making process is thus carried out throughout the negotiations, so that both agents choose an appropriate trajectory. The negotiation, line 14, translates the proposal of a_i's trajectory to a_j, who then considers it. Both agents aim to minimize their cost, which subsequently updates their characteristics, in particular their motivation, through the success or failure of negotiations. When a negotiation succeeds, i.e., a_j has reduced the cost of a path T' favorable to a_i's trajectory (cf. section 3.4.3), agent a_i becomes more inclined to negotiate thereafter. This translates into a $GAIN$ (a positive real) in his motivation (lines 16-20) and vice versa if the cost of a path T' has not changed. A direct impact is on the cost function C_T (1).

The selection of pairs (a_i, a_j) is pseudo-random. Negotiations are symmetrical, i.e., a_i negotiates with a_j and vice versa. We choose to run the vote for a fixed time step Δ (fixed at 3 in the experiments), in which negotiations, assumed to be fast cognitive processes, will take place in sub-time steps. Voting therefore takes place during an iteration of the overall system, enabling active, convergent negotiation.

Algorithm 2 VOTE AND NEGOTIATIONS

Require: Subset of voting agents A, e.g., neighbors
1: **function** VOTE_AND_NEGOTIATIONS(A)
2: $N = 0$
3: **while** $N < \Delta$ **do**
4: $collisions = \{\}$
5: **for** $(a_i, a_j)_{i \neq j} \in A$ **do** ▷ Paths' proposal
6: **if** $collides(path(a_i), path(a_j))$ **then**
7: $collisions = collisions \cup \{(a_i, a_j)\}$
8: **end if**
9: **end for**
10: **if** $collisions == \{\}$ **then**
11: **return**
12: **end if**
13: **for** $(a_i, a_j) \in collisions$ **do** ▷ Negotiations
14: $success = NEGOTIATE(a_i, a_j)$
15: ▷ a_j considers a_i's trajectory
16: **if** $success$ **then**
17: $a_i.motivation+ = GAIN$
18: **else**
19: $a_i.motivation- = GAIN$
20: **end if**
21: $success2 = NEGOTIATE(a_j, a_i)$
22: **if** $success2$ **then**
23: $a_j.motivation+ = GAIN$
24: **else**
25: $a_j.motivation- = GAIN$
26: **end if**
27: **end for**
28: $N+ = 1$
29: **end while**
30: **for** $(a_i, a_j) \in collisions$ **do** ▷ Remaining conflicts
31: **if** $mot(a_i) > mot(a_j)$ **then** ▷ Wait-based resolution
32: $wait(a_i)$
33: **else**
34: $wait(a_j)$
35: **end if**
36: **end for**
37: **end function**

The decision-making process focuses on cost minimization, based on a cost function. Thus, suboptimal solutions are chosen locally for each agent, based on the proposals of the other. Negotiations, as well as decisions, are refined throughout the voting process to reach an appropriate consensus. In the event that negotiations are unsuccessful within the allotted time, for any remaining collision between neighbors, a wait for the pedestrian most motivated to wait is imposed (lines 30-35).

3.4.3 The negotiation

We argue here the negotiation process between two agents a_1 and a_2, formally proposed in the Algorithm 3 (cf.[3] for used acronyms).

Suppose a collision between a_1 and a_2 is looming, a_1 enters into negotiation with a_2 — without loss of generality — asking a_2 to consider its path in its trajectory. Before the negotiation itself, a_2 will already have constructed a set $\mathcal{T}_2$ (in the way of line 5) of different trajectories (alternatives) to $goal(a_2)$ with associated costs $C_{T_i}(a_2)\ \forall T_i \in \mathcal{T}_2$, and where T'_2 aims to be the best trajectory

[3] CF: collision-free; alt: alternative; $a_i.choose\_path()$ selects the path of minimum cost in $\mathcal{T}_i$; $a_i.recompute\_set(obs)$ calculates A^* paths with and without waiting, taking obs obstacles into account to build the set $\mathcal{T}_i$.

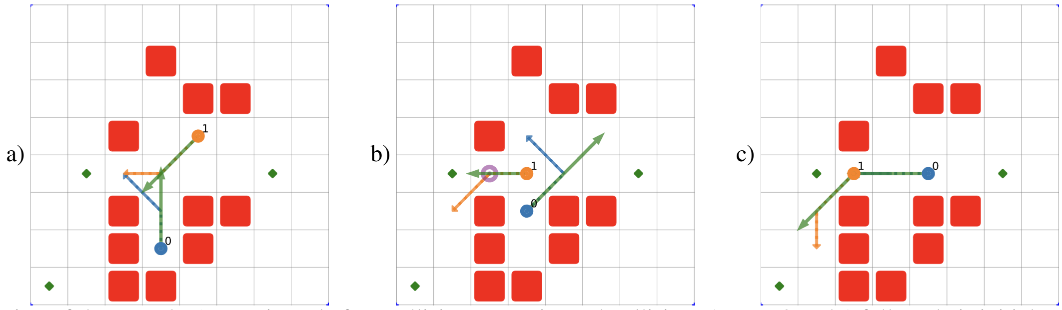

Figure 2: Simulation of the *Map 2*. a) Iteration i before collision, no estimated collision. Agents 0 and 1 follow their initial path. b) Iteration $i+1$, estimated potential collision in double purple circle: agents 0 and 1 have negotiated and 0 is considering an alternative. c) Iteration $i+2$, agent 1 follows its path and agent 0 recomputes its optimal trajectory.

Algorithm 3 NEGOTIATE

Require: Two different agents a_1, a_2

1: $\triangleright$ $cost(path(a_i))$ is sent with $path(a_i)$ to a_2
2: **function** NEGOTIATE(a_1, a_2) $\triangleright$ a_2 considers a_1's path
3:     $T_2' \leftarrow a_2.get\_best\_CF\_alt(path(a_1))$
4:     **if** T_2' not found **then** $\triangleright$ New CF-trajectory or best
5:         $a_2.recompute\_set(NEIGH\_OBST)$
6:         $\triangleright$ a_2 recomputes its set $\mathcal{T}_2$
7:         $T_2' \leftarrow a_2.get\_best\_CF\_alt(path(a_1))$
8:         **if** T_2' not found **then**
9:             $T_2' \leftarrow a_2.get\_best\_alt(path(a_1))$
10:         **end if**
11:     **end if**
12:     **if** $T_2' == path(a_2)$ **then return true**
13:     **end if**
14:     $worth\_value \leftarrow cost(path(a_1)) - cost(T_2')$
15:     **if** $worth\_value > a_2.motivation$ **then**
16:         $new\_cost \leftarrow cost(T_2') - worth\_value$
17:         $a_2.reconsider\_cost(T_2', max(\epsilon, new\_cost))$
18:         $a_2.choose\_path()$
19:         **return true**
20:     **end if**
21:     **return false**
22: **end function**

---

adapting to that of a_1. With these basics, we define our negotiation.

<u>*Def. Negotiation*</u>: A negotiation is a situation of influence involving two agents who have the **capacity to communicate implicitly or explicitly with each other, i.e., to exchange information (HYP1)**. In the algorithm, the cost $cost(path(a_1))$ is therefore communicated to a_2 (e.g., lines 3 and 14), in addition to a_1's path $T_1 = path(a_1)$, T_2' being the item of negotiation. Agent a_2 then seeks to optimize his preferences, i.e., minimize his cost function (lines 3, 7 and 9) for this collision, by selecting a path of minimum cost in $\mathcal{T}_2$.

(HYP2) All agents in this system aim to avoid collision. a_2 considers the negotiation of a_1 by looking for an alternative trajectory $T_i \in \mathcal{T}_2$ without collision with T_1 (line 3), if this is already its path, then the negotiation concludes successfully (line 12). Otherwise, a_2 attempts to calculate new alternatives by considering neighboring obstacles and a_1's trajectory (lines 5 and 7) to update T_2', its potential new trajectory. On failure, a_2 selects its best alternative with potential collision with a_1 (line 8-9). In the event of an unresolved collision, one of the two pedestrians will wait (cf. Alg. 2).

(HYP3) Each agent compares its cost with that communicated by the other and tolerates a certain degree of adaptation of its choices in favor of the other. This assumption underlines the collective rationality of pedestrians, in that it will benefit everyone if each pedestrian acts under the same postulate [16]. Each agent will therefore adjust its trajectory in coordination with the other, while minimizing the cost of change, and can refuse to be influenced if it is not worth it: the negotiation ends in failure (line 21). In lines 14 to 18, the motivation expresses a_2's willingness to tolerate a potential change of trajectory by lowering the cost of T_2' in order to possibly choose it in line 18. If so, the negotiation is successful (line 19).

In the end, both agents made their choice of trajectory, eventually avoiding collision with the other. Negotiation here is a cognitive process fueled by social interactions between pedestrians. It is therefore repeated during the associated vote in order to guarantee the relevance of the exchanges and to fructify the resolution of spatial conflicts: the success or failure of a negotiation is reflected in the vote by a change in the $motivation$ of the pedestrians, implicated in the costs $C_T(a_i)$ (1) and therefore in the negotiation (line 15). A negotiation will always conclude with the choice — potentially unchanged — of two trajectories for the agents, in order to guarantee movement for the next step. In particular, the final choice, i.e., the trajectories chosen, may involve a waiting process for one of the agents, especially in the event of physical blockage. This is a disadvantageous situation for the agent, but one that respects HYP3.

3.5 *An example of simulation*

We consider custom-made maps and detail an execution for one. Figure 2 illustrates three stages over which two agents negotiate to find respective trajectories suitable for avoiding a potential collision in *Map 2*. We display potential collisions as double purple circles, which appear when a collision between two agents is imminent. In this example, we consider attention at 100%. For this execution, agent 0 considers an alternative after agent 1 has successfully negotiated. Here, the cost of an alternative for agent 1 was more expensive than for agent 0, so agent 0 gave in. There is another possible execution where agent 1 gives in, depending on the characteristics generated at initialization. Note that $motivation$ prevails over randomness, as it is refined through negotiation. We will see below that this makes it possible to balance traffic.

4 Experimentation

CNM is developed in Python with the Mesa library[4] allowing for easy simulation of MAS in a discrete-time grid. We propose an open-source simulator for experiments (the git url will be provided if the paper gets accepted).

[4] https://mesa.readthedocs.io/

4.1 *Configurations*

We consider *maps* of various sizes (up to 18 x 12) with different configurations. ***Maps* correspond to scenarios of increasing complexity, considering topology and number of agents.** Simulations are carried out on a local MacBook M3 Pro machine with 36 GB ram. Given the size of the *maps*, simulations have been limited to a maximum of 50 iterations, a number far greater than the distances covered. Otherwise, the simulation is stopped if all agents have reached their goal. Simulations were run for each *map* 1000 times to average the results. A video presenting some simulation clips is available as supplementary material.

4.2 *Evaluation*

4.2.1 *Metrics*

Let a_i be an agent in a *map*, and its *resulting* trajectory is denoted T_i. Consider $T_i^+ = [(x_1, y_1), ...(x_{size(T_i^+)}, y_{size(T_i^+)})]$ its path T_i *without waiting*, i.e.,

$$\forall k \in [1, size(T_i^+) - 1], (x_k, y_k) \neq (x_{k+1}, y_{k+1})$$

We note T_i^* the optimal trajectory computed with A^* of a_i *without* the presence of the other agents. Finally, we note τ_i the time it takes a_i to reach its goal in the simulation, starting from the beginning, with its path T_i. We will thus use the following three metrics, applied to the set of pedestrians A in a given system.

Distance metric

$$dist(A) = \frac{1}{|A|} \sum_{a_i \in A} \frac{size(T_i^*)}{size(T_i^+)}$$

Wait metric

$$wait(A) = \frac{1}{|A|} \sum_{a_i \in A} (\tau_i - size(T_i^+))$$

Inequity metric

$$inequity(A) = std(wait(A))$$

The distance metric $dist(A)$ is the average of the ratio between the length of the optimal trajectory calculated with A^* and that of the realized path. The metric $wait(A)$ is the average of pedestrian waitings, and $inequity(A)$ represents the standard deviation of pedestrian waitings, a pragmatic fairness.

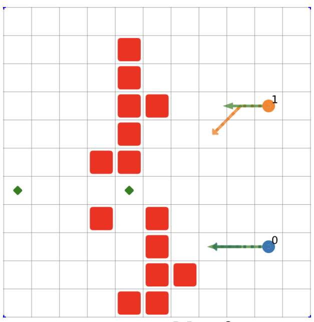

Figure 3: *Map 3*

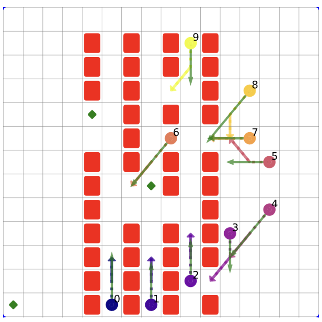

Figure 4: *Map 5*

4.2.2 *Results*

The results are presented as curves in Figure 7, which displays results of $dist(A)$ and $wait(A)$ as curves, with $inequity(A)$ shown as error bars. The mean execution times are provided in Figure 8.

We compare CNM with two other approaches, SFM (social forces) and CBS-MAPF (planning with waiting times), cf. section 2.

As SFM is not initially designed for discrete environments, we adapt the algorithm so that a pedestrian moves one cell in its direction if the sum of the calculated forces $F \leq 1$ is greater than a certain threshold. To this end, for each simulation, we test a threshold $threshold \in [0.1, 0.2, ..., 0.9]$ for F and retain only the averaged results with the best performance. In addition, pedestrians have a margin of one cell to reach their target, being subject to approximation problems. Finally, as SFM has no default planner, we include intermediate objectives on the *maps* to guide pedestrians in SFM.

For the ad-hoc version of CBS-MAPF, the algorithm initially uses the offline planner *Space Time* A^* [20] to compute paths with static and semi-dynamic obstacles, i.e., pedestrians only block a cell at certain time steps. In order to have a fair comparison, we replace it with the classic offline A^*, including initial waitings to unblock congested situations. Note then that these are two suboptimal versions of the algorithms, in a dynamic application setting that may differ from the initial use in the face of CNM.

4.3 *Discussion*

In ad-hoc implementation, SFM shows $dist(A)$ performances 5 to 50 % lower than CBS-MAPF and CNM, which is expected given that the model is a discretized variant. CBS-MAPF's results stop at *Map 4*, as its switch to the A^* offline planner, less efficient than Space Time A^* for semi-dynamic obstacles, prevents it from calculating trajectories beyond this. Its centralized, offline nature limits its ability to find optimal solutions in complex environments.

Conversely, CNM takes a decentralized, online approach, generating suboptimal solutions by allowing pedestrians to move to temporary zones to smooth traffic flow. Despite this, CNM and CBS-MAPF maintain similar, near-optimal average distances. SFM shows a waiting $wait(A)$ two to five times higher than CBS-MAPF and CNM, and an inequity $inequity(A)$ four to seven times higher. CBS-MAPF guarantees near-optimal waiting and equity, with similar performance for CNM, both with relatively small inequity. After *Map* 4, CNM maintains waitings in average to less than five iterations, close to the optimum.

We highlight *Maps 3, 5 and 6*, illustrated in Figures 3, 4 and 1. *Map 3* features a contention point leading to high waiting, while *Map 5* requires careful planning due to the complexity of the terrain. Finally, *Maps 6 and 7* reflect classic pedestrian crossing scenarios according to SFM [9], testifying to marked waitings and emergent

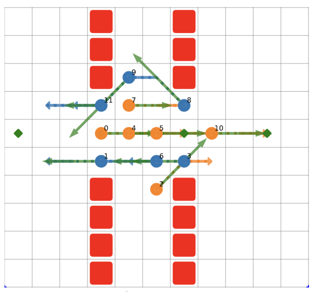

Figure 5: Queue emergence, *Map 6*

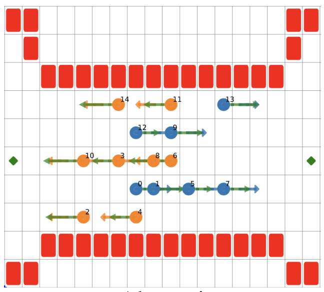

Figure 6: Alternating queues, *Map 7*

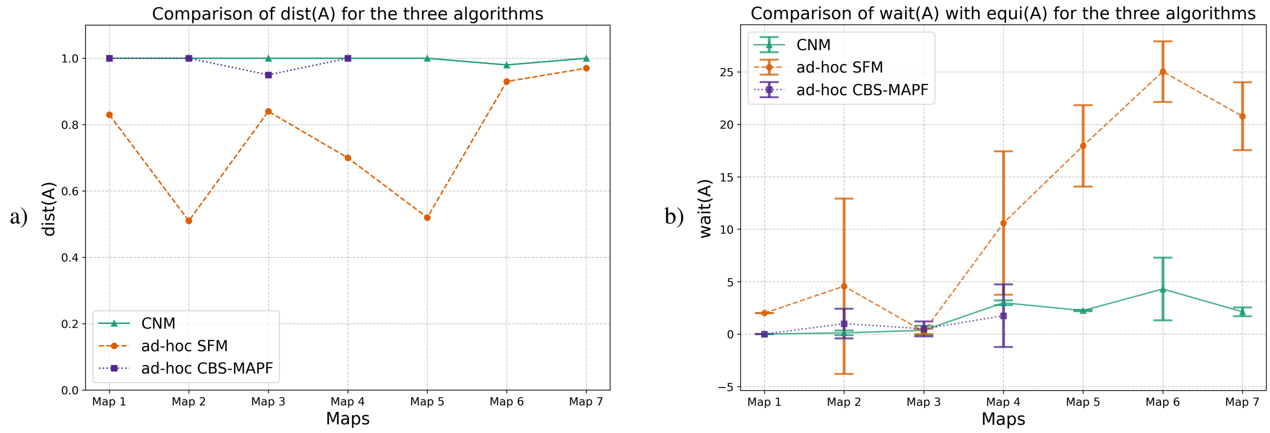

Figure 7: Metrics calculated on different maps using the CBS-MAPF, SFM and CNM algorithms. Values are averaged over 1000 simulations. CNM is shown as a solid green line, SFM as a dashed orange line and CBS-MAPF as a dotted purple line. a) Comparison of the $dist(A)$ metric on the three algorithms. A value close to 1 indicates that the computed paths are of similar length to those obtained with A^*. b) Comparison with the metrics $wait(A)$ and $inequity(A)$ (error bars). A value close to 0 indicates low waiting. A small error bar indicates a homogeneous waiting between agents. Each *map* contains a fixed number of agents, increasing from *Map* 1 to *Map* 7: 2, 2, 2, 4, 10, 12, 15 agents, respectively.

group organization, cf. Figures 5 and 6. CNM manages to generate the same pedestrian queuing phenomena as SFM in these scenarios, highlighting a compromise between time and organization. CNM also shows reasonable computation time between CBS-MAPF and SFM as shown in Fig. 8, giving potential to scale up the model. Globally, average execution times across the first four maps for SFM, CNM and CBS-MAPF are **0.005**, **0.02**, and **0.49 s**, respectively.

The different configurations highlight CNM's optimized dynamic pedestrian behavior and low waiting times. This demonstrates the robustness and relevance of the model for managing pedestrians in complex environments. CNM generates near-optimal trajectories, comparable to CBS-MAPF, while maintaining a decentralized and online nature. Unlike strictly optimal models, CNM aims to provide a cognitive model for pedestrians, who often fall short of optimality in real-world scenarios. A key phenomenon observed with CNM is that the least spatially constrained pedestrians tend to unblock complex situations. In a situation of congestion, pedestrians' $motivation$ increases (i.e., negotiations lead to no viable alternative) and the *freest* pedestrians (i.e., with more maneuvering space) adopt alternatives that are less favorable for them, but which streamline traffic.

In our experiments, the attention parameter in CNM was fixed at 100%. In our tool, attention can be manually enabled, assigning agents random attention probabilities at initialization. We compared CNM with attention (CNMw) and without (CNMwo) across all maps in Table 1. Performance differences between CNMw and CNMwo were minimal; however, CNMwo exhibited a 17% increase in mean execution time relative to CNMw. Conversely, CNMw resulted in approximately 10% greater agent waiting time. This outcome is expected, as the attention mechanism allows agents to ignore some negotiations, potentially imposing waiting for others. Finally, CNMw displayed a 38% greater inequity compared to CNMwo, suggesting that pedestrians who persist along their path contribute to a more balanced distribution of waiting times within the crowd.

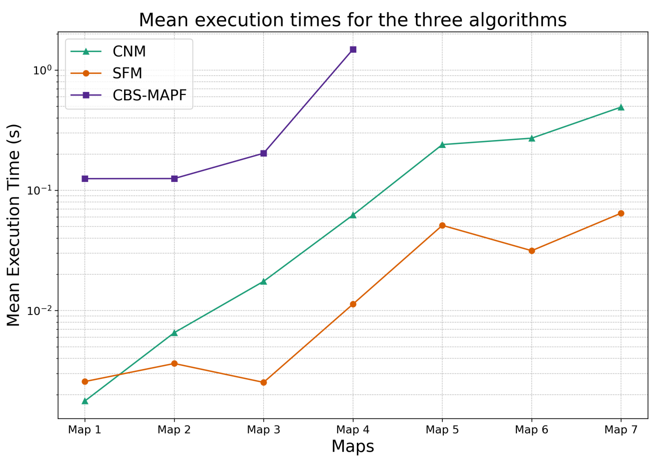

Figure 8: Mean execution times of the SFM, CBS-MAPF and CNM algorithms. The results have been averaged over 1000 executions for each *map*. The time values are given in logarithmic scale.

Version	Execution Time (s)	$dist(A)$	$wait(A)$	$inequity(A)$
CNMw	0.112	0.995	2.652	0.602
CNMwo	0.131	0.995	2.412	0.832

Table 1: Comparison of CNM with and without attention in average

5 Conclusion

We have proposed CNM (Conflict Negotiation Model), a cognitive model of pedestrian behaviors based on social choice, in particular negotiation. CNM provides a decentralized online algorithm for managing collective pedestrian movements, offering suboptimal but effective solutions in complex environments (i.e., with many obstacles limiting space access). By conducting extensive simulations, we observed that certain emerging behaviors, such as queuing and the interpretation of the movements of the least constrained pedestrians, are similar to those observed in real crowds. We compared our pedestrian model with a classic force model (SFM) and a well known multi-agent pathfinding algorithm (CBS-MAPF). Results showed that our negotiation-based model outperforms SFM in most scenarios and achieves performance close to CBS-MAPF without suffering from its computational costs. Overall, CNM shows pedestrians capable of adapting in real time, making it particularly suitable for simulating pedestrian flows in highly constrained urban contexts.

The model is still subject to improvements, such as agent perception, the *spontaneous* evolution of agents' preferences with attention — a feature of pedestrian cognition that filters certain interactions [24], influencing waiting behavior — or the calculation of the trajectories with partial knowledge to underline the randomness of human cognition. A natural perspective would be to evaluate CNM on real-life scenarios, accompanied by realism tests, e.g., Turing tests.

References

[1] K. Al-Kodmany. Crowd management and urban design: New scientific approaches. *Urban Design International*, 18:282–295, 2013.

[2] R. Amini, A. Dhamaniya, and C. Antoniou. Towards a game theoretic approach to model pedestrian road crossings. *Transportation research procedia*, 52:692–699, 2021.

[3] T. Bonnemain, M. Butano, T. Bonnet, I. Echeverría-Huarte, A. Seguin, A. Nicolas, C. Appert-Rolland, and D. Ullmo. Pedestrians in static crowds are not grains, but game players. *Physical Review E*, 107(2): 024612, 2023.

[4] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.

[5] A. Dommes and V. Cavallo. The role of perceptual, cognitive, and motor abilities in street-crossing decisions of young and older pedestrians. *Ophthalmic and physiological optics*, 31(3):292–301, 2011.

[6] J. Ferber and J. Müller. Influences and reaction: a model of situated multiagent systems. In *Proceedings of second international conference on multi-agent systems (ICMAS-96)*, pages 72–79, 1996.

[7] D. Foead, A. Ghifari, M. Kusuma, N. Hanafiah, and E. Gunawan. A systematic literature review of a* pathfinding. *Procedia Computer Science*, 179:507–514, 2021.

[8] M. Hartmann, M. Viehweger, W. Desmet, M. Stolz, and D. Watzenig. "pedestrian in the loop": An approach using virtual reality. In *2017 XXVI Int. Conf. on Information, Communication and Automation Technologies (ICAT)*, pages 1–8. IEEE, 2017.

[9] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[10] F. Johansson, A. Peterson, and A. Tapani. Waiting pedestrians in the social force model. *Physica A: Statist. Mechanics and its Appli.*, 419: 95–107, 2015.

[11] Ö. Kıbrıs. Cooperative game theory approaches to negotiation. *Handbook of group decision and negotiation*, pages 151–166, 2010.

[12] A. Lachapelle and M. Wolfram. On a mean field game approach modeling congestion and aversion in pedestrian crowds. *Transportation research part B: methodological*, 45(10):1572–1589, 2011.

[13] J. Liu, Y. Chen, and Y. Chen. Emergency and disaster management-crowd evacuation research. *Journal of Industrial Information Integration*, 21:100191, 2021.

[14] B. Mesmer and C. Bloebaum. Modeling decision and game theory based pedestrian velocity vector decisions with interacting individuals. *Safety science*, 87:116–130, 2016.

[15] F. Pascucci, N. Rinke, C. Schiermeyer, V. Berkhahn, and B. Friedrich. A discrete choice model for solving conflict situations between pedestrians and vehicles in shared space. *preprint arXiv:1709.09412*, 2017.

[16] A. Rautureau. *Winning is not everything-Human-like agents for tabletop games*. PhD thesis, UCLouvain, 2024.

[17] R. Saunders and J. Gero. Curious agents and situated design evaluations. *AI EDAM*, 18(2):153–161, 2004.

[18] A. Sen. Social choice theory. *Handbook of mathematical economics*, 3: 1073–1181, 1986.

[19] G. Sharon, R. Stern, A. Felner, and N. Sturtevant. Conflict-based search for optimal multi-agent pathfinding. *Artificial intelligence*, 219:40–66, 2015.

[20] D. Silver. Cooperative pathfinding. In *Proceedings of the aaai conf. on Artificial Intelligence and interactive digital entertainment*, volume 1, pages 117–122, 2005.

[21] E. Sisbot, L. Marin-Urias, R. Alami, and T. Simeon. A human aware mobile robot motion planner. *IEEE Transactions on Robotics*, 23(5): 874–883, 2007.

[22] R. Stern, N. Sturtevant, A. Felner, S. Koenig, H. Ma, T. Walker, J. Li, D. Atzmon, L. Cohen, T. Kumar, et al. Multi-agent pathfinding: Definitions, variants, and benchmarks. In *Proc. of the Int. Symp. on Combinatorial Search*, volume 10, pages 151–158, 2019.

[23] J. Švancara, M. Vlk, R. Stern, D. Atzmon, and R. Barták. Online multi-agent pathfinding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7732–7739, 2019.

[24] A. Tom, J.-M. Auberlet, and R. Bremond. Perceptive and cognitive process in the pedestrian decision-making: How do pedestrians cross at intersection? In *Proceedings of the Extra ICTCT Workshop*, pages 44–54, 2007.

[25] Y. Tong and N. Bode. The principles of pedestrian route choice. *Journal of the Royal Society Interface*, 19(189):20220061, 2022.

[26] A. Turnwald and D. Wollherr. Human-like motion planning based on game theoretic decision making. *International Journal of Social Robotics*, 11:151–170, 2019.

[27] F. Zanlungo, T. Ikeda, and T. Kanda. Social force model with explicit collision prediction. *Europhysics Letters*, 93(6):68005, 2011.