Article

Explicit Energy-Minimal Short-Term Path Planning for Collision Avoidance in Crowd Simulation

Saran Sillapaphiromsuk[a] and Pizzanu Kanongchaiyos[b,*]

Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand
E-mail: [a]saran.sil@student.chula.ac.th, [b]pizzanu.k@chula.ac.th (Corresponding author)

Abstract. In traditional crowd simulation methods, global path planning (GPP) and local collision avoidance (LCA) were mostly used to advance pedestrians toward their own goals without colliding. However, we found that using those methods in bidirectional flows can force a pedestrian to get stuck among the incoming people, walk through the congestion, or even unintentionally occupy in a dense area, although more comfortable passageway exists. These odd behaviors are usually produced and simply noticeable in bidirectional case. This paper aims at reducing these artifacts to achieve more behavioral fidelity, by adding the explicit metabolic-energy-minimal short-term path planning (MEM) in between GPP and LCA. For energy analysis, the optimal control theory with the objective energy function from the study of biomechanics was employed, which finally leads to the useful optimal walking characteristics for the pedestrians. The simulation results show that the pedestrians with MEM can adapt their moving to avoid the congestion, resulting in more promising lane changing and overtaking behaviors. Even though MEM was mainly developed to deal with the artifacts in bidirectional flows, it can be extended with a little modification and can produce significant behavioral improvement for multi-directional case as shown in the last part of the paper.

Keywords: Crowd simulation, path planning, metabolic energy, principle of least effort.

1. Introduction

Crowd simulation is a very important, challenging task in a production pipeline of games, films, and pedestrian analysis softwares. Behind such pipeline, several issues were taken into consideration with an intent to achieve high fidelity of behavioral realism, and, with no doubt, one of the most essential issues is crowd navigation.

In traditional crowd simulation, the *global path planning* (GPP) and the *local collision avoidance* (LCA) have been successively used to advance virtual pedestrians throughout the time. GPP takes responsibility to provide a set of collision-free paths among static obstacles. Such paths allow each pedestrian to walk freely between any two particular places in the simulated world, while LCA entirely copes with preventing collisions between pedestrians and obstacles by offering the sensory information to pedestrians so that they can dodge the neighbors while walking along the paths taken from GPP.

Simulating with GPP and LCA can produce satisfactory results. However, in case of bidirectional crowd flow, which can be seen everywhere in real life, for example, at the pavement and corridor, some autonomous pedestrians overtake the others in an awkward direction, which results in getting stuck among the incoming people or being in a congested area, even though there is another direction that could bring them into the more comfortable place. This is due to the lack of the consideration on the successive avoidance motion, which cannot be found in the GPP and LCA.

Contribution: In this paper, we present a metabolic-energy-minimal short-term path planning technique (MEM) to deal with such consideration. Our approach begins with the GPP to compute the collision-free paths among the static obstacles, then instead of directly doing the LCA, the MEM will find the best desired walking direction among the predicted walking paths of neighboring pedestrians. MEM does not consider only the current positions, but also the future. This prevents a pedestrian from the successive awkward motion. Finally the LCA employs such desired direction to find the actual one on the condition that the collision with the others should not occur.

The process to find the best desired walking direction inside the MEM comes from the principle of least effort [1] stating that it is a human nature to want the greatest outcome for the least amount of work. According to the principle, each virtual pedestrian in our approach is desired to walk on a path that causes the least amount of energy expenditure. This is definitely consistent with the real world, for example, people would not walk on a detour if there is a shortcut to the destination, and would not walk through a congested area if there is a clearer one, because walking either on a detour or through a congested area is inclined to fatigue people more than on a shortcut or in an open space.

To obtain such energy-minimal walking path, we transform the problem to the constrained optimization problem with the energy calculation based on the biomechanical study of the real human walking [2], and utilize the optimal control theory as a solver, which results in very useful characteristics of the energy-minimal walking paths in a dynamic environment. However, it is not trivial to find out the closed form solution to the problem so we present the approximation method which is practical and simple for implementaion.

Our results show the improvements in crowd navigation in bidirectional flow as shown in Figure 1, the red pedestrian chose to walk through the moving huge crowd via a clear passageway. This reflects the intelligence in his navigation. Moreover, our approach can automatically exhibit the lane formation phenomenon, which is an important behavior usually emerging in real life, and also the more promising lane changing and overtaking behavior as compared to the previous methods.

Organization: The remainder of this paper is organized as follows. In section 2, the previous work is surveyed. In section 3, we give an overview of our approach, and detail our MEM in section 4. In section 5, we demonstrate the simulation results and discuss the efficiency and impact on bidirectional crowd flow by comparing with the previous methods, and also show the usage of our approach on multi-directional crowd simulation. We summarize the paper and point out the future work in the last section.

Fig. 1. Simulation result generated by our approach. The red-colored pedestrian walks through the huge crowd split by a bent, narrow passageway. At each time step, he observes his surroundings and chooses a comfortable way allowing him to reach to the front area.

2. Previous Work

Computer-aided simulation of the creatures' behaviors dates back to the work of Reynolds [3] who proposed the model to simulate the movement of the flock of birds. Since then, crowd behaviors have been extensively studied by the researchers in different disciplines, and the plenty of approaches were then developed in an effort to imitate the pedestrian navigation, which will be briefly overviewed in this section.

Global Path Planning (GPP): Dealing with avoiding static obstacles has been much addressed in the robotics literatures where the robot is treated as an intelligent machine capable of sensing the surroundings and planning the collision-free trajectories. We refer the reader to the valuable book [4] for the literature review and the useful planning algorithms. Likewise, autonomous pedestrians need to recognize the simulated world, and plan for a route to the destination. The simple way is to discretize the simulated world into the single uniform grid and use the well-known A⋆ search algorithm for pathfinding, but it is inefficient for a large-scale environment in terms of computation time and memory used, so the multi-resolution grids [5] and the hierarchical pathfinding [6, 7] were introduced to enrich the performance. Owing to the tradeoff between the level of discretization and the performance, using a set of connected graphs to represent the walkable regions is a good choice to compromise between both of them. Many researchers construct such graphs based on different approaches, including the randomized method [8], Delaunay triangulation [9], navigation graph [10], voronoi diagram [11, 12], and medial axis [13, 14, 15, 16, 17]. The graph-based path planners yield a small-sized search space but the queried path, if it exists, is not the shortest one as produced in the grid-based planners. Moreover, the graphs can store additional information of each walkable region, for example, the crowd density [18], to be used as the heuristic value in the traditional graph search algorithms.

Local Collision Avoidance (LCA): Without neighboring people, an autonomous pedestrian will walk on a path queried in the GPP, but when others exist and are within close proximity, the pedestrian needs to dodge to avoid the collision. In cellular automata model [19, 20], preventing collisions is easily handled, but the behavioral realism must be sacrificed. Rather, the local field-based method [21, 22] computes the possible pathway on a discretized field centered at the pedestrian's current location. The rule-based method [23, 24, 25] uses a collection of predefined rules for pedestrian steering. It produces remarkable results in case of low-density crowds, but lack realism in high-density because of the ambiguity of the dominant rule selection. The force-based method [26, 27, 28] in which pedestrians are driven by repulsive and attractive forces, can reflect the social psychology of the human. However, the oscillation may occur due to improper weight setting. Instead, the velocity-based method [29, 30, 31] directly computes a set of admissible velocities, using geometric means, and then reasonably picks out one of them as a new velocity for the pedestrian. Some works [32, 33, 34, 35] enhance the Velocity Obstacles [36] to construct the better admissible velocity regions, resulting in collision-free, oscillation-free, smooth

trajectories, and parallel-computing capability. With regard to the realism, the example-based method [37, 38, 39, 40], psychology-based method [41, 42], and vision-based method [43] steer the pedestrians, based on the human movement video data, the human psychological factors, and the human visual perception, respectively. Moreover, the continuum crowds [44] and the fluid-like motion [45] can well demonstrate the crowd-level interaction between highly densely-packed crowds, where each individual movement undeniably complies with a group.

Uni- and Bidirectional Crowd Flows: In these situations, the GPP is easily defined due to the simplicity of static obstacle formation, which diverts the researcher interest to the local interaction between pedestrians. Some researchers simulate these circumstances using the lattice-gas model [46, 47] and cellular automata [48, 49, 50] with their own specific rules to determine the lane changing direction on a uniform grid. Although the rules were developed in different ways, the lane changing direction depends on the same attributes, including the crowd density and the walking directions of neighboring pedestrians being in frontal areas. Specifically, these rules will direct the pedestrians to walk on a more comfortable lane such as a low-density lane or a lane having the same-walking-direction pedestrians. By the nature of discretization, limiting pedestrian movement to a discrete set produces unrealistic results. Instead, the counterflow model [51] computes a new desired walking direction, based again on the crowd density and others' walking directions, enabling pedestrians in any continuous crowd simulators to walk toward a more comfortable area. Moreover, the overtaking analysis based explicitly on the social repulsive forces [52, 53] allows pedestrians to weave their way through a crowd, but the repulsive forces may cancel each other, causing a pedestrian to get stuck into a moving group in front even though walkable pathways are available.

The above-mentioned works determine a new walking direction pointing to a more comfortable lane or area, by considering merely the current state of the neighborings. This does not guarantee the forthcoming movements, and often results in strange-looking behaviors, e.g., confronting the oncoming people, getting stuck into pedestrians in front, or unintentionally being in a dense area, even though other pathways exist. As we point out, the GPP and LCA do not consider the successive walking motion so the awkward behaviors are supposed to emerge. Recently, the navigational system called the Effective Avoidance Combination Strategy (EACS) [54] presented a mid-term motion planning technique, like our MEM, to compute an energy-efficient avoidance path made of successive adaptations. But their resulting path does not guarantee the minimal energy. It depends on the order of collision testing.

In our approach, the presence of MEM produces different behavioral results, comparing to the previous methods using only GPP and LCA. Because in GPP and LCA no successive motions will be taken into account. Although some works exploit crowd density as a heuristic for guiding pedestrian walking, it is limited to some scenarios, for example, the scenario shown in Figure 1. Using crowd density for lane changing direction cannot guide the pedestrian to walk on a narrow passageway. Instead, pedestrians in our approach are guided by a collision-free path that yields the minimal energy. Our approach differs from EACS on the aspect that EACS may not consider some feasible paths because some orders of collision testing cannot be reachable so some paths may be skipped. But in our approach the energy-minimal collision-free path is computed from all feasible paths which guarantees that the resulting path yields the global minimum energy.

3. Overview of Our Approach

Our steering approach assumes that each pedestrian in bidirectional flow is already given the desired walking direction to one of the exits of the corridor, which can be simply defined by any GPP; for example, in the graph-based approach, the desired walking direction may be the tangent to a piecewise smooth curve at a point a pedestrian is corresponding to. Given the desired walking direction, the pedestrian then employs the MEM by firstly perceiving nearby people, secondly predicting their walking paths, and lastly planning for the energy-minimal path that compromises between the walking energy expended and the distance to destination. The energy-minimal path describes not only the spatial information but also the temporal one, which serves as the desired velocity at the present time
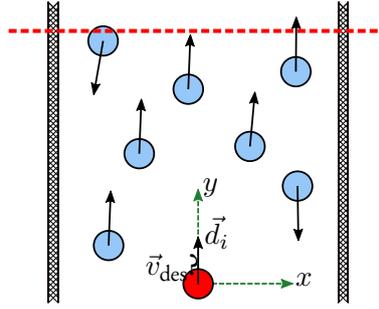
Fig. 2. The red pedestrian, who is located at the origin of the reference frame, is desired to walk toward the front line, or the red dashed line, with the lowest walking energy expenditure.

for the considered pedestrian. Finally, the LCA will exploit the desired velocity as an input to compute the actual one in the sense that the pedestrian attempts to walk along the energy-minimal path simultaneously with preventing collisions with the nearby walkers.

Algorithm 1 shows our simulation loop where the desired walking direction, the desired velocity, and the actual velocity of the ith pedestrian were represented by $\vec{d_i}$, $\vec{v}_{i,\text{des}}$, and $\vec{v}_{i,\text{act}}$ respectively. The $ComputeDesiredWalkingDirGPP$ function will be called in every time step to allow the pedestrian to anytime change his mind on the direction towards an exit of the corridor, but in case the direction is fixed, calling it once is enough. Our main contribution is in finding the energy-minimal path in the MEM, which performs through the $EnergyMinimalPathMEM(\cdot)$ function and will be detailed in the next section. The proposed MEM can be seamlessly connected with any previous LCA methods by using the desired velocity as a connector.

Algorithm 1 Crowd Simulation in Bidirectional Flow

1: **while** $simulating$ **do**
2:     **for each** $pedestrian\ i$ **do**
3:         $\vec{d_i} \leftarrow ComputeDesiredWalkingDirGPP()$
4:         $\vec{v}_{i,\text{des}} \leftarrow EnergyMinimalPathMEM(\vec{d_i})$
5:         $\vec{v}_{i,\text{act}} \leftarrow LocalCollisionAvoidance(\vec{v}_{i,\text{des}})$
6:         $UpdatePosition(\vec{v}_{i,\text{act}})$
7:     **end for**
8: **end while**

---

4. Metabolic-Energy-Minimal Short-Term Path Planning

Our MEM computes the desired velocity $\vec{v}_{i,\text{des}}$ for each pedestrian on the basis of the principle of least effort and the biomechanical walking energy. The problem we are dealing with was shown in Figure 2. Instead of planning a route to the one end of the corridor, the considered pedestrian i, which is depicted by the red circle in Figure 2, will plan its walking path toward the front line being far away from his current position with a distance specified by the user. We assume that the pedestrian i is positioned at the origin of the reference frame x-y and only responses to the perceived people in front.

The future position of the perceived people in front will be predicted by two nuanced ways, subject to the distance to the pedestrian i. If the distance is below some threshold l_{th}, the future position is obtained by linearly extrapolating its current velocity, otherwise the projected current velocity onto the desired walking direction $\vec{d_i}$ of the pedestrian i. The threshold l_{th} is set to 3.66 meters. This is the maximum distance of the social space [55] where the direct interactions and responses between the real humans happen. For the people in front beyond the social space (farther than the threshold l_{th}), prediction by using the projected velocity onto the desired walking direction $\vec{d_i}$ is reasonable enough

in case of bidirectional crowd flow where two pedestrians have a high tendency to meet each other at a future time, if extrapolating the current velocity is used instead, the future position at a large time period may lie outside the corridor and cause no influence on the pedestrian i.

The measurement for the walking energy is solely based on the biomechanical study of the real human walking [2] in which the oxygen uptake of a subject walking on a treadmill at varying speeds was recorded, resulting in a mathematical equation that manifests the relationship between the instantaneous metabolic energy expenditure and the walking speed, as shown below.

$$\frac{dE}{dt} = m(e_s + e_w\|\vec{v}(t)\|^2) \tag{1}$$

where E is the total metabolic energy measured in joules (J), m is the mass measured in kilograms (kg), $\vec{v}(t)$ is the velocity at time t measured in m/s, e_s and e_w are the constants measured in J/kg/s and Js/kg/m^2 respectively, and $\|\cdot\|$ is the Euclidean norm. The constant e_s and e_w can be viewed as the rates of the energy expended while standing and walking, respectively. These constants are unique for each pedestrian, and for the average human, the constant e_s is equal to 2.23 J/kg/s while the constant e_w is 1.25 Js/kg/m^2.

According to Eq. (1), we can compute the total metabolic energy expended by the pedestrian i over arbitrary time period Δt by using the following equation.

$$E = m \int^{\Delta t} (e_s + e_w\|\vec{v}(t)\|^2)dt \tag{2}$$

The pedestrian i will plan its walking path from the current time t_c to the unknown future time t_e (time at which the pedestrian i reaches the front line), so $\Delta t = t_e - t_c$.

4.1. Constrained Optimization Problem

From the principle of least effort [1], the pedestrian i is supposed to walk with the least amount of energy expenditure. So we compute the desired velocity $\vec{v}_{i,\text{des}}(t)$ for the pedestrian i over time period $[t_c, t_e]$ such that the total metabolic energy E is minimized:

$$\vec{v}_{i,\text{des}}(t) = \underset{\vec{v}_i(t) \in \mathcal{V}}{\arg\min} E \tag{3}$$

where $\mathcal{V}$ is a set of collision-free velocities over time period $[t_c, t_e]$. Although our objective function is similar to PLEdestrians [56], they are different in purposes. In PLEdestrians, a desired velocity is given and the energy-minimal actual velocity is then computed. Refer to Algorithm 1, PLEdestrians addresses the problem in $LocalCollisionAvoidance(\cdot)$ function.

To define a set of collision-free velocities $\mathcal{V}$, the mathematical representation of the front line and the perceived people in front must be well established. We ignore the boundary of the corridor momentarily to examine the energy-minimal walking characteristics of the pedestrian i against the perceived people. The front line and the jth perceived people at time t are illustrated by the implicit equations $f(\vec{p}) = 0$ and $w_j(\vec{p}, t) = 0$, respectively, where $\vec{p}$ is a point (x, y) in the reference frame. The geometric shape of jth perceived people is defined as a circle with the radius $(r_i + r_j)$, where r_i and r_j are the radius of the considered pedestrian i and jth preceived people. So

$$w_j(\vec{p}, t) \triangleq (r_j + r_i)^2 - \|\vec{p} - \vec{p}_j(t)\|^2 \tag{4}$$

where $\vec{p}_j(t)$ is the position of the jth perceived people at time t. For the future position of the jth perceived people, we linearly extrapolate its current velocity when the distance to considered pedestrian i below the threshold l_{th}, otherwise the projected one onto $\vec{d}_i$, so

$$\vec{p}_j(t) = \begin{cases} \vec{p}_j(t_c) + (t - t_c)\vec{v}_j(t_c), & \|\vec{p}_j(t_c)\| \leqslant l_{\text{th}} \\ \vec{p}_j(t_c) + (t - t_c)(\vec{v}_j(t_c) \cdot \vec{d}_i)\vec{d}_i, & \text{otherwise.} \end{cases} \tag{5}$$

With above definition, a velocity $\vec{v}_i(t)$ in a set $\mathcal{V}$ must conform to:

$$\dot{\vec{p}}_i(t) = \vec{v}_i(t)$$
$$w_j(\vec{p}_i(t), t) \leq 0, j = 1, ..., N \qquad (6)$$
$$f(\vec{p}_i(t_e)) = 0$$

where $\vec{p}_i(t)$ is the position of the pedestrian i at time t, and N is the number of perceived people. The first equation describes the motion of the pedestrian i, the second forces the pedestrian i not to collide with the perceived people, and the last one ensures that the pedestrian i must reach the front line at time $t = t_e$.

4.2. Metabolic-Energy-Minimal Walking Characteristics

The objective function in Eq.(3) and the constraints in Eq.(6) are investigated to compute $\vec{v}_{i,\text{des}}(t)$ for $t \in [t_c, t_e]$ by using the optimal control theory [57, 58, 59]. The position $\vec{p}_i(t)$ and the velocity $\vec{v}_i(t)$ are respectively the state and the control variable, which characterized by the pure state inequality constraints $w_j(\vec{p}_i(t), t) \leq 0$ and the free endpoint condition $f(\vec{p}_i(t_e)) = 0$. The result of the investigation provides us the following equations:

$$\vec{a}_i^*(t) = -\frac{1}{me_w} \sum_{j=1}^{N} \alpha_j(t)(\vec{p}_i^*(t) - \vec{p}_j(t)) \qquad (7)$$

$$\alpha_j(t) w_j(\vec{p}_i^*(t), t) = 0 \quad \text{and} \quad \alpha_j(t) \geq 0 \qquad (8)$$

$$\vec{v}_i^*(t_e) = -\frac{1}{2me_w} \left(\gamma \frac{\partial f}{\partial \vec{p}} + \sum_{j=1}^{N} \beta_j \frac{\partial w_j}{\partial \vec{p}} \right) \Bigg|_{\vec{p}_i^*, t_e} \quad \text{and} \quad \gamma \in \mathcal{R} \qquad (9)$$

$$\|\vec{v}_i^*(t_e)\| = \sqrt{\frac{e_s}{e_w} + \frac{1}{me_w} \sum_{j=1}^{N} \beta_j \frac{\partial w_j}{\partial t} \Bigg|_{\vec{p}_i^*, t_e}} \qquad (10)$$

$$\beta_j w_j(\vec{p}_i^*(t_e), t_e) = 0 \quad \text{and} \quad \beta_j \geq 0 \qquad (11)$$

where $\vec{a}_i(t)$ is the acceleration of the pedestrian i at time t, and the asterisk means the variable is computed at the optimal point. Eq.(7) and Eq.(8) explain the characteristic of the optimal acceleration that make the pedestrian i expend the (local) minimal metabolic energy. While Eq.(9) - Eq.(11) tell us about the optimal velocity $\vec{v}_i(t)$ at time $t = t_e$ (time at which the pedestrian i touches the front line f). For the derivation, please see the appendix A.

The energy-minimal walking characteristics that we can deduce from the equation (7) - (11) are:

Characteristic A.1: *For any time period when the pedestrian i walks without touching any perceived people, he must walk with constant velocity.*

Characteristic A.2: *For any time period when the pedestrian i touches one of the perceived people; in other words, walks on a circle boundary w_j, the relative velocity must be tangent to the circle, and the relative speed must be constant throughout the time when he is touching.*

A.1 and A.2 result from observing Eq.(7) and Eq.(8). Considering the walking characteristics at the time period when pedestrian i does not touch any perceived people at the optimal position $\vec{p}_i^*(t)$, we get $w_j(\vec{p}_i^*(t), t) < 0$ and from Eq.(8) we get $\alpha_j(t) = 0$ for all $j = 1, ..., N$. This results in zero acceleration $\vec{a}_i^*(t)$ (constant velocity) in Eq.(7) (A.1). Now considering if the pedestrian i touches the jth perceived people (walk on a circle boundary w_j), we get $w_j(\vec{p}_i^*(t), t) = 0$ and then $\alpha_j(t) \geq 0$. If $\alpha_j(t) = 0$, $\vec{a}_i^*(t) = 0$ and its velocity is constant. But the pedestrian i is assumed to walk on the circle boundary w_j in this period of time so this case cannot happen, forcing $\alpha_j(t) > 0$. When $\alpha_j(t) > 0$, the direction of $\vec{a}_i^*(t)$ will point to the center of the circle w_j due to the term $\vec{p}_j(t) - \vec{p}_i^*(t)$. Note that the velocity of the jth perceived people is constant as explained in Eq.(5), so $\vec{a}_i^*(t)$ can be viewed as a

relative acceleration of the pedestrian i against the jth perceived people, and since its direction points to the center, the pedestrian i will undergo the uniform circular motion on this period, and this results to A.2.

Characteristic B.1: *At the time t_e (when the pedestrian i reaches the front line f), if he does not simultaneously touch any perceived people, his velocity at that time must be perpendicular to the front line f, and his speed must be equal to $\sqrt{e_s/e_w}$.*

Characteristic B.2: *At the time t_e (when the pedestrian i reaches the front line f), if he simultaneously touches one of the perceived people, his velocity at that time depends on his state at the time before he reaches the front line f.*

B.1 and B.2 are the consequence of observing the boundary conditions at time t_e, as shown in Eq.(9) - Eq.(11). If the pedestrian i reaches the front line without touching any perceived people at time t_e, we obtain $w_j(\vec{p}_i^*(t_e), t_e) < 0$ and then $\beta_j = 0$ for all $j = 1, ..., N$ in Eq.(11). Therefore, the velocity $\vec{v}_i^*(t_e)$ in Eq.(9) must be parallel to the gradient of f (the normal of the front line f) at the position $\vec{p}^*(t_e)$, and its magnitude must be equal to $\sqrt{e_s/e_w}$ as depicted in Eq.(10). This results to B.1. In case he touches one of the perceived people at the time t_e, so $w_j(\vec{p}_i^*(t_e), t_e) = 0$ and $\beta_j \geq 0$, which gives $\vec{v}_i^*(t_e)$ the additional dependency on the gradient of w_j (the normal of the circle boundary w_j). As the position $\vec{p}_i^*(t)$ and the velocity $\vec{v}_i^*(t)$ are continuous for all time $t \in [t_c, t_e]$, and $\vec{p}_i^*(t_e)$ lies on the circle boundary w_j, if the position before he reaches the front line f, denoted by $\vec{p}_i^*(t_e - \epsilon)$ where ϵ is a small positive infinitesimal quantity, does not touch any circle boundary, the velocity $\vec{v}_i^*(t_e)$ will be characterized by A.1. But if $\vec{p}_i^*(t_e - \epsilon)$ lies on a circle boundary w_j, A.2 tells us that he is moving in a uniform circular motion at that time and keeps doing this until the time t_e, so the velocity $\vec{v}_i^*(t_e)$ will be characterized by A.2.

Characteristic C.1: *If there is no perceived people, the pedestrian i must walk straight with the constant speed $\sqrt{e_s/e_w}$ in the direction that is perpendicular to the front line f.*

Characteristic C.2: *If all perceived people stand still, the pedestrian i must walk with constant speed $\sqrt{e_s/e_w}$ throughout the time along the shortest path towards the front line.*

C.1 and C.2 explain the walking characteristics in special scenarios. C.1 simply deduces from A.1 and B.1, while C.2 from A.1, A.2, and Eq.(10) with the removal of the time derivatives of all w_j. Notice that the walking speed formula $\sqrt{e_s/e_w}$ matches the most efficient walking speed of the average human studied in the biomechanics [2], $\sqrt{2.23/1.25} = 1.34$ m/s.

4.3. Near-Global Optimal Solution

A velocity $\vec{v}_i^*(t)$ that conforms to the above walking characteristics is the local optimal solution to the problem. To find the global one, all possible $\vec{v}_i^*(t)$ need to be given but calculating each velocity $\vec{v}_i^*(t)$ analytically is not trivial because the transition at a point between A.1 and A.2 is restricted to be continuous. So we present the approximation method by replacing a circle w_j with two perpendicular lines, one is parallel to the x-axis of the reference frame, which we called a horizontal line, while the other one is parallel to the y-axis, called a vertical line. The intersection point of these lines is at the center of a circle w_j, and the endpoints of each line are at the middle between the corners of the inner and outer rectangles as shown in Figure 3. When the pedestrian i touches the jth perceived people in a period of time (A.2), he will walk between these endpoints instead of the circle boundary. If the endpoints are at the corners of the inner rectangle in Figure 3(b), the pedestrian i will think that he can walk through two adjacent perceived people, but actually he cannot. If the endpoints are at the corners of the outer rectangle, he will think that he cannot walk through, but actually he can. To compromise these situations, we choose to use the middle points instead.

With this approximation, the energy-minimal paths that conforms to the aforementioned walking characteristics turn out to be piecewise linear curves (a connected sequence of line segments) in space-time coordinate system of the pedestrian i as shown in Figure 4. We define without loss of generality that the state at time $t = t_c$ happens at $t = 0$ in this space-time coordinate, and the current position of the pedestrian i is $(x, y, t) = \vec{0}$. The horizontal and vertical lines generate two perpendicular planar

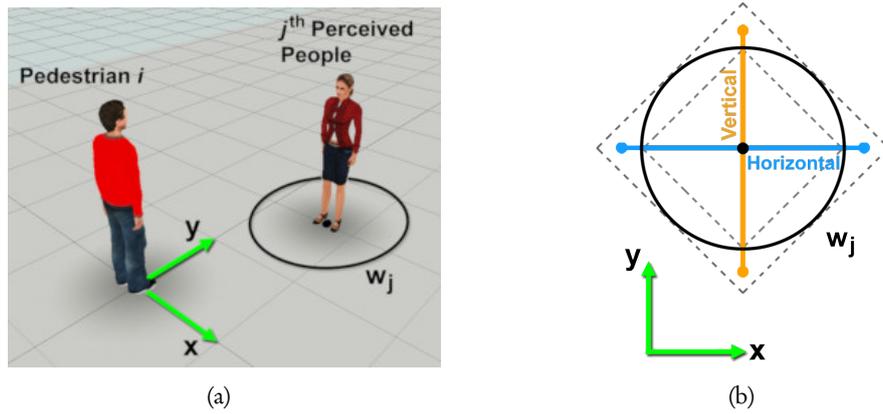(a)   (b)

Fig. 3. Replacing the circle w_j of the jth perceived people (a) with two perpendicular lines (b), called horizontal and vertical lines. While the pedestrian i touches the jth perceived people (A.2), he will walk between the endpoints of lines instead of the circle boundary.

strips. The front line f creates a plane parallel to the time axis, as shown in Figure 4(b).

To find the energy-minimal path, the time axis will be discretized into levels according to the user-defined sampling time Δt_{samp} and the maximum time t_{max} (Figure 4(c)). At each level, the endpoints of the horizontal and vertical lines are defined as the *critical points*, which perform as the transition points between A.1 and A.2. For the level $t = 0$, there is only one critical point locating at the origin ($\vec{\mathbf{0}}$). Instead of naively searching the energy-minimal path from the critical point at $t = 0$ to the plane f, we use the dynamic programming technique by finding the energy-minimal path starting at the critical points on the top-most level ($t = t_{\text{max}}$), and then the critical points on the lower level, until the critical point $\vec{\mathbf{0}}$ is reached.

For each critical point being examined, two types of line segments must be considered, based on the energy-minimal walking characteristics; (1) a $\sqrt{e_s/e_w}$-slope line segment from the critical point being examined to the closest point on the plane f (B.1), and (2) a line segment from the critical point being examined to the critical points on the higher levels (A.1 and A.2). A line segment will be selected as a candidate for constituting the energy-minimal path if no collision with the perceived people occurs, and thanks to each planar strip, the collision detection is very simple by checking only line-plane intersection. If the latest-examined line segment and its successor promote the lowest energy, such line segment and its successor will be stored at the critical point being examined, and they will be used as a successor for the critical points on the lower level. If the critical point $\vec{\mathbf{0}}$ is examined and there is no connected sequence of line segments from the critical point $\vec{\mathbf{0}}$ to the plane f, the pedestrian i will be given the desired velocity $\sqrt{e_s/e_w}\vec{d_i}$, but if there is a sequence (Figure 4(d)), the desired velocity will be computed from the first line segment (line attached to the critical point $\vec{\mathbf{0}}$) by the following equation:

$$\vec{v}_{i,\text{des}}(t_c) = (\overrightarrow{\mathbf{cp}}_1 - \overrightarrow{\mathbf{cp}}_0)_{xy}/(\overrightarrow{\mathbf{cp}}_1 - \overrightarrow{\mathbf{cp}}_0)_t \qquad (12)$$

where $\overrightarrow{\mathbf{cp}}_1$ and $\overrightarrow{\mathbf{cp}}_0 (= \vec{\mathbf{0}})$ are the critical points that constitute the first line segment.

4.4. Line Segment Pruning

To improve computational performance, line segments that agree with the following conditions will be pruned before checking the line-plane intersection: (1) one of the critical points at the end of a line segment lies inside a circle w_j, (2) the speed computed from the slope of a line segment exceeds the maximum speed of the pedestrian i, (3) a part of a line segment lies outside the corridor, (4) the successor of a line segment does not reach the plane f, (5) a line segment and its successor produce more energy than the previously-examined one, (6) in case the pedestrian i is restricted to plan his walking path only
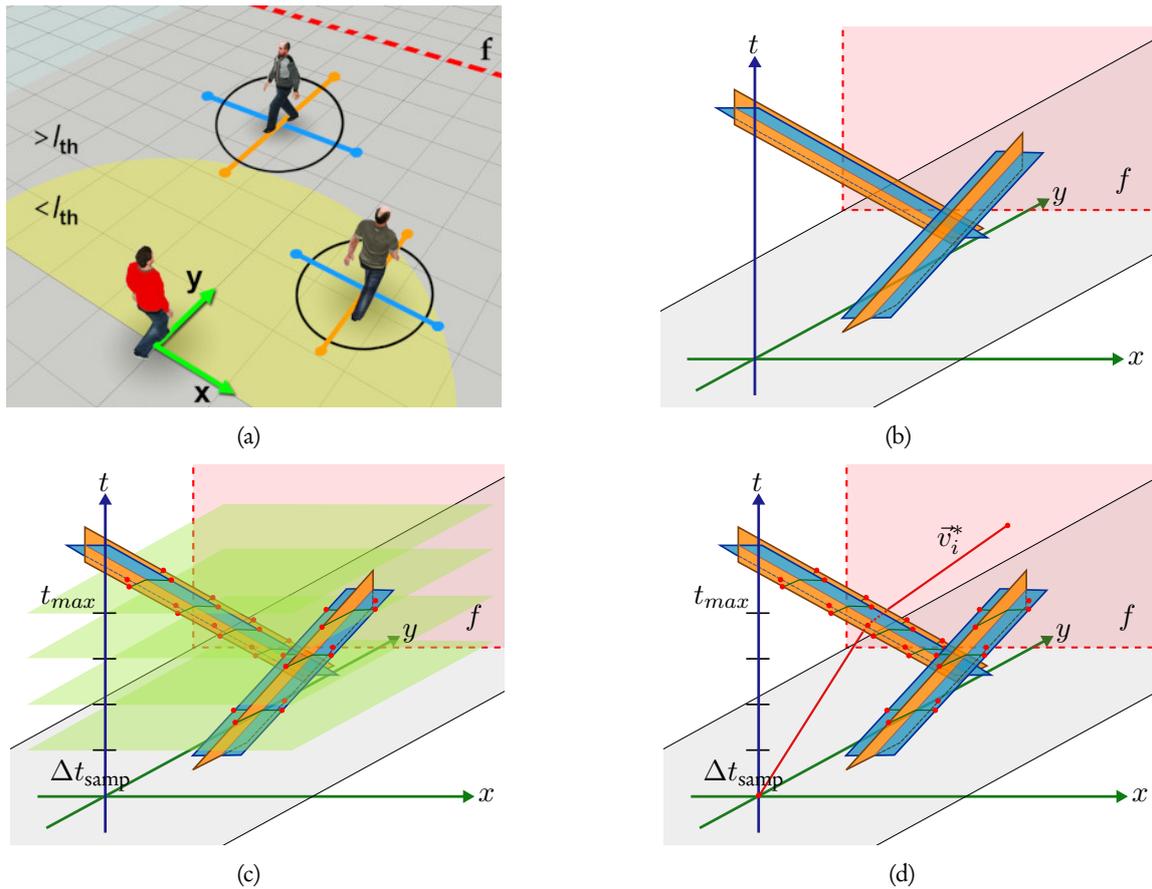
(a) (b)

(c) (d)

Fig. 4. Near-global energy-minimal velocity. (a) Pedestrian i perceived two people in front. The future position of the outside-l_{th}-zone pedestrian is predicted by extrapolating its projected velocity, and extrapolating the current velocity for the inside one. (b) Horizontal and vertical lines of each perceived people generate two perpendicular planar strips, and the front line f creates a plane in space-time coordinate system. (c) The sampling time and maximum time are defined to discretize the time axis into levels to construct critical points. (d) The energy-minimal velocity is computed from $\vec{v}_i^*$ obtained by the dynamic programming technique with the knowledge of the energy-minimal walking characteristics.

in the forward direction (the positive direction of the y-axis), a line segment that points to the negative direction of the y-axis will be pruned.

Moreover, if we found a $\sqrt{e_s/e_w}$-slope line segment from a critical point being examined to the closest point on the plane f, we can discard the line segments between levels since the $\sqrt{e_s/e_w}$-slope line segment produces the most minimal walking energy from the critical point being examined.

4.5. The Constant e_s, e_w, and t_{max}

As pedestrians walk at different preferred speed due to their own physiological attributes, for examples, a tall man naturally walks faster than a short one, we handle this diversity in our crowd simulation by setting the constant e_s and e_w for each virtual pedestrian on the assumption that virtual pedestrians expend the same amount of metabolic energy while standing but different while walking. That is:

$$e_s = 2.23$$
$$e_w = e_s/v_{\text{pref}}^2 \tag{13}$$

where v_{pref} is the preferred speed of a virtual pedestrian. The above assumption comes from the situation when a tall and a short man are at the same position and would like to walk to the same location on the condition that they must reach that location at the same time. The tall man naturally walks faster than the short one by his preferred speed, making the tall man expend the lowest energy, whereas the short one must accelerate himself to pursue the tall man, making the short man walk with higher speed than he prefers, so the short man must spend more energy while walking (e_w) for the instantaneous acceleration, and this conduces to Eq.(13).

For the time $t_{\max}$, it should be equal to or greater than the time t_e to secure the energy-minimal walking path towards the front line f, however, the time t_e cannot be known in advance. Nevertheless, $t_{\max}$ must be greater than the time t_e in the situation when the pedestrian i walks without the perceived people in front (C.1), so the lower bound for $t_{\max}$ is $c\sqrt{e_w/e_s}$, where c is the minimum distance to the front line f. For the upper bound, we define it from the situation when the pedestrian i is closely obstructed by all perceived people which horizontally-packed into a single row, so the pedestrian i must walk to the left or right to avoid the perceived people before walking straight to the front line f. Therefore,

$$c\sqrt{\frac{e_w}{e_s}} \leq t_{\max} \leq \left((\text{largest}_{r_j} + r_i)N + c\right)\sqrt{\frac{e_w}{e_s}} \tag{14}$$

where largest_{r_j} is the largest radius among the radii of the perceived people.

5. Results and Discussion

In this section, we will show the results through a set of scenarios, and discuss the efficiency by comparing with the previous work and the real-world bidirectional crowd flow. We implemented our work in C++ and used OpenGL for visualization on a 64-bit machine with 8GB of RAM, an Intel i7-2600 3.40GHz processor, and with the NVIDIA Geforce GTX 550 Ti. All simulation results are demonstrated in the companion video.

Lane Changing and Overtaking Behavior: The first scenario was shown in Figure 5(a) where the red and yellow pedestrians are walking upward with the desired speed 1.5 m/s and 0.5 m/s respectively, and the three green pedestrians are walking downward with 1.3 m/s. The initial position of the red pedestrian slants a bit to the right side of the yellow one. We compare our method with the social force model, the velocity-based PLEdestrians, and the counterflow model. Since the original social force and the velocity-based models allow the simulated pedestrians to perceive the people in back, causing the people in front to be pushed and/or sidestep, these behaviors should not occur in normal situation of bidirectional crowd flow so we modified by restricting the visual angle to 180 degrees. For our approach, the front line f is set to be a straight line $y - c$ where c is the planning distance. In this scenario, we set $c = 7$m and the perception radius is equal to c. The $t_{\max}$ is set to the halfway between the lower and upper bound defined in Eq.(14), and the $\Delta t_{\text{samp}} = 0.25$s.

The results show that If the group of green people does not exist, the red pedestrian in all methods will overtakes the yellow on the right-handed side, but when the green exists, the red pedestrian in the social force model, the PLEdestrians, and the counterflow model still tries to overtake the yellow on the right-handed side and eventually get into trouble among the green people. This is because in the social force and PLEdestrians, the desired velocity was fixed, which always points to the one end of the corridor, and the resulting walking direction was computed based on the simple actions like walking right when someone being left, which produces the awkward results as shown in Figure 5(b) and 5(c). Although in the counterflow model the desired velocity was recomputed in every timestep, it was obtained in a greedy fashion by considering only the present states of the perceived people occupying in three overlapped front areas, and choosing a predefined walking direction towards a lowest-cost area. This still exhibits an awkward behavior as shown in Figure 5(d). In our MEM (Figure 5(e) and 5(f)), the red pedestrian still tries to overtake the yellow on the right-handed side but when he perceives the green ones, he changes his direction to the left to reach more comfortable area.
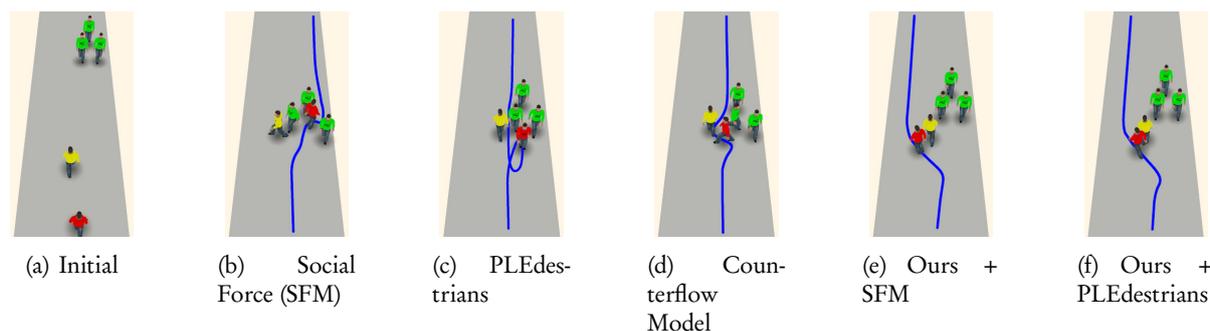
(a) Initial	(b) Social Force (SFM)	(c) PLEdestrians	(d) Counterflow Model	(e) Ours + SFM	(f) Ours + PLEdestrians

Fig. 5. From the initial (a), the red pedestrian in the social force model (b), PLEdestrians (c) and Counterflow model (d) walks towards the people in front in an uncomfortable manner by approaching to the incoming pedestrians although the left area is available. However, the red pedestrian with our MEM ((e) and (f)) can pass through the crowd on the left area, which is collision-free and more comfortable.

The second scenario was shown in Figure 6. The red, blue, and yellow pedestrians are walking upward with desired speed 2.0 m/s, 1.3 m/s, and 0.5 m/s, respectively, and the green ones are walking downward with 1.3 m/s. The front line f, t_{max}, and Δt_{samp} for the red pedestrian are identical to the previous scenario except that $c = 10$m. The results show that with the social force model (Figure 6(b)) the red pedestrian walked straight towards the group of green people and then was pushed back before escaping to the left, whereas with the PLEdestrians (Figure 6(c)) and the counterflow model (Figure 6(d)), the red pedestrian immediately turned his walking direction to the left when he perceived the green but he afterwards got stuck between the two slow yellow pedestrians. This was due to the opposed influences produced by each yellow pedestrian. If perceiving the people in back is allowed, the two yellow pedestrians will either be pushed or sidestep so that the red one can walk through, which should not be occurred since the adjacent lanes/areas are available.

With our MEM, the red pedestrian walks to the left to avoid the incoming green people, and then overtakes the yellow on the left and finally the blue people on the right, as shown in Figure 6(e) and 6(f). In case the desired speed of the red pedestrian decreases from 2.0 m/s to 1.3 m/s, he still avoids the incoming green people in the same direction as before, but this time he chooses to overtake the yellow on the right, as shown in Figure 6(g), because this path is collision-free, shortest, and energy-minimal for the desired speed 1.3 m/s. On the other hand, when the desired speed increases to 3.0 m/s, he walks straight towards the incoming green people and passes through the crowds via a collision-free gap as shown in Figure 6(h). These behavioral varieties reflect the intelligence in his navigation.

Mimick the Real-World Bidirectional Flow: We also mimicked the real-world bidirectional crowd flow by using our MEM along with the PLEdestrians for LCA, as shown in Figure 7 where the top rows show the image sequence from the video footage of bidirectional crowd flow, and the bottom rows show our mimicking results. Each simulated pedestrian has its own front line f with different planning distance c, and the t_{max} and Δt_{samp} are identical to the previous scenarios. After setting and tuning for c, the simulated pedestrians performed in the same manner as ones in the captured video, which can be seen from the movement of the rectangle-marked pedestrian in Figure 7(a) who runs fast in an upward direction (left column), then slows down for the expected gap (middle column), and finally accelerates to overtake the front people (right column), as well as the movement of the marked pedestrian in Figure 7(b) who walked fast (left column), then overtakes the people in front on the right (middle column), and finally goes through the crowd to the left (right column).

Lane Formation: One important phenomenon that inherently occurs in bidirectional crowd flow is lane formation. We conduct the experiment to observe this capability in our approach by placing a group of approximately one hundred pedestrians at each one end of the 13-meter-width corridor. The simulated pedestrians are placed randomly in the group and prefer to walk to the opposite side at the desired speed 1.3 m/s. The planning distance c and the perception radius are set to 6 meters for all

(a) Initial (b) Social Force (SFM) (c) PLEdestrians (d) Counterflow Model

(e) Ours + SFM (f) Ours + PLEdestrians (g) Ours + PLEdestrians (lower desired speed) (h) Ours + PLEdestrians (higher desired speed)
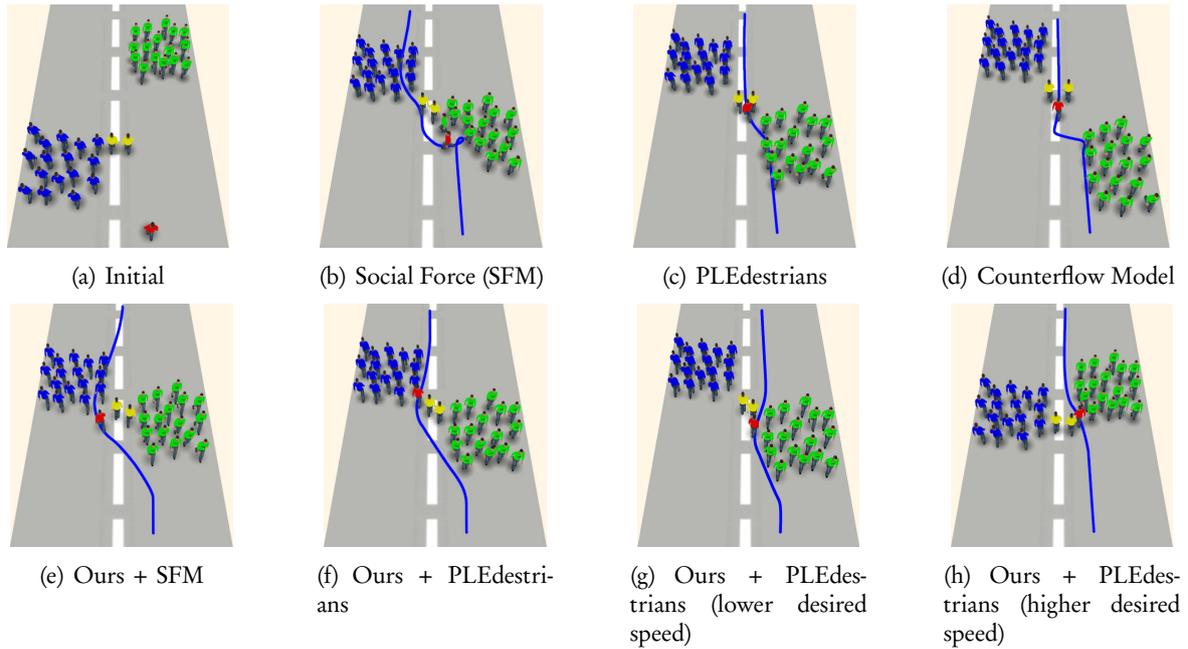
Fig. 6. From the initial (a), the red pedestrian in the social force model (b) approached to the incoming people and then was pushed back by the green pedestrians before escaping to the left, while in the PLEdestrians (c) and counterflow model (d) the red pedestrian got stuck between two yellow people. However, when equipped with our MEM ((e) - (h)), the red pedestrian can comfortably pass through the crowd on different energy-minimal paths depending on his desired speed.

(a) (b)

Fig. 7. Mimicking the real-world bidirectional crowd flows. The top row shows the image sequence (from left to right) of the real-world bidirectional flow, while the bottom row shows the simulation result generated by our approach. (a) The marked pedestrian runs towards the crowd with high speed (left column), then slows down (middle column), and finally overtakes the people in front (right column). (b) The marked pedestrian walks towards the crowd (left column), then overtakes the people in front on the right (middle column), and finally goes through to the left (right column).
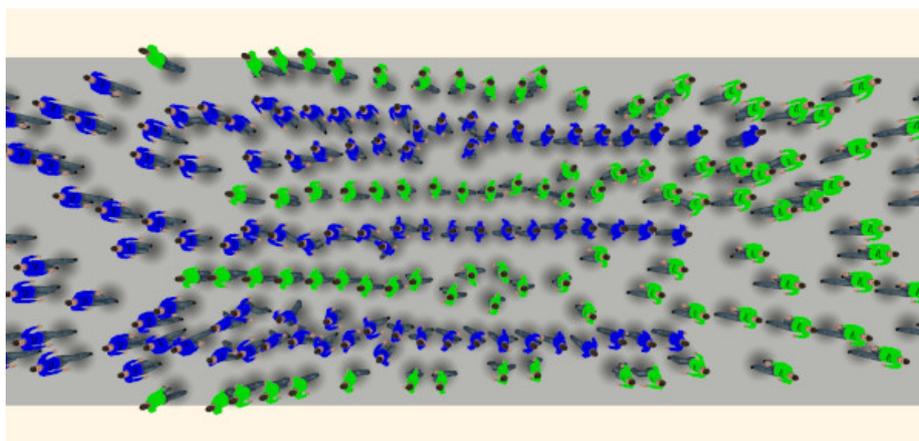
Fig. 8. Lane formation simulation from a group of approximately one hundred pedestrians at each one end of the 13-meter-width corridor. The simulated pedestrians are placed randomly in the group and prefer to walk to the opposite side at the desired speed 1.3 m/s.

pedesrians. The simulation result shows that after two groups meet each other around the middle of the corridor, the simulated pedestrians form 7 lanes as shown in Figure 8. If we use longer planning distance c, the lanes are formed faster. The lane formation was made in order to decrease the overall walking energy of the pedestrians, because following the people in front to reach the front line produces lower energy than facing the incoming people.

Fundamental Diagrams of Traffic Flow: We also quantitatively examine our approach in bidirectional crowd flows through the fundamental diagrams. The simulated pedestrians with the desired speed in a range from 1.3 m/s to 1.7 m/s are placed randomly in the 3m-wide and 15m-long corridor. The number of the pedestrians walking towards the right end of the corridor is equal to one towards the left end. If a pedestrian reaches to the one end, he will show up at the opposite and starts walking again. The t_{max} is set to be halfway between the lower and the upper bound defined in Eq.(14), $\Delta t_{samp} = 0.25$s, and the planning distance c is 3.66 m for all pedestrians. To obtain the fundamental diagrams, we measure the average speed $\bar{v}$ and crowd density ρ in three areas locating at the middle and the ends of the corridor. Given $\bar{v}$ and ρ, the specific flow J_s is computed by using hydrodynamic relation $J_s = \rho\bar{v}$.

The flow will be examined at different numbers of pedestrians ranging from 10 to 120 people (equivalently to 0.37 m^2 to 4.50 m^2 maximum occupation area for a single pedestrian). The crowd density and the average speed in each area will be measured every frame and averaged over a second interval. After running the simulation, the fundamental diagrams are obtained as shown in Figure 9. Notice that in case of 10 people (4.50 m^2/ppl) and 20 people (2.25 m^2/ppl), the walking speed of the pedestrians clings around the desired speed (1.3 m/s - 1.7 m/s) throughout the simulation time, but in case of 30 people (1.50 m^2/ppl) it sometimes a bit decreases due to higher population. When the number of pedestrians increases to 60, or the maximum occupation area per pedestrian reduces to 0.75 m^2, the walking speeds spread widely over the range from approximately 0.3 m/s to 1.6 m/s in a linearly-decreasing pattern as the density increases. This distribution results from dissolving the congestion into the free flow lanes. However, in a highly-dense crowd as demonstrated by the cases of 90 people (0.50 m^2/ppl) and 120 people (0.37 m^2/ppl), the free flow lanes are hardly constructed, which makes the pedestrians walk most of the time at the speeds ranging from approximately 0.15 m/s to 0.5 m/s. Given the specific flow, the relationship between the specific flow and the crowd density is shown in Figure 9(b), while the specific flow and the average speed shown in Figure 9(c). These diagrams have similar trend to the empirical data of bidirectional crowd flow [60] and the traffic flow theory [61].

Computation Time: The number of critical points, which depends on the user-defined parameters and the number of the perceived pedestrians, obviously has a great impact on the overall simulation time. To show the trend of the computation time on our approach, we measure the average time consumption in the situation when the pedestrians are randomly populated with different densities over the 3m-wide
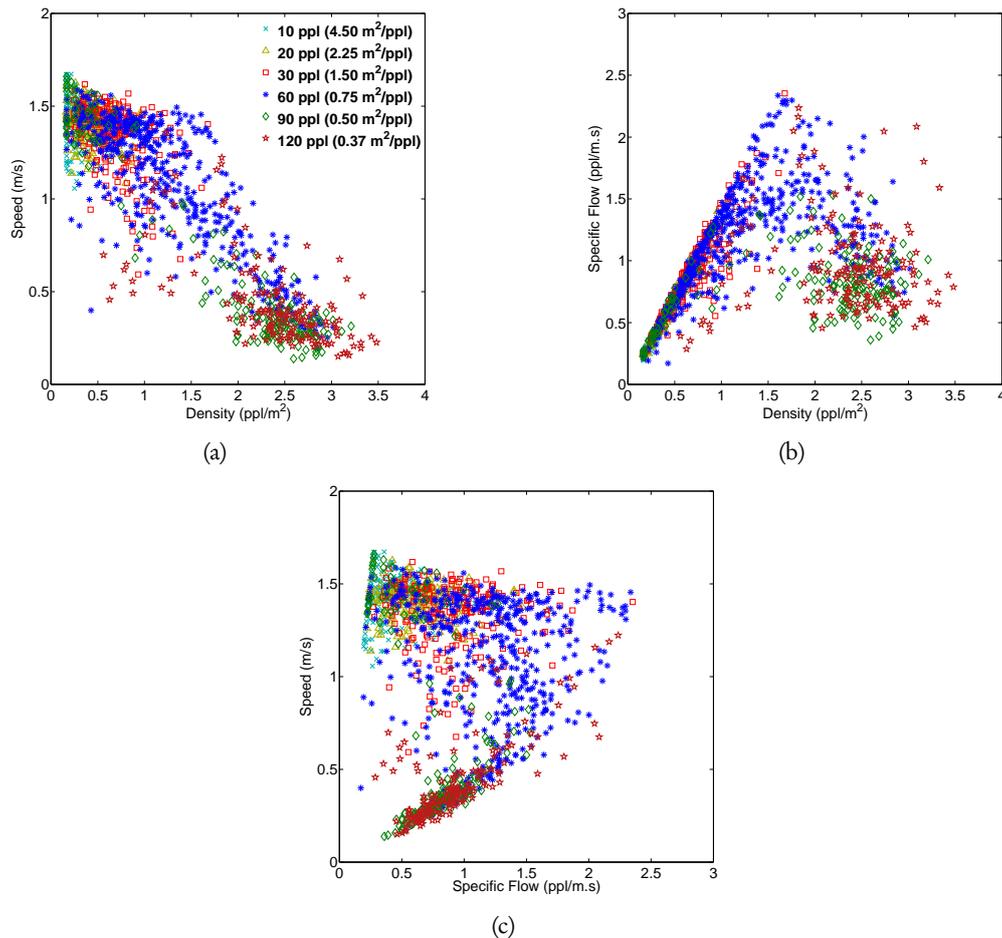
(a)

(b)

(c)

Fig. 9. The fundamental diagrams of bidirectional crowd flows generated by our approach.

and 15m-long corridor. The desired speed for each pedestrian is randomly set in a range from 1.3 m/s to 1.7 m/s. At a certain population density, three different setting for the user-defined parameters: (1) $c = 3.66$ m, $\Delta t_{samp} = 0.25$ s, (2) $c = 3.66$ m, $\Delta t_{samp} = 0.50$ s, and (3) $c = 5.00$ m, $\Delta t_{samp} = 0.25$ s, will be used for the quantitative comparison.

After measuring at 12 population densities ranging from 0.2 ppl/m^2 to 2.66 ppl/m^2, the trend of the average computation times has been produced as shown in Figure 10. It is not surprising that at a certain population density the average computation time increases as the planning distance c increases and/or the sampling time Δt_{samp} decreases because the increase of c and/or the decrease of Δt_{samp} cause the higher number of critical points. However, when observing their margins, the average computation times in all three settings are not significantly different at the population densities below 1 ppl/m^2, but dramatically expand at the higher population densities. If the long planning distance c with the precise time sampling Δt_{samp} is used, the computation time must be expensive in high-density crowds. This is the limitation of our approach if the real-time computation for the dense crowds is required.

Multi-Directional Crowd Simulation: Our approach can be used in multi-directional crowd simulation with a little modification. In bidirectional crowd flow, the pedestrian i is desired to walk towards the front line f but in multi-directional case the pedestrian i is desired to walk towards a circle with the radius specified by the user. The circle is centered at the pedestrian i's goal position. Figure 11 shows our simulation result in the scenario when two pedestrians try to walk across a flow of crowds. Pedestrians in the flow walk to the right with the same speed 1.3 m/s while two red pedestrians would like to walk to their own goals which located in the other side of the flow. Red pedestrian has a desired speed 2.0 m/s. The result shows that the red pedestrians can pass through the flow via the two tunnels
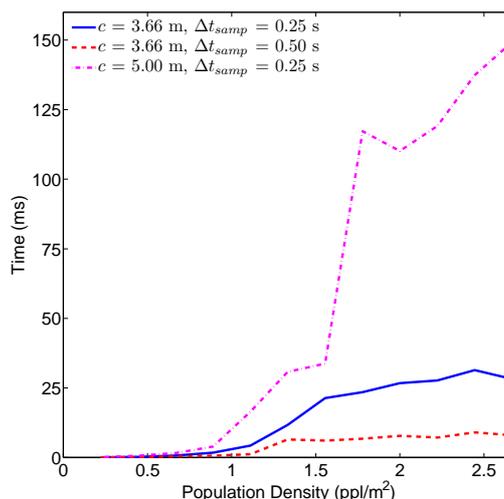
Fig. 10. The average computation time of our approach in three different setting at different population densities.

that are marked with the red-colored rectangles.

In the other example as shown in Figure 12, pedestrians with the desired speeds ranging from 1.0 m/s to 1.4 m/s are placed at two circle boundaries (Figure 12(a)), and the goal position for each pedestrian is located at the opposite. At frame 725 (∼12 seconds), the pedestrians in the social force model (Figure 12(b)), PLEdestrians (Figure 12(c)), and ORCA (Figure 12(d)) are mostly packed at the center, but when equipped with our MEM, the pedestrians are scattered and some of them almost reach to their own goal positions. This is because the pedestrians with our MEM respond to the others in an early time of simulation by planning the collision-free, comfortable paths towards their goals.

6.   Conclusion and Future Work

The short-term path planning based on the principle of least effort with the energy calculation using the metabolic energy equation of the real human walking was proposed. The technique can be seamlessly integrated with previous local collision avoidance methods, which allows the virtual pedestrians in bidirectional crowd flows to walk on energy-minimal paths. This results in more promising overtaking behavior and more reasonable lane changing direction, and in addition can achieve lane formation phenomenon and also generate the same trend of the fundamental diagrams as ones in the empirical data and the traffic flow theory. To obtain energy-minimal paths, we formulate the problem as the optimization problem and employ the optimal control theory with the dynamic programming as a solver. The algorithm can perform well in low-to-medium population density but yields the expensive computation in dense crowds. Also, our approach can be used for multi-directional crowd simulation with a little modification. For the future work, we plan to reduce the computational burden by finding the heuristic to determine when our MEM should operate, and the efficient method for adaptive planning distance c.
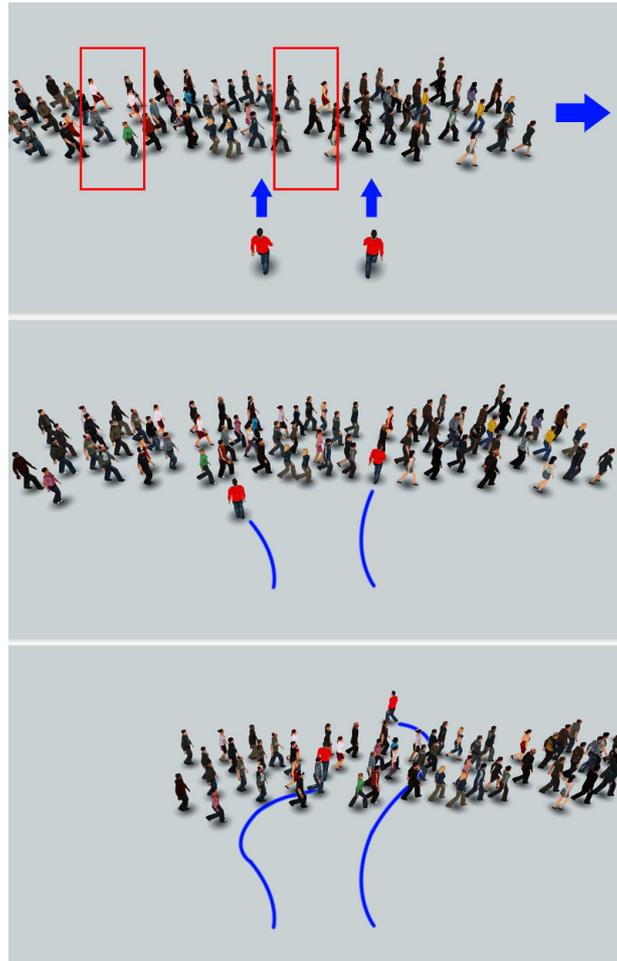
Acknowledgements

Fig. 11. Image sequence (from top to bottom) of two red pedestrians walking upward against a flow of crowds by using our approach. The red pedestrians can walk through the flow via the two tunnels marked by the red-colored rectangles.
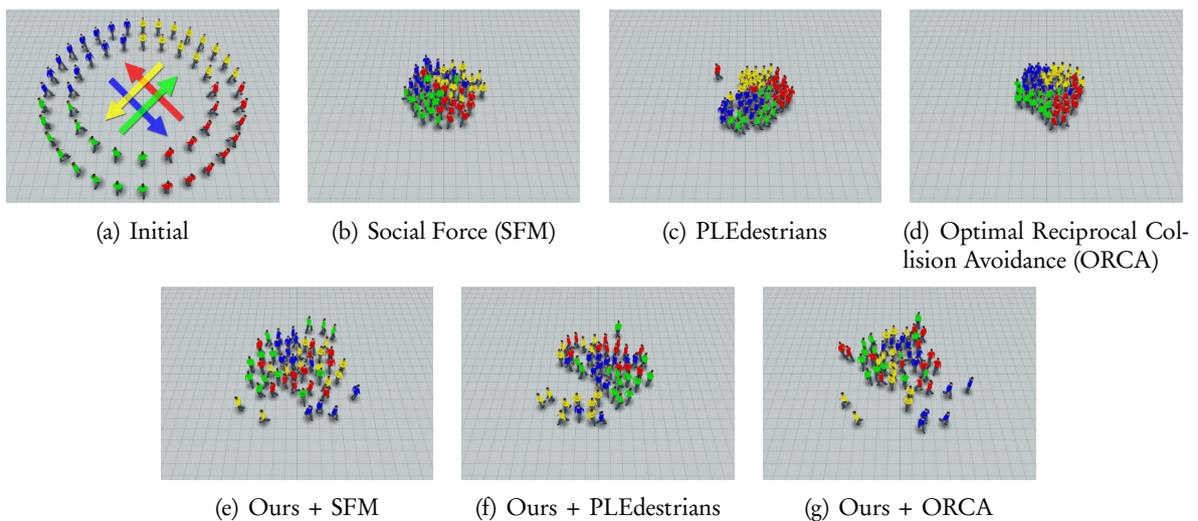

| (a) Initial | (b) Social Force (SFM) | (c) PLEdestrians | (d) Optimal Reciprocal Collision Avoidance (ORCA) |

| (e) Ours + SFM | (f) Ours + PLEdestrians | (g) Ours + ORCA |

Fig. 12. Multi-directional crowd simulation. The pedestrians are placed at two circle boundaries. At frame 725 (∼12 seconds), the pedestrians in the social force model (b), PLEdestrians (c), and ORCA (d) are mostly packed at the center, but when equipped with our MEM ((e) - (g)), the pedestrians are scattered, and some of them almost reach to their own goal positions

Appendix A   Derivation of Energy-Minimal Walking Characteristics

A.1   Optimization Problem

Our optimization problem begins with minimizing the objective function in the integral form as shown in the following equation:

$$E = \int_{t_0}^{t_1} g(t, \vec{p}(t), \vec{v}(t))dt, \tag{15}$$

where $\vec{p} \in \mathcal{R}^m$ and $\vec{v} \in \mathcal{R}^n$ are called the state and control variables, respectively, and $g(\cdot)$ is a real-valued function. When the values of the control variables $\vec{v}$ change, the values of the state variables $\vec{p}$ will be changed simultaneously by this differential equation:

$$\dot{\vec{p}}(t) = \vec{h}(t, \vec{p}(t), \vec{v}(t)), \tag{16}$$

where $\vec{h}(\cdot)$ is arbitrary vector function that has the same dimension as $\vec{p}$. The state $\vec{p}(t)$ is also constrained by the N inequality equations:

$$w_j(\vec{p}(t), t) \leq 0, \quad j = 1, ..., N \tag{17}$$

for all $t \in [t_0, t_1]$, and one endpoint constraint:

$$f(\vec{p}(t_1)) = 0. \tag{18}$$

The function w_j and f are the real-valued functions. We assume that all functions are continuously differentiable with respect to their own independent arguments. Notice that the control function $\vec{v}(t)$ influences the functional E directly by its own values and indirectly by its impact on the state function $\vec{p}(t)$ in Eq.(16). Moreover, in our problem, the initial point of the state $\vec{p}(t)$ is fixed in both space and time, and can be computed in advance. That is:

$$t_0 \text{ and } \vec{p}(t_0) \text{ are fixed to the known values.}$$

A.2   Necessary Conditions for Optimality

To find the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$, we first eliminate Eq.(16) by appending it into the functional E with the Lagrange multiplier vector function $\vec{\lambda}(t)$, which results in:

$$E = \int_{t_0}^{t_1} g(t, \vec{p}(t), \vec{v}(t)) + \vec{\lambda}(t) \cdot (\vec{h}(t, \vec{p}(t), \vec{v}(t)) - \dot{\vec{p}}(t))dt, \tag{19}$$

where the Lagrange multiplier $\vec{\lambda}(t) \in \mathcal{R}^m$ can be arbitrary vector function. Then, we expand the product term and apply the integration by parts, so Eq.(19) turns out to be:

$$E = \int_{t_0}^{t_1} \left(g + \vec{\lambda} \cdot \vec{h} + \dot{\vec{\lambda}} \cdot \vec{p} \right) dt + \vec{\lambda}(t_0) \cdot \vec{p}(t_0) - \vec{\lambda}(t_1) \cdot \vec{p}(t_1), \tag{20}$$

Note that we discard the arguments of the functions in the integral just because of the limited space and for the clear explanation. Please remember that such functions still depend on their own independent variables that are previously displayed. We assume that the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$ over time period $[t_0, t_1]$ produces the minimum functional E^* within some neighborhoods E, so from Eq.(20) we obtain:

$$E^* = \int_{t_0}^{t_1} \left(g^* + \vec{\lambda} \cdot \vec{h}^* + \dot{\vec{\lambda}} \cdot \vec{p}^* \right) dt + \vec{\lambda}(t_0) \cdot \vec{p}^*(t_0) - \vec{\lambda}(t_1) \cdot \vec{p}^*(t_1), \tag{21}$$

where the asterisk in the superscript means the function is evaluated at the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$.

For the neighborhoods of $(t, \vec{p}^*(t), \vec{v}^*(t))$, if trajectories $(t, \vec{p}(t), \vec{v}(t))$ for $t \in [t_0, t_1 + \delta t_1]$ are its neighborhoods, they must produce higher or equal functional E to the minimum functional E^*. Note that the state at the endpoint in our case is free in both space and time, so the time of the endpoint of neighboring trajectories can be shifted and this is the reason why the upperbound must be $t_1 + \delta t_1$, where δt_1 is a small infinitesimal quantity. From Eq.(19), the functional E produced by the neighborhoods $(t, \vec{p}(t), \vec{v}(t))$ can be computed by:

$$E = \int_{t_0}^{t_1+\delta t_1} \left(g + \vec{\lambda} \cdot \vec{h} - \vec{\lambda} \cdot \dot{\vec{p}} \right) dt. \tag{22}$$

Rewrite Eq.(22) by splitting the integral at time t_1 into two separate terms, so

$$E = \int_{t_0}^{t_1} \left(g + \vec{\lambda} \cdot \vec{h} - \vec{\lambda} \cdot \dot{\vec{p}} \right) dt + \int_{t_1}^{t_1+\delta t_1} \left(g + \vec{\lambda} \cdot \vec{h} - \vec{\lambda} \cdot \dot{\vec{p}} \right) dt. \tag{23}$$

The first integral term on the right side of Eq.(23) is the same form as one in Eq.(19), so it can be replaced with the right side of Eq.(20). Therefore,

$$\begin{aligned} E = &\int_{t_0}^{t_1} \left(g + \vec{\lambda} \cdot \vec{h} + \dot{\vec{\lambda}} \cdot \vec{p} \right) dt + \vec{\lambda}(t_0) \cdot \vec{p}(t_0) - \vec{\lambda}(t_1) \cdot \vec{p}(t_1) \\ &+ \int_{t_1}^{t_1+\delta t_1} \left(g + \vec{\lambda} \cdot \vec{h} - \vec{\lambda} \cdot \dot{\vec{p}} \right) dt. \end{aligned} \tag{24}$$

Consider the last integral term in Eq.(24), it can be approximated by:

$$\begin{aligned} \int_{t_1}^{t_1+\delta t_1} \left(g + \vec{\lambda} \cdot \vec{h} - \vec{\lambda} \cdot \dot{\vec{p}} \right) dt &\approx \left. \left(g + \vec{\lambda} \cdot \vec{h} - \vec{\lambda} \cdot \dot{\vec{p}} \right) \right|_{t_1} \delta t_1 \\ &= g|_{t=t_1} \delta t_1 \\ &\approx \left. \left(g^* + \frac{\partial g^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial g^*}{\partial \vec{v}} \cdot \Delta \vec{v} + R_2^* \right) \right|_{t=t_1} \delta t_1 \\ &\approx g^*|_{t=t_1} \delta t_1 \end{aligned} \tag{25}$$

where $\Delta \vec{p}(t) = \vec{p}(t) - \vec{p}^*(t)$, $\Delta \vec{v}(t) = \vec{v}(t) - \vec{v}^*(t)$, and R_2^* is the remainder. The second line in Eq.(25) is from eliminating the last two terms in the first line. This was due to the equality constraint specified in Eq.(16). The third line results from applying the Taylor series expansion to the function g about the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$. The term $\Delta \vec{p} \delta t_1$, $\Delta \vec{v} \delta t_1$, and $R_2^* \delta t_1$ are very small, so they are eliminated, and the approximation ends in the last line. Therefore, Eq.(24) becomes:

$$\begin{aligned} E = &\int_{t_0}^{t_1} \left(g + \vec{\lambda} \cdot \vec{h} + \dot{\vec{\lambda}} \cdot \vec{p} \right) dt + \vec{\lambda}(t_0) \cdot \vec{p}(t_0) - \vec{\lambda}(t_1) \cdot \vec{p}(t_1) \\ &+ g^*|_{t=t_1} \delta t_1 \end{aligned} \tag{26}$$

Since the functional E^* is a local minima within some neighborhood E, which is expressed by Eq.(26), so we get

$$E - E^* \geq 0. \tag{27}$$

Consider $E - E^*$ from Eq.(21) and Eq.(26),

$$\begin{aligned} E - E^* \triangleq &\int_{t_0}^{t_1} \left((g - g^*) + \vec{\lambda} \cdot (\vec{h} - \vec{h}^*) + \dot{\vec{\lambda}} \cdot (\vec{p} - \vec{p}^*) \right) dt \\ &+ \vec{\lambda}(t_0) \cdot \left(\vec{p}(t_0) - \vec{p}^*(t_0) \right) - \vec{\lambda}(t_1) \cdot \left(\vec{p}(t_1) - \vec{p}^*(t_1) \right) \\ &+ g^*|_{t=t_1} \delta t_1. \end{aligned}$$

Then, we use the Taylor series expansion on the terms $g - g^*$ and $\vec{h} - \vec{h}^*$, and change $\vec{p} - \vec{p}^*$ to $\Delta \vec{p}$.

$$
\begin{aligned}
E - E^* \triangleq \int_{t_0}^{t_1} &\left\{ \left(\frac{\partial g^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial g^*}{\partial \vec{v}} \cdot \Delta \vec{v} \right) \right. \\
&+ \left. \sum_{i=1}^{m} \lambda_i \left(\frac{\partial h_i^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial h_i^*}{\partial \vec{v}} \cdot \Delta \vec{v} \right) + \dot{\vec{\lambda}} \cdot \Delta \vec{p} \right\} dt \\
&+ \vec{\lambda}(t_0) \cdot \Delta \vec{p}(t_0) - \vec{\lambda}(t_1) \cdot \Delta \vec{p}(t_1) \\
&+ g^*|_{t=t_1} \delta t_1,
\end{aligned}
\tag{28}
$$

where λ_i is the ith component of the Lagrange multiplier vector function $\vec{\lambda}$, and h_i is the ith component of the vector function $\vec{h}$. As the state at initial point, the state at time t_0, in our case is fixed in both space and time, so $\Delta \vec{p}(t_0) = 0$, and Eq.(28) becomes:

$$
\begin{aligned}
E - E^* \triangleq \int_{t_0}^{t_1} &\left\{ \left(\frac{\partial g^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial g^*}{\partial \vec{v}} \cdot \Delta \vec{v} \right) \right. \\
&+ \left. \sum_{i=1}^{m} \lambda_i \left(\frac{\partial h_i^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial h_i^*}{\partial \vec{v}} \cdot \Delta \vec{v} \right) + \dot{\vec{\lambda}} \cdot \Delta \vec{p} \right\} dt \\
&- \vec{\lambda}(t_1) \cdot \Delta \vec{p}(t_1) + g^*|_{t=t_1} \delta t_1.
\end{aligned}
\tag{29}
$$

Let $\delta \vec{p}_1$ be the difference between the endpoint of the optimal state $\vec{p}^*$, which ends at the time t_1, and the endpoint of the neighborhood $\vec{p}$, which ends at the time $t_1 + \delta t_1$. Specifically, $\delta \vec{p}_1 = \vec{p}(t_1 + \delta t_1) - \vec{p}^*(t_1)$. So $\Delta \vec{p}(t_1)$ can be approximated by:

$$
\Delta \vec{p}(t_1) \approx \delta \vec{p}_1 - \dot{\vec{p}}^*(t_1) \delta t_1.
\tag{30}
$$

Replacing it in Eq.(29) results in:

$$
\begin{aligned}
E - E^* \triangleq \int_{t_0}^{t_1} &\left\{ \left(\frac{\partial g^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial g^*}{\partial \vec{v}} \cdot \Delta \vec{v} \right) \right. \\
&+ \left. \sum_{i=1}^{m} \lambda_i \left(\frac{\partial h_i^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial h_i^*}{\partial \vec{v}} \cdot \Delta \vec{v} \right) + \dot{\vec{\lambda}} \cdot \Delta \vec{p} \right\} dt \\
&+ \left(g^*|_{t=t_1} + \vec{\lambda}(t_1) \cdot \dot{\vec{p}}^*(t_1) \right) \delta t_1 - \vec{\lambda}(t_1) \cdot \delta \vec{p}_1.
\end{aligned}
\tag{31}
$$

Rearrange terms in the integrand in Eq.(31), so we get:

$$
\begin{aligned}
E - E^* \triangleq \int_{t_0}^{t_1} &\left\{ \left(\frac{\partial g^*}{\partial \vec{p}} + \sum_{i=1}^{m} \lambda_i \frac{\partial h_i^*}{\partial \vec{p}} + \dot{\vec{\lambda}} \right) \cdot \Delta \vec{p} \right. \\
&+ \left. \left(\frac{\partial g^*}{\partial \vec{v}} + \sum_{i=1}^{m} \lambda_i \frac{\partial h_i^*}{\partial \vec{v}} \right) \cdot \Delta \vec{v} \right\} dt \\
&+ \left(g^*|_{t=t_1} + \vec{\lambda}(t_1) \cdot \dot{\vec{p}}^*(t_1) \right) \delta t_1 - \vec{\lambda}(t_1) \cdot \delta \vec{p}_1.
\end{aligned}
\tag{32}
$$

Let $\mathcal{H}(t, \vec{p}(t), \vec{v}(t), \vec{\lambda}(t)) = g + \vec{\lambda} \cdot \vec{h}$, which is called the Hamiltonian. So Eq.(32) turns out to be:

$$
\begin{aligned}
E - E^* \triangleq \int_{t_0}^{t_1} &\left\{ \left(\frac{\partial \mathcal{H}^*}{\partial \vec{p}} + \dot{\vec{\lambda}} \right) \cdot \Delta \vec{p} + \left(\frac{\partial \mathcal{H}^*}{\partial \vec{v}} \right) \cdot \Delta \vec{v} \right\} dt \\
&+ \left(\mathcal{H}^*|_{t=t_1} \right) \delta t_1 - \vec{\lambda}(t_1) \cdot \delta \vec{p}_1.
\end{aligned}
\tag{33}
$$

From Eq.(27) and Eq.(33), the optimal trajectory $(t, \vec{p}^*, \vec{v}^*)$ which produces the local minimum functional E^* must satisfy the following equation:

$$\int_{t_0}^{t_1} \left\{ \left(\frac{\partial \mathcal{H}^*}{\partial \vec{p}} + \dot{\vec{\lambda}} \right) \cdot \Delta \vec{p} + \left(\frac{\partial \mathcal{H}^*}{\partial \vec{v}} \right) \cdot \Delta \vec{v} \right\} dt$$
$$+ \left(\mathcal{H}^*|_{t=t_1} \right) \delta t_1 - \vec{\lambda}(t_1) \cdot \delta \vec{p}_1 \geq 0. \tag{34}$$

The optimal trajectory $(t, \vec{p}^*, \vec{v}^*)$ yields the local minimum E^* over *all* admissible neighborhoods $(t, \vec{p}, \vec{v})$, and *some* neighborhoods $(t, \vec{p}, \vec{v})$ could have the same endpoint in both space and time as the endpoint of the optimal trajectory $(t, \vec{p}^*, \vec{v}^*)$, that is $\delta t_1 = 0$ and $\delta \vec{p}_1 = 0$, which turns Eq.(34) into

$$\int_{t_0}^{t_1} \left\{ \left(\frac{\partial \mathcal{H}^*}{\partial \vec{p}} + \dot{\vec{\lambda}} \right) \cdot \Delta \vec{p} + \left(\frac{\partial \mathcal{H}^*}{\partial \vec{v}} \right) \cdot \Delta \vec{v} \right\} dt \geq 0. \tag{35}$$

So the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$ must satisfy

$$\left(\frac{\partial \mathcal{H}^*}{\partial \vec{p}} + \dot{\vec{\lambda}} \right) \cdot \Delta \vec{p} + \left(\frac{\partial \mathcal{H}^*}{\partial \vec{v}} \right) \cdot \Delta \vec{v} \geq 0, \tag{36}$$

for all $t \in [t_0, t_1]$. Back to the N inequality constraints in Eq.(17). We call the constraint w_j is *inactive* at time t, if $w_j(\vec{p}(t), t) < 0$; otherwise, *active* at time t. If the optimal state $\vec{p}^*(t)$ makes the jth constraint w_j inactive at a certain time t, so $w_j(\vec{p}^*(t), t) < 0$. However, for any neighborhood $\vec{p}(t)$, it must satisfy Eq.(17), so $w_j(\vec{p}(t), t) \leq 0$. This conduces to:

$$w_j(\vec{p}(t), t) \ \{>, =, \text{or} \ <\} \ w_j(\vec{p}^*(t), t)$$

which places no restriction on $\Delta \vec{p}(t)$. On the other hand, if the optimal state $\vec{p}^*(t)$ makes the jth constraint w_j active at a certain time t, so $w_j(\vec{p}^*(t), t) = 0$, and, as before, the neighborhood $\vec{p}(t)$ must satisfy $w_j(\vec{p}(t), t) \leq 0$. This conduces to:

$$w_j(\vec{p}(t), t) \leq w_j(\vec{p}^*(t), t)$$

and results in the restriction on $\Delta \vec{p}(t)$ as shown below:

$$\frac{\partial w_j^*}{\partial \vec{p}} \cdot \Delta \vec{p} \leq 0. \tag{37}$$

In case the optimal state $\vec{p}^*(t)$ makes *all* constraints w_j *inactive* at a certain time t, so $\Delta \vec{p}(t)$ will not be constrained by any w_j. This leads to any possibilities of values of $\Delta \vec{p}$ at time t. Likewise, the control $\vec{v}(t)$ is also arbitrary, leading to arbitrary $\Delta \vec{v}$ as well. Therefore, to satisfy Eq.(36) when $\Delta \vec{p}(t)$ and $\Delta \vec{v}(t)$ can be arbitrary, the terms in the parentheses must be zero:

$$\frac{\partial \mathcal{H}^*}{\partial \vec{p}} + \dot{\vec{\lambda}} = 0 \quad \text{and} \quad \frac{\partial \mathcal{H}^*}{\partial \vec{v}} = 0. \tag{38}$$

Eq.(38) expresses the characteristics of the optimal state $\vec{p}^*(t)$ and optimal control $\vec{v}^*(t)$ in a certain time period when all inequality constraints are inactive. In case the optimal state $\vec{p}^*(t)$ makes *some/all* constraints w_j *active*. All active w_j must place the restriction on $\Delta \vec{p}(t)$, as shown in Eq.(37). So

$$\frac{\partial w_j^*}{\partial \vec{p}} \cdot \Delta \vec{p} \leq 0, \quad j \in \mathcal{A}_t, \tag{39}$$

where $\mathcal{A}_t$ is a set of indices of the active constraints at time t. To satisfy Eq.(36) when $\Delta \vec{p}(t)$ is constrained by Eq.(39), the Farkas's lemma, described in the appendix B, is then employed. This results in:

$$\frac{\partial \mathcal{H}^*}{\partial \vec{p}} + \dot{\vec{\lambda}} + \sum_{j \in \mathcal{A}_t} \alpha_j(t) \frac{\partial w_j^*}{\partial \vec{p}} = 0 \quad \text{and} \quad \frac{\partial \mathcal{H}^*}{\partial \vec{v}} = 0, \tag{40}$$

where $\alpha_j(t) \geq 0$ for all $j \in \mathcal{A}_t$. Eq.(40) expresses the characteristics of the optimal state $\vec{p}^*(t)$ and optimal control $\vec{v}^*(t)$ in a certain time period when some/all inequality constraints are active.

Notice from Eq.(38) and Eq.(40) that if an inequality constraint becomes active by the optimal state $\vec{p}^*$, the term $\alpha_j \partial w_j^* / \partial \vec{p}$ will be added. So we generalize this by raising the additional equation shown below:

$$\alpha_j(t) w_j(\vec{p}^*(t), t) = 0, \quad j = 1, ..., N.$$

In summary, the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$ must satisfy:

$$\begin{aligned} \dot{\vec{\lambda}} + \frac{\partial \mathcal{L}^*}{\partial \vec{p}} &= 0 \\ \frac{\partial \mathcal{L}^*}{\partial \vec{v}} &= 0 \\ \alpha_j(t) w_j(\vec{p}^*(t), t) &= 0, \quad j = 1, ..., N \\ \alpha_j(t) &\geq 0, \quad j = 1, ..., N \\ \vec{\lambda}(t) &\in \mathcal{R}^m, \end{aligned} \tag{41}$$

for all $t \in [t_0, t_1]$, where $\mathcal{L}$ is called the Lagrangian and equals to:

$$\begin{aligned} \mathcal{L}(t, \vec{p}(t), \vec{v}(t), &\vec{\lambda}(t), \alpha_1(t), ..., \alpha_N(t)) \\ &= \mathcal{H} + \sum_{j=1}^{N} \alpha_j(t) w_j(\vec{p}(t), t). \end{aligned}$$

So far we completely investigate the characteristics of the optimal trajectory $(t, \vec{p}^*, \vec{v}^*)$ against the neighborhoods $(t, \vec{p}, \vec{v})$ that have the same endpoint in both space and time as one of the optimal trajectory. Now it is time to investigate the characteristics of the optimal trajectory $(t, \vec{p}^*, \vec{v}^*)$ against the neighborhoods $(t, \vec{p}, \vec{v})$ that have different endpoint to one of the optimal trajectory. That is $\delta t_1 \neq 0$ and $\delta \vec{p}_1 \neq 0$. Do not forget that Eq.(34) must hold for the optimal trajectory $(t, \vec{p}^*, \vec{v}^*)$, and because Eq.(35) holds in the previous investigation, so Eq.(34) turns out to be:

$$\left(\mathcal{H}^* |_{t=t_1} \right) \delta t_1 - \vec{\lambda}(t_1) \cdot \delta \vec{p}_1 \geq 0. \tag{42}$$

However, the endpoints are constrained by Eq.(18), which yields:

$$f(\vec{p}^*(t_1)) = 0 \quad \text{and} \quad f(\vec{p}(t_1 + \delta t_1)) = 0.$$

Recall that the optimal state $\vec{p}^*$ and the neighborhood $\vec{p}$ are assumed to end at the time t_1 and $t_1 + \delta t_1$, respectively. Because of $\vec{p}(t_1 + \delta t_1) = \vec{p}^*(t_1) + \delta \vec{p}_1$, so we get:

$$f(\vec{p}^*(t_1)) = 0 \quad \text{and} \quad f(\vec{p}^*(t_1) + \delta \vec{p}_1) = 0.$$

Using the Taylor series expansion to above equations yields the following constraint towards $\delta \vec{p}_1$:

$$\left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta \vec{p}_1 = 0. \tag{43}$$

Eq.(43) can be added into Eq.(42) without loss of generality by multiplying with the real-valued constant variable γ, and then adding into Eq.(42). Therefore, the optimal trajectory $(t, \vec{p}^{*}, \vec{v}^{*})$ must satisfy:

$$\left(\mathcal{H}^{*}|_{t=t_1}\right)\delta t_1 + \left(\gamma\frac{\partial f^{*}}{\partial \vec{p}}\bigg|_{t=t_1} - \vec{\lambda}(t_1)\right)\cdot\delta\vec{p}_1 \geq 0, \tag{44}$$

where $\gamma \in \mathcal{R}$ and can be arbitrary real value. However, the endpoints are not constrained by only Eq.(18) but Eq.(17) as well. If the endpoint of the optimal state $\vec{p}^{*}$ makes w_j inactive, so $w_j(\vec{p}^{*}(t_1), t_1) < 0$, and again, $w_j(\vec{p}(t_1 + \delta t_1), t_1 + \delta t_1) \leq 0$ must hold for the endpoint of the neighborhood $\vec{p}$, which results in:

$$w_j(\vec{p}(t_1 + \delta t_1), t_1 + \delta t_1) \ \{>, =, \text{ or } <\} \ w_j(\vec{p}^{*}(t_1), t_1).$$

The above equation places no restriction on $\delta\vec{p}_1$. On the one hand, if it makes w_j active, so $w_j(\vec{p}^{*}(t_1), t_1) = 0$, and

$$w_j(\vec{p}(t_1 + \delta t_1), t_1 + \delta t_1) \leq w_j(\vec{p}^{*}(t_1), t_1).$$

The above equation places the restriction on both $\delta\vec{p}_1$ and δt_1, as shown below:

$$\left(\frac{\partial w_j^{*}}{\partial t}\bigg|_{t=t_1}\right)\delta t_1 + \left(\frac{\partial w_j^{*}}{\partial \vec{p}}\bigg|_{t=t_1}\right)\cdot\delta\vec{p}_1 \leq 0. \tag{45}$$

In case the endpoint of the optimal state $\vec{p}^{*}(t)$ makes *all* constraints w_j inactive, $\delta\vec{p}_1$ and also δt_1 can be arbitrary values, and in order to satisfy Eq.(44), the terms in the parentheses must be zero:

$$\mathcal{H}^{*}|_{t=t_1} = 0 \quad \text{and} \quad \left(\gamma\frac{\partial f^{*}}{\partial \vec{p}}\bigg|_{t=t_1}\right) - \vec{\lambda}(t_1) = 0. \tag{46}$$

Eq.(46) expresses the characteristics of the optimal state $\vec{p}^{*}(t)$ and optimal control $\vec{v}^{*}(t)$ at the time t_1 when all inequality constraints are inactive.

In case the endpoint of the optimal state $\vec{p}^{*}(t)$ makes *some/all* constraints w_j active. All active w_j must place the restriction on both $\delta\vec{p}_1$ and δt_1, as shown in Eq.(45). So

$$\left(\frac{\partial w_j^{*}}{\partial t}\bigg|_{t=t_1}\right)\delta t_1 + \left(\frac{\partial w_j^{*}}{\partial \vec{p}}\bigg|_{t=t_1}\right)\cdot\delta\vec{p}_1 \leq 0, \quad j \in \mathcal{A}_{t_1}. \tag{47}$$

To satisfy Eq.(44) when $\delta\vec{p}_1$ and δt_1 are constrained by Eq.(47), the Farkas's lemma is employed again, which results in:

$$\begin{aligned}
\mathcal{H}^{*}|_{t=t_1} + \sum_{j\in\mathcal{A}_{t_1}}\beta_j\frac{\partial w_j^{*}}{\partial t}\bigg|_{t=t_1} &= 0, \\
\left(\gamma\frac{\partial f^{*}}{\partial \vec{p}}\bigg|_{t=t_1}\right) - \vec{\lambda}(t_1) + \sum_{j\in\mathcal{A}_{t_1}}\beta_j\frac{\partial w_j^{*}}{\partial \vec{p}}\bigg|_{t=t_1} &= 0,
\end{aligned} \tag{48}$$

where $\beta_j \geq 0$ for all $j \in \mathcal{A}_{t_1}$. Eq.(48) expresses the characteristics of the optimal state $\vec{p}^{*}(t)$ and optimal control $\vec{v}^{*}(t)$ at the time t_1 when some/all inequality constraints are active.

Notice from Eq.(46) and Eq.(48) that if an inequality constraint becomes active at time t_1 by the optimal state $\vec{p}^{*}$, the term $\beta_j\partial w_j^{*}/\partial t$ and $\beta_j\partial w_j^{*}/\partial\vec{p}$ will be added. We generalize this by raising the additional equation shown below:

$$\beta_j w_j(\vec{p}^{*}(t_1), t_1) = 0, \quad j = 1, ..., N.$$

In summary, the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$ at time t_1 must satisfy:

$$\mathcal{H}^*|_{t=t_1} + \sum_{j=1}^{N} \beta_j \frac{\partial w_j^*}{\partial t}\bigg|_{t=t_1} = 0,$$

$$\left(\gamma \frac{\partial f^*}{\partial \vec{p}}\bigg|_{t=t_1}\right) - \vec{\lambda}(t_1) + \sum_{j=1}^{N} \beta_j \frac{\partial w_j^*}{\partial \vec{p}}\bigg|_{t=t_1} = 0,$$

$$\beta_j w_j(\vec{p}^*(t_1), t_1) = 0, \quad j = 1, ..., N$$

$$\beta_j \geq 0, \quad j = 1, ..., N$$

$$\gamma \in \mathcal{R}.$$

(49)

For the initial point of the optimal trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$, the initial point of the state $\vec{p}(t)$ in our problem is fixed in both space and time, and can be known in advance. So

$$\vec{p}^*(t_0) = \vec{p}_0,$$

(50)

where $\vec{p}_0$ is already known.

Conclusion: If a trajectory $(t, \vec{p}(t), \vec{v}(t))$ is the optimal trajectory to the problem, it must satisfy the necessary conditions defined by Eq.(41) along with the boundary conditions defined by Eq.(49) and Eq.(50).

A.3 Sufficient Conditions for Optimality

So far we know that if a trajectory $(t, \vec{p}(t), \vec{v}(t))$ is the optimal trajectory to the problem, which yields the minimum functional E^*, it must satisfy Eq.(41). But we cannot say that any trajectory $(t, \vec{p}(t), \vec{v}(t))$ that satisfies Eq.(41) is the *(local) minimum* trajectory, because in the previous derivation the only first-order Taylor series was used, causing $E - E^* = 0$ when Eq.(41) holds. So we cannot conclude that it is the minimum trajectory. This is analogous to the problem of finding a point x that minimizes the function $f(x)$, which we cannot know that the point x that satisfy $f'(x) = 0$ is the minimum point unless $f''(x) \geq 0$ is satisfied. Therefore, the sufficient conditions for optimality are essential, and we will examine in this section.

What we need is the conditions that assert $E - E^* \geq 0$ about the (local) minimum trajectory $(t, \vec{p}^*(t), \vec{v}^*(t))$. Please remember that the point $(t, \vec{p}^*(t), \vec{v}^*(t))$ satisfies the necessary conditions defined by Eq.(41) and the boundary conditions defined by Eq.(49) and Eq.(50). From the definition E in Eq.(15),

$$\Delta \triangleq E - E^* = \int_{t_0}^{t_1} (g - g^*)dt + \int_{t_1}^{t_1 + \delta t_1} g dt.$$

(51)

According to the Hamiltonian $\mathcal{H} = g + \vec{\lambda} \cdot \vec{h}$, Eq.(51) becomes:

$$\Delta = \int_{t_0}^{t_1} (\mathcal{H} - \mathcal{H}^*)dt + \int_{t_0}^{t_1} \vec{\lambda} \cdot (\vec{h}^* - \vec{h})dt + \int_{t_1}^{t_1 + \delta t_1} g dt,$$

(52)

and we can deduce that $\Delta \geq 0$ from these following steps:

$$\Delta = \int_{t_0}^{t_1} (\mathcal{H} - \mathcal{H}^*)dt + \int_{t_0}^{t_1} \vec{\lambda} \cdot (\vec{h}^* - \vec{h})dt + \int_{t_1}^{t_1 + \delta t_1} g dt$$

$$\Delta \overset{(1)}{\geq} \int_{t_0}^{t_1} \left(\frac{\partial \mathcal{H}^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial \mathcal{H}^*}{\partial \vec{v}} \cdot \Delta \vec{v}\right)dt + \int_{t_0}^{t_1} \vec{\lambda} \cdot (\vec{h}^* - \vec{h})dt + \int_{t_1}^{t_1 + \delta t_1} g dt$$

$$\Delta \overset{(2)}{=} \int_{t_0}^{t_1} \left(-\dot{\vec{\lambda}} \cdot \Delta \vec{p} - \sum_{j=1}^{N} \alpha_j \frac{\partial w_j^*}{\partial \vec{p}} \cdot \Delta \vec{p} + \frac{\partial \mathcal{L}^*}{\partial \vec{v}} \cdot \Delta v\right)dt + \int_{t_0}^{t_1} \vec{\lambda} \cdot (\vec{h}^* - \vec{h})dt + \int_{t_1}^{t_1 + \delta t_1} g dt$$

$$\Delta \stackrel{(3)}{=} \int_{t_0}^{t_1} \left(-\dot{\vec{\lambda}} \cdot \Delta\vec{p} - \vec{\lambda} \cdot \Delta\dot{\vec{p}} \right) dt + \int_{t_0}^{t_1} \left(-\sum_{j=1}^{N} \alpha_j \frac{\partial w_j^*}{\partial \vec{p}} \cdot \Delta\vec{p} \right) dt + \int_{t_1}^{t_1+\delta t_1} g \, dt$$

$$\Delta \stackrel{(4)}{=} \int_{t_0}^{t_1} \left(-\frac{d}{dt}(\vec{\lambda} \cdot \Delta\vec{p}) \right) dt + \int_{t_0}^{t_1} \left(-\sum_{j=1}^{N} \alpha_j \frac{\partial w_j^*}{\partial \vec{p}} \cdot \Delta\vec{p} \right) dt + \int_{t_1}^{t_1+\delta t_1} g \, dt$$

$$\Delta \stackrel{(5)}{\geq} \int_{t_0}^{t_1} \left(-\frac{d}{dt}(\vec{\lambda} \cdot \Delta\vec{p}) \right) dt + \int_{t_1}^{t_1+\delta t_1} g \, dt$$

$$\Delta \stackrel{(6)}{\approx} -\vec{\lambda}(t_1) \cdot \Delta\vec{p}(t_1) + \vec{\lambda}(t_0) \cdot \Delta\vec{p}(t_0) + g^*|_{t=t_1} \delta t_1$$

$$\Delta \stackrel{(7)}{=} -\vec{\lambda}(t_1) \cdot \Delta\vec{p}(t_1) + g^*|_{t=t_1} \delta t_1$$

$$\Delta \stackrel{(8)}{=} -\gamma \left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \Delta\vec{p}(t_1) - \sum_{j=1}^{N} \beta_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \Delta\vec{p}(t_1) + g^*|_{t=t_1} \delta t_1$$

$$\Delta \stackrel{(9)}{=} -\gamma \left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \Delta\vec{p}(t_1) - \sum_{j=1}^{N} \beta_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \Delta\vec{p}(t_1)$$

$$+ \gamma \left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \Delta\vec{p}(t_1) + \sum_{j=1}^{N} \beta_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \Delta\vec{p}(t_1)$$

$$- \gamma \left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta\vec{p}_1 - \sum_{j=1}^{N} \beta_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta\vec{p}_1$$

$$+ \mathcal{H}^*|_{t=t_1} \delta t_1$$

$$\Delta \stackrel{(10)}{=} -\gamma \left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta\vec{p}_1 - \sum_{j=1}^{N} \beta_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta\vec{p}_1 + \mathcal{H}^*|_{t=t_1} \delta t_1$$

$$\Delta \stackrel{(11)}{=} -\gamma \left(\frac{\partial f^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta\vec{p}_1 - \sum_{j=1}^{N} \beta_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \bigg|_{t=t_1} \right) \cdot \delta\vec{p}_1 - \sum_{j=1}^{N} \beta_j \frac{\partial w_j^*}{\partial t} \bigg|_{t=t_1} \delta t_1$$

$$\Delta \stackrel{(12)}{\geq} 0.$$

Explanation:
(1): Assume that $\mathcal{H}$ is convex in $(\vec{p}, \vec{v})$. So $\mathcal{H} - \mathcal{H}^* \geq \partial\mathcal{H}^*/\partial\vec{p} \cdot \Delta\vec{p} + \partial\mathcal{H}^*/\partial\vec{v} \cdot \Delta\vec{v}$.
(2): Use the fact that $\mathcal{H} = \mathcal{L} - \sum \alpha_j w_j$ and $\partial\mathcal{L}^*/\partial\vec{p} = -\dot{\vec{\lambda}}$.
(3): $\partial\mathcal{L}^*/\partial\vec{v} = 0$ and $\Delta\dot{\vec{p}} = \vec{h} - \vec{h}^*$.
(4): $d(\vec{\lambda} \cdot \Delta\vec{p}) = \dot{\vec{\lambda}} \cdot \Delta\vec{p} + \vec{\lambda} \cdot \Delta\dot{\vec{p}}$.
(5): Assume that $\alpha_j(\partial w_j^*/\partial\vec{p}) \cdot \Delta\vec{p} \leq 0$ for all $j = 1, ..., N$.
(6): The first two terms come from the rule of integration, and the last one from Eq.(25).
(7): Due to the fact that the initial point in our case is fixed, so $\Delta\vec{p}(t_0) = 0$.
(8): Replace $\vec{\lambda}(t_1)$ with the boundary condition in Eq.(49).
(9): Replace $g^*|_{t=t_1}$ with $\mathcal{H}^*|_{t=t_1} - \vec{\lambda}(t_1) \cdot \vec{h}^*|_{t=t_1}$, and follow from the fact in Eq.(16), Eq.(30), and Eq.(49).
(10): First four terms are cancelled out.
(11): Replace $\mathcal{H}^*|_{t=t_1}$ with the boundary condition in Eq.(49).
(12): Assume that $\gamma(\partial f^*/\partial\vec{p})|_{t=t_1} \cdot \delta\vec{p}_1 \leq 0$, and $\beta_j(\partial w_j^*/\partial\vec{p})|_{t=t_1} \cdot \delta\vec{p}_1 + \beta_j(\partial w_j^*/\partial t)|_{t=t_1}\delta t_1 \leq 0$.

Conclusion: A trajectory $(t, \vec{p}, \vec{v})$ that yields the (local) minimum functional E^*; in other word $E - E^* \geq 0$, must satisfy not only the necessary conditions defined by Eq.(41) and the boundary conditions in Eq.(49), but these following conditions as well:

$$\mathcal{H} \text{ is convex in } (\vec{p}, \vec{v}) \tag{53}$$

$$\alpha_j \left(\frac{\partial w_j^*}{\partial \vec{p}} \right) \cdot \Delta \vec{p} \leq 0, \quad j = 1, ..., N. \tag{54}$$

$$\gamma \left(\frac{\partial f^*}{\partial \vec{p}} \right) \Bigg|_{t=t_1} \cdot \delta \vec{p}_1 \leq 0. \tag{55}$$

$$\beta_j \left\{ \left(\frac{\partial w_j^*}{\partial \vec{p}} \Bigg|_{t=t_1} \right) \cdot \delta \vec{p}_1 + \left(\frac{\partial w_j^*}{\partial t} \Bigg|_{t=t_1} \right) \delta t_1 \right\} \leq 0, \quad j = 1, ..., N. \tag{56}$$
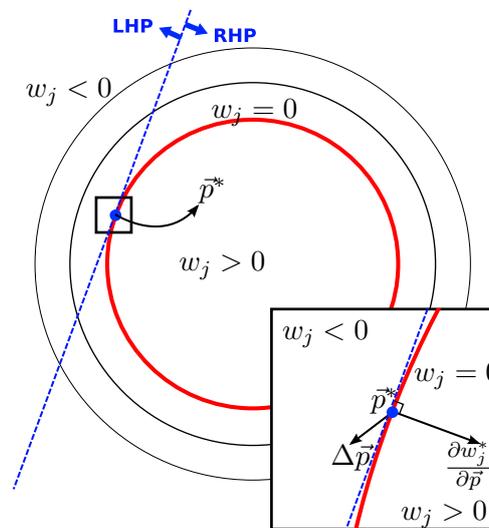

Fig. 13. Graphical meaning of the inequality constraint $w_j(\vec{p}(t), t) \leq 0$ used in our optimization problem. Suppose at a certain time t the optimal point $\vec{p}^*(t)$ on the optimal trajectory makes the constraint w_j active as shown in the figure, we can conclude that $\partial w_j^*/\partial \vec{p} \cdot \Delta \vec{p} \leq 0$ holds for every neighborhood $\vec{p}(t)$, when the distance between those two points ($\Delta \vec{p}$) is infinitesimal.

A.4 Energy-Minimal Walking Characteristics

Our energy-minimal walking characteristics are acquired by replacing the functions in the previous sections with these following functions: (1) $g(t, \vec{p}(t), \vec{v}(t)) = m(e_s + e_w \|\vec{v}(t)\|^2)$, (2) $t_0 = t_c$ and $t_1 = t_e$, and (3) $\vec{h}(t, \vec{p}(t), \vec{v}(t)) = \vec{v}(t)$, then eliminating the variable $\vec{\lambda}$ in Eq.(41), which results in the optimal acceleration defined by Eq.(7). In addition, the characteristics of the optimal velocity at time t_e, defined by Eq.(9) and Eq.(10), were obtained by using Eq.(49).

Considering the Hamiltonian $\mathcal{H} = g + \vec{\lambda} \cdot \vec{h}$, we get:

$$\mathcal{H} = m(e_s + e_w \|\vec{v}(t)\|^2) + \vec{\lambda} \cdot \vec{v}(t),$$

where the Hessian matrix of this $\mathcal{H}$, corresponding to $\vec{p}$ and $\vec{v}$ where $\vec{p}, \vec{v} \in \mathcal{R}^2$, is:

$$\text{Hessian}(\mathcal{H}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2me_w & 0 \\ 0 & 0 & 0 & 2me_w \end{bmatrix}$$
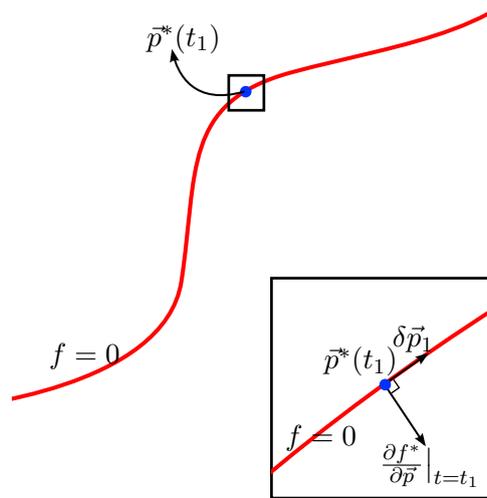
Fig. 14. Graphical meaning of the endpoint constraint $f(\vec{p}(t_1)) = 0$. Since the endpoints of the optimal trajectory and the neighborhood must lie on this curve, so we can conclude that $(\partial f^*/\partial \vec{p})|_{t=t_1} \cdot \delta \vec{p}_1 = 0$ holds for every neighborhood of the optimal point $\vec{p}^*(t_1)$, when $\delta \vec{p}_1$ is infinitesimal.

which is positive-definite matrix. This means that the Hamiltonian $\mathcal{H}$ is convex in $(\vec{p}, \vec{v})$. So the sufficient condition in Eq.(53) is satisfied.

For the inequality constraints $w_j(\vec{p}(t), t) \leq 0$, we use the circle equation to represent a virtual pedestrian's boundary. Mathematically,

$$w_j(\vec{p}, t) \triangleq r^2 - \|\vec{p} - \vec{p}_j(t)\|^2,$$

and in addition we obtain the fact that

$$w_j(\vec{p}(t), t) - w_j(\vec{p}^*(t), t) = \frac{\partial w_j^*}{\partial \vec{p}} \cdot \Delta \vec{p}(t) - \|\Delta \vec{p}(t)\|^2.$$

At a certain time t, if the optimal point $\vec{p}^*(t)$ makes the constraint w_j inactive ($w_j < 0$), so $\alpha_j = 0$, causing the sufficient condition in Eq.(54) to be satisfied. If the optimal point $\vec{p}^*(t)$ makes w_j active ($w_j = 0$) as shown in Figure 13, every neighborhood $\vec{p}(t)$ that lies in the left half plane specified by the tangent line at the optimal point $\vec{p}^*(t)$ will always cause $\partial w_j^*/\partial \vec{p} \cdot \Delta \vec{p} \leq 0$, and due to $\alpha_j \geq 0$, the sufficient condition in Eq.(54) is then satisfied. However, for the neighborhoods $\vec{p}(t)$ that lie in the right half plane and yield $w_j < 0$, they could produce $\partial w_j^*/\partial \vec{p} \cdot \Delta \vec{p} \geq 0$. As $\Delta \vec{p}(t)$ is infinitesimal, the admissible neighborhood $\vec{p}(t)$ within some small radius ϵ will cause $\partial w_j^*/\partial \vec{p} \cdot \Delta \vec{p} \to 0$, as shown in the bottom-right rectangle in Figure (13). Therefore, the sufficient condition in Eq.(54) is satisfied.

For the endpoint constraint $f(\vec{p}(t)) = 0$, the endpoint of the optimal trajectory ($\vec{p}^*(t_1)$) and the endpoint of the neighboring trajectory ($\vec{p}(t_1 + \delta t_1)$) must lie on this curve f, and because of the infinitesimal distance between these two endpoints, we obtain:

$$\left(\frac{\partial f^*}{\partial \vec{p}}\right)\bigg|_{t=t_1} \cdot \delta \vec{p}_1 = 0.$$

This results in the satisfaction to the sufficient condition in Eq.(55) no matter what γ is. For the graphical description, please see Figure 14.

The explanation for the satisfaction towards the sufficient condition in Eq.(56) is similar to one in Eq.(54) except that instead of examining the level curves $w_j(\vec{p}, t)$ at a specific time t as shown in Figure 13, the level surfaces emerge when the time t is considered. As virtual pedestrians are supposed to walk with constant velocity in MEM, the level surface are from extruding a circle in an upward direction

(in time direction), resulting in a slanted cylinder if a pedestrian walks with some speed; otherwise a straight cylinder. In case the optimal point $\vec{p}^*(t_1)$ makes the constraint w_j inactive ($w_j < 0$), so $\beta_j = 0$, causing the sufficient condition in Eq.(56) to be satisfied. Likewise, if the optimal point $\vec{p}^*(t_1)$ makes w_j active ($w_j = 0$), every neighborhood $\vec{p}(t_1 + \delta t_1)$ in an outer region specified by the tangent plane at the optimal point $\vec{p}^*(t_1)$ (similar to the left half plane in Figure 13) always satisfy Eq.(56). However, for the neighborhoods $\vec{p}(t_1 + \delta t_1)$ that lie in the inner region (similar to the right half plane in Figure 13) and yield $w_j < 0$, they could cause the left-hand side of Eq.(56) to greater than or equal to zero. But $\delta \vec{p}_1$ is infinitesimal, those terms will approach to zero. Therefore, the sufficient condition in Eq.(56) is satisfied.

As all sufficient conditions are attained, any position $\vec{p}(t)$ and the velocity $\vec{v}(t)$ that conform to the energy-minimal walking characteristics, defined by Eq.(7) - (10), are the solution that yields the (local) minimum walking energy.

Appendix B   Farkas's Lemma

Let $\vec{q} \in \mathcal{R}^n$ and $\vec{x} \in \mathcal{R}^n$ be n-dimensional vectors, and let $\mathbf{A} \in \mathcal{R}^{m \times n}$ denote a real $m \times n$ matrix:

$$
\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}
$$

Then, the statement that:

- $\vec{q} \cdot \vec{x} \leq 0$ for all $\vec{x}$ such that $\mathbf{A}\vec{x} \geq 0$

is equivalent to the statement that:

- there exists an m-dimensional vector $\vec{v} \in \mathcal{R}^m$ such that $\vec{v}^T \mathbf{A} + \vec{q}^T = 0$.

References

[1] G. K. Zipf, *Human Behavior and the Principle of Least Effort.*   Addison-Wesley, 1949.

[2] J. Richards, *Biomechanics in clinic and research: an interactive teaching and learning resource.* Churchill Livingstone, 2008.

[3] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 25–34, Aug. 1987.

[4] S. M. LaValle, *Planning Algorithms.*   New York, NY, USA: Cambridge University Press, 2006.

[5] W. Shao and D. Terzopoulos, "Environmental modeling for autonomous virtual pedestrians," in *SAE Symposium on Digital Human Modeling for Design and Engineering*, 2005.

[6] A. Botea, M. Müller, and J. Schaeffer, "Near optimal hierarchical path-finding," *Journal of Game Development*, vol. 1, pp. 7–28, 2004.

[7] A. W. Kring, A. J. Champandard, and N. Samarin, "Dhpa* and shpa*: Efficient hierarchical pathfinding in dynamic and static game worlds." in *AIIDE*, G. M. Youngblood and V. Bulitko, Eds.   The AAAI Press, 2010.

[8] M. T. Rantanen and M. Juhola, "Using probabilistic roadmaps in changing environments," *Computer Animation and Virtual Worlds*, vol. 25, no. 1, pp. 17–31, 2014.

[9] F. Lamarche and S. Donikian, "Crowd of virtual humans: a new approach for real time navigation in complex and structured environments," *Computer Graphics Forum*, vol. 23, no. 3, pp. 509–518, 2004.

[10] J. Pettré, P. d. H. Ciechomski, J. Ma⬛m, B. Yersin, J.-P. Laumond, and D. Thalmann, "Real-time navigating crowds: scalable simulation and rendering," *Computer Animation and Virtual Worlds*, vol. 17, no. 3-4, pp. 445–455, 2006.

[11] A. Sud, R. Gayle, E. Andersen, S. Guy, M. Lin, and D. Manocha, "Real-time navigation of independent agents using adaptive roadmaps," in *Proceedings of the 2007 ACM Symposium on Virtual Reality Software and Technology*, ser. VRST '07. New York, NY, USA: ACM, 2007, pp. 99–106.

[12] A. Sud, E. Andersen, S. Curtis, M. Lin, and D. Manocha, "Real-time path planning for virtual agents in dynamic environments," in *ACM SIGGRAPH 2008 Classes*, ser. SIGGRAPH '08. New York, NY, USA: ACM, 2008, pp. 55:1–55:9.

[13] R. Geraerts and M. H. Overmars, "The corridor map method: A general framework for real-time high-quality path planning: Research articles," *Comput. Animat. Virtual Worlds*, vol. 18, no. 2, pp. 107–119, May 2007.

[14] I. Karamouzas, R. Geraerts, and M. Overmars, "Indicative routes for path planning and crowd simulation," in *Proceedings of the 4th International Conference on Foundations of Digital Games*, ser. FDG '09. New York, NY, USA: ACM, 2009, pp. 113–120.

[15] W. G. van Toll, A. F. Cook, and R. Geraerts, "A navigation mesh for dynamic environments," *Comput. Animat. Virtual Worlds*, vol. 23, no. 6, pp. 535–546, Nov. 2012.

[16] A. Kumar and A. Ojha, "Natural path planning using wavelet noise in static environment," *Computer Animation and Virtual Worlds*, vol. 24, no. 1, pp. 17–24, 2013.

[17] N. Jaklin, A. F. C. IV, and R. Geraerts, "Real-time path planning in heterogeneous environments." *Journal of Visualization and Computer Animation*, vol. 24, no. 3-4, pp. 285–295, 2013.

[18] W. G. van Toll, A. F. Cook, IV, and R. Geraerts, "Real-time density-based crowd simulation," *Comput. Animat. Virtual Worlds*, vol. 23, no. 1, pp. 59–69, Feb. 2012.

[19] S. Sarmady, F. Haron, and A. Z. Talib, "Simulating crowd movements using fine grid cellular automata," *12th International Conference on Computer Modeling and Simulation*, vol. 0, pp. 428–433, 2010.

[20] S. Bandini, F. Rubagotti, G. Vizzari, and K. Shimura, "A cellular automata based model for pedestrian and group dynamics: Motivations and first experiments," in *Parallel Computing Technologies*, ser. Lecture Notes in Computer Science, V. Malyshkin, Ed., vol. 6873. Springer Berlin Heidelberg, 2011, pp. 125–139.

[21] L. Fischer, R. Silveira, and L. Nedel, "Gpu accelerated path-planning for multi-agents in virtual environments," in *Games and Digital Entertainment (SBGAMES), 2009 VIII Brazilian Symposium on*, Oct 2009, pp. 101–110.

[22] M. Kapadia, S. Singh, W. Hewlett, and P. Faloutsos, "Egocentric affordance fields in pedestrian steering," in *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games*, ser. I3D '09. New York, NY, USA: ACM, 2009, pp. 215–223.

[23] C. W. Reynolds, "Steering behaviors for autonomous characters," in *the proceedings of Game Developers Conference 1999*, 1999, pp. 763–782.

[24] C. Hartman and B. Benes, "Autonomous boids," *Comput. Animat. Virtual Worlds*, vol. 17, no. 3-4, pp. 199–206, Jul. 2006.

[25] W. Shao and D. Terzopoulos, "Autonomous pedestrians," *Graph. Models*, vol. 69, no. 5-6, pp. 246–274, Sep. 2007.

[26] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, pp. 4282–4286, May 1995.

[27] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, pp. 487–490, 2000.

[28] N. Pelechano, J. M. Allbeck, and N. I. Badler, "Controlling individual agents in high-density crowd simulation," in *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '07. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2007, pp. 99–108.

[29] S. Paris, J. Pettré, and S. Donikian, "Pedestrian reactive navigation for crowd simulation: a predictive approach," *Computer Graphics Forum*, vol. 26, no. 3, pp. 665–674, 2007.

[30] J. Pettré, J. Ondřej, A.-H. Olivier, A. Cretual, and S. Donikian, "Experiment-based modeling, simulation and validation of interactions between virtual walkers," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '09. New York, NY, USA: ACM, 2009, pp. 189–198.

[31] A. Golas, R. Narain, and M. Lin, "Hybrid long-range collision avoidance for crowd simulation," in *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, ser. I3D '13. New York, NY, USA: ACM, 2013, pp. 29–36.

[32] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2008, pp. 1928–1935.

[33] J. van den Berg, S. Patil, J. Sewall, D. Manocha, and M. Lin, "Interactive navigation of multiple agents in crowded environments," in *Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games*, ser. I3D '08. New York, NY, USA: ACM, 2008, pp. 139–147.

[34] S. J. Guy, J. Chhugani, C. Kim, N. Satish, M. Lin, D. Manocha, and P. Dubey, "Clearpath: Highly parallel collision avoidance for multi-agent simulation," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '09. New York, NY, USA: ACM, 2009, pp. 177–187.

[35] J. van den Berg, S. J. Guy, M. C. Lin, and D. Manocha, "Reciprocal n-body collision avoidance." in *ISRR*, ser. Springer Tracts in Advanced Robotics, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds., vol. 70. Springer, 2009, pp. 3–19.

[36] P. Fiorini and Z. Shillert, "Motion planning in dynamic environments using velocity obstacles," *International Journal of Robotics Research*, vol. 17, pp. 760–772, 1998.

[37] S. R. Musse, C. R. Jung, J. C. S. Jacques, Jr., and A. Braun, "Using computer vision to simulate the motion of virtual agents: Research articles," *Comput. Animat. Virtual Worlds*, vol. 18, no. 2, pp. 83–93, May 2007.

[38] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," *Computer Graphics Forum*, vol. 26, no. 3, pp. 655–664, 2007.

[39] A. Lerner, E. Fitusi, Y. Chrysanthou, and D. Cohen-Or, "Fitting behaviors to pedestrian simulations," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '09. New York, NY, USA: ACM, 2009, pp. 199–208.

[40] L. Sun, X. Li, and W. Qin, "Simulating realistic crowd based on agent trajectories," *Computer Animation and Virtual Worlds*, vol. 24, no. 3-4, pp. 165–172, 2013.

[41] S. J. Rymill and N. A. Dodgson, "A Psychologically-Based Simulation of Human Behaviour," in *EG UK Theory and Practice of Computer Graphics*, L. M. Lever and M. McDerby, Eds. The Eurographics Association, 2005.

[42] S. J. Guy, S. Kim, M. C. Lin, and D. Manocha, "Simulating heterogeneous crowd behaviors using personality trait theory," in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '11. New York, NY, USA: ACM, 2011, pp. 43–52.

[43] J. Ondřej, J. Pettré, A.-H. Olivier, and S. Donikian, "A synthetic-vision based steering approach for crowd simulation," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 123:1–123:9, Jul. 2010.

[44] A. Treuille, S. Cooper, and Z. Popović, "Continuum crowds," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 1160–1168, Jul. 2006.

[45] R. Narain, A. Golas, S. Curtis, and M. C. Lin, "Aggregate dynamics for dense crowd simulation," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 122:1–122:8, Dec. 2009.

[46] Y. Tajima, K. Takimoto, and T. Nagatani, "Pattern formation and jamming transition in pedestrian counter flow," *Physica A: Statistical Mechanics and its Applications*, vol. 313, no. 3–4, pp. 709 – 723, 2002.

[47] M. Isobe, T. Adachi, and T. Nagatani, "Experiment and simulation of pedestrian counter flow," *Physica A: Statistical Mechanics and its Applications*, vol. 336, no. 3–4, pp. 638 – 650, 2004.

[48] V. J. Blue and J. L. Adler, "Cellular automata microsimulation for modeling bi-directional pedestrian walkways," *Transportation Research Part B: Methodological*, vol. 35, no. 3, pp. 293 – 312, 2001.

[49] L. Xiang and L.-Y. Dong, "Modeling and simulation of pedestrian counter flow on a crosswalk," *Chinese Physics Letters*, vol. 29, no. 9, pp. 9 – 13, 2012.

[50] J. Gonz□lez, M. Sandoval, and J. Delgado, "Social field model to simulate bidirectional pedestrian flow using cellular automata," in *Traffic and Granular Flow '11*, V. V. Kozlov, A. P. Buslaev, A. S. Bugaev, M. V. Yashina, A. Schadschneider, and M. Schreckenberg, Eds. Springer Berlin Heidelberg, 2013, pp. 197–206.

[51] S. Heliövaara, T. Korhonen, S. Hostikka, and H. Ehtamo, "Counterflow model for agent-based simulation of crowd dynamics," *Building and Environment*, vol. 48, pp. 89 – 100, 2012.

[52] J. Yuen and E. Lee, "The effect of overtaking behavior on unidirectional pedestrian flow," *Safety Science*, vol. 50, no. 8, pp. 1704 – 1714, 2012, evacuation and Pedestrian Dynamics.

[53] X. Ji, X. Zhou, and B. Ran, "A cell-based study on pedestrian acceleration and overtaking in a transfer station corridor," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 8, pp. 1828 – 1839, 2013.

[54] J. Bruneau and J. Pettré, "Energy-efficient mid-term strategies for collision avoidance in crowd simulation," in *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, ser. SCA '15. New York, NY, USA: ACM, 2015, pp. 119–127.

[55] E. Hall, *The Hidden Dimension*, ser. A Doubleday anchor book. Anchor Books, 1990.

[56] S. J. Guy, J. Chhugani, S. Curtis, P. Dubey, M. Lin, and D. Manocha, "Pledestrians: A least-effort approach to crowd simulation," in *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '10. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2010, pp. 119–128.

[57] A. Seierstad and K. Syds□ter, *Optimal Control Theory with Economic Applications*. Elsevier Science, 1987.

[58] M. I. Kamien and N. L. Schwartz, *Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management (Advanced Textbooks in Economics)*. Elsevier Science; 2nd edition (October 25, 1991), 1991.

[59] D. Kirk, *Optimal Control Theory: An Introduction*, ser. Dover Books on Electrical Engineering Series. Dover Publications, 2004.

[60] J. Zhang, W. Klingsch, A. Schadschneider, and A. Seyfried, "Ordering in bidirectional pedestrian flows and its influence on the fundamental diagram," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2, p. 02002, Feb. 2012.

[61] P. P. Kachroo, S. A. Wadoo, S. J. Al-nasur, and A. Shende, *Pedestrian Dynamics*. Springer Berlin Heidelberg, 2008.