CSC 497 Fall 2022
University of Victoria

Integration of Humanoid Animation with Footsteps Navigation for Realistic Crowd Simulation

Julia Putko
V00889506

Abstract

Being able to model a realistic animation of a human on top of a footstep plan that adheres to biological constraints is a crucial step to being able to model crowds. This paper focuses on realism in agent representation as well as in agent navigation with the intention of producing realistic crowd models that can be used to represent real diverse crowds for a variety of applications. Decisions to combine a footstep model and realistic humanoid animation are explored by researching various methods related to character animation, crowd simulation and navigation control in a multi-agent dynamic environment. This paper records the work done to begin the integration of the virtual humans from the work in realistic humanoid animation including full body biped inverse kinematics, and barycentric blending with a footstep based navigation planner. Integrating navigation derived from multiple agent reinforcement learning is also explored. The result of the current work hopes to lead to humanoid animations that follow realistic navigation paths, are able to resolve collisions in a natural manner, and can represent diverse human motion. An investigation into various analysis techniques are proposed that can be later utilized to compare the completed current method against other methods to further confirm its usefulness.

1. Introduction

Realistic crowd simulation involves modeling the movement of several individual agents, representing humans, in a dynamic environment. Key factors in crowd simulation are the representation of the agent, how an individual agent navigates to their goal, and the environment itself, including obstacles and other agents that may cause changes to the navigation plan of the agent. Individual actions of a collection of agents is reflected in the movement of a crowd, influenced by the interaction between agents and the structure of the environment. Different approaches to agent representation and navigation and the type of scenarios agents are put in can result in dramatically different results. Another important consideration is the purpose of choosing a particular representation or navigation type for an agent. Crowd simulation has important practical applications in determining evacuation times (Wong, et al., 2017), ensuring safety during crowded events and managing safe capacity (Mahmood, Haris, & Sarjoughian, 2017), improving crowd flow (Sharma, Bhondekar, Shukla, & Ghanshyam, 2018), and evaluating social distance policies in different environments in the interest of public health (Usman, et al., 2020), among other things. Realistic crowd simulators can better inform further research within these areas. In natural crowds the flow of people through an environment form meaningful structures that can be analyzed. A crowd simulation technique that mimics real crowd behaviors should respect realistic constraints of biological motion, allow for individual agents to act differently within a crowd, and allow for adjustments to represent different crowd scenarios in different environments and with different goals.

Realistic animations that include agents with natural looking motions and navigation and can produce realistic crowd models. It is important to note that what is considered natural human motion in this paper is extremely simplified to a bipedal humanoid figure that respects a simplification of biological constraints and does not represent the diversity of motion that exists within the global population. The animation mechanism proposed in Shoulson, Marshak, Kapadia & Badler (2013), and developed in the work of Beacco, Pelechano, Kapadia, & Badler (2015) is chosen for this project to allow for full body character control that can be combined with navigation. They produce a complex representation of agents that is also computationally efficient and can be used to represent crowds of people while being able to represent the subtleties of human motion. In order to integrate the animation of a biped figure with footstep position further, an inverse kinematics technique developed by Aristidou & Lasenby, (2011) is proposed. It is crucial for the purposes of this paper to include an inverse kinematic solution that produces realistic motion and is suitable for real time crowd simulation. In order to produce realistic navigation behavior, a footstep controller is chosen for this project (Singh, Kapadia, Reinman, & Faloutsos, 2011; Berseth, Kapadia, & Faloutsos, 2015) to allow for more refined character control, and more precise navigation in complex scenarios where the environment is dynamic, including confined and densely packed spaces with high likelihood of collisions. The footstep model uses steering, informed by the biological motion, that begins at the level of footstep placement allowing for subtle motions such as sidestepping that result in the production of natural looking collision free paths. A multi-agent reinforcement learning framework allows the footstep navigation paths to be revised dynamically (Hu, Haworth, et al., 2021). This paper explores this framework for future integration purposes as it allows for reasoning abilities for agents that mirror real crowds by implementing reinforcement learning techniques that model real cognitive processes by implementing rewards that reward realistic human motion.

Simulations of agents should be able to interact with different environments in appropriate ways that mimic real scenarios of crowd motion. Agents in crowd simulations should be able to resolve complex environments, such as those with concavities and narrow doorways. It is important to have path planning algorithms that can be successful in scenarios that are not known before the start of the simulation. The purpose of this paper is to focus on an underlying representation of an agent that can capture the complexities in human motion through a humanoid representation, and integrate this representation with a navigation planning method that allows for character autonomy and the ability to an resolve dynamic scenarios with varying other obstacles and agents and investigate how this integration would produce the goal of creating a crowd simulation that overcomes limitations of past models and techniques.

2. Related Work

Navigation and character representation techniques considered for this work should model the constraints and capabilities of human movement, and support character control in complex scenarios.

Crowd simulations using particle models represent human navigation as particles with force or velocity vectors as navigation decisions, based on the concept that repulsion between particles in physics mimics how humans might avoid one another in crowds. The social forces model (Helbing & Molnar, 1995) achieves collision avoidance by applying repulsion forces to particle agents and obstacles to prevent agents from colliding with one another and achieving collision free paths to their goal. While computationally efficient, the results of social forces models suffer from the downfalls of particle models, and do not resemble human movement. The work of Pelechano, Allbeck, & Badler (2007) is related to the problem of simulating the motion of realistic crowds with autonomous agents. They extend the social forces model to attempt to overcome limitations of the social forces model that results in more realistic emergent crowd behaviors. Particle models, however, do not have the capacity to represent the subtleties and constraints of human motion and cannot inform the agent to perform complex navigation movements. Parameters are limited to speed and particle diameter to represent diversity within crowds. This limited capacity for movement, ultimately results in motion that is unrealistic. Xiong, Lees, Cai, Zhou, & Low (2009) developed a rule-based motion planning for agent-based crowd simulation. Agents are categorized according to their inner states with the intention of modeling how agents behave in real life. The model chooses velocities whose expected time to collision is above a certain threshold, and the largest expected time otherwise which may still result in a collision. Rule based models cannot produce realistic movement in complex high dense scenarios such as an evacuation scenario. Karamouzas, Skinner, & Guy (2014) use the fact that crowd interactions resemble particle interactions, in that agent interactions look like the repulsive potential energies between particles, and the analysis of human motion data, to uncover a power law within human crowds that relies on projected time to future collisions. This work using particle models uncovered that human crowd interactions are anticipatory.

Predictive models were produced to avoid collisions in a natural way, allowing for navigation in more compact scenarios. These models improve upon past models that produce unnatural motion such as pushing back and forth between agents. Based on the fact that humans make steering decisions a few steps in advance Karamouzas, Heil, Beek, & Overmars (2009) developed a method of collision avoidance using prediction of future locations based on current velocities of other agents. The result of their work was more natural collision avoidance, that also displayed emergent crowd behavior such as lane formation. This work reflected human behavior in that humans only pay attention to pedestrians they are likely to collide with, and predict their motions only a few steps in advance. Van den Berg, Lin, & Manocha (2008) developed a technique for collision avoidance that includes reciprocal velocity obstacles, and explored the need for developing a model with multi-agent navigation capabilities. The collision avoidance technique assumes that other agents have similar collision avoidance reasoning to produce multi-agent, autonomous navigation in real time.

Data driven models improve the realism of simulations, improving upon particle models, but acquiring and interpreting real time data is difficult. Data driven models extract steering behaviors from motion data of real crowd interactions, and produce motion that resembles human motion, but may not be able to handle complex navigation or constrained environments because it relies on a limited set of data. The research of van Basten, Peeters, & Egges (2010) looks into the placement of feet during character locomotion by extracting steps from motion data, highlighting the importance of exact foot placement during navigation for realistic locomotion, in constrained scenarios, but also for computational efficiency. They use motion capture data to get steps that are then blended together in order to create a footsteps plan. These models are more focused on general crowd behaviors than individual steering decisions, and have limitations modeling a larger variety of navigation behaviors with a limited dataset. There is a limit to the amount of crowd data that is available, and definitely does not encompass the variety of scenarios that we would like to represent in crowd simulations, including types of scenarios, possible interactions between other agents or obstacles, or the diversity of pedestrians globally. There has been further work in this area exploring creating natural looking locomotion and animating human-like characters using motion capture data (Kuffner, 1998) (Pettré, Laumond, & Siméon, 2003) (Choi, Lee, & Shin 2003). Zhang, Pan, & Manocha (2009) concluded that motion planning and motion capture methods cannot produce natural looking human motion on their own and so their work exploited hybrid techniques in order to try to achieve more realistic and natural looking motion in constrained environments with multiple obstacles.

An additional step in the direction of creating human-like navigation is to consider human motion from the perspective of footstep planning, which has been explored in both the animation and in the robotics community. Footstep models allow for tighter bounds around agents to allow for more realistic interaction between agents and obstacles and other agents than in particle models. Kuffner et al. (2003) proposed an online algorithm that computed footstep sequences to a specific goal location for humanoid robots. This work was done in the context of creating software and algorithms for humanoid robots and not for realistic movement of real humans. Movements created for humanoid robots are generally efficient and mechanical, but are not natural looking. An optimization-based scheme was proposed by van de Panne (1997) to produce biped locomotion from given footprints, with a focus on developing natural motion. This work expressed the need for work to improve fidelity in muscle joints of the animated character. Realistic animation is an important consideration to provide crowds models with a realistic radius and movement based on biological constraints. Geijtenbeek, Van De Panne, & Van Der Stappen (2013) proposed a muscle based control method for simulated bipeds that respect biomechanical constraints for realistic motion using physics based animation.

It is also important to consider evaluation techniques to determine how well crowd simulation techniques represent real world crowds qualitatively and quantitatively. Karamouzas, Sohre, Hu, & Guy (2018) propose a method to perform comparisons between simulation models to determine which crowd simulation technique should be used for scenarios. Kapadia, Wang, Singh, Reinman, & Faloutsos (2011) highlighted the challenges of evaluating steering algorithms. They evaluated the algorithms based on scenario completion, total time and path length, and used these metrics to compare across algorithms and scenarios in challenging scenarios. They recognized that the steering algorithms analyzed do not include sidestepping and careful foot placement that is possible with the footsteps model. Haworth (2019) developed an evaluation technique that allows for the comparison of different steering methods quantitatively and qualitatively. It was shown that the performance of the footstep models described in Berseth, Kapadia, & Faloutsos (2015) and Singh, Kapadia, Reinman, & Faloutsos (2011) reflect their purpose of generating natural motion which may result in longer paths or a longer time to reach goals, but ultimately avoids the shortcomings in other models just as unnatural particle motion when collisions happen in particle based steering models, the inability to solve certain scenarios.

3. Overview

In order to achieve the ideals described above, a combination of intelligent footstep placement based on biological motion, a realistic animator that can be used in crowd simulations with hundreds of agents, supplemented with a inverse kinematics technique for realistic integration, and a reinforcement learning technique can be implemented in order to generate a model that can produce realistic crowd simulations. More realistic movement as well as spatially more realistic bounds that are informed by realistic human animation.

3.1 Animation

The agent development and prototyping testbed (ADAPT) (Shoulson, et al. 2013) provides a feature set for animation for full body control, navigation and behavior, and allows for the coupling of animation controllers with navigation and steering. It allows for the incorporation of new steering systems and provides the ability to modify components without making other components unusable. This characteristic makes it an ideal platform to animate realistic virtual humans in multi-agent scenarios and integrating a navigation technique.

3.1.1 Full Body Humanoid Animation

Beacco, et al. (2015) combine natural looking animated virtual humans with footstep trajectories using the animation technique developed in Shoulson, et al. (2013). The animated humans in Beacco, et al (2015) follow footstep trajectories allowing for the ability to control the different styles of walking and the type of steps. From a small set of motion clips, parametric spaces of the left and right foot are computed and the individual root of the parametric space is used to compute a weight for each available animation based on root velocity. The weights are used to synthesize animations that accurately follow footsteps. Given a simplex containing the next desired footstep position, derived from the computed delaunay triangulation calculated for the possible landing positions, the vertices of the simplex are used to compute the blending weights for the corresponding animations that can be used to arrive at that footstep destination. There may be a small offset between the final landing position of the foot and the position given by the footstep trajectory, and so an inverse kinematic solution is also implemented to improve the foot placement along the trajectory. This method allows for different walking gaits, the possibility to alternate between running and walking phases and includes sidesteps and backward motion. It also allows for scaling to handle large crowds of characters in crowd simulations. This paper takes this animation blending technique and the inverse kinematic correction used, but applies it to a different footstep trajectory.

3.1.2 Inverse Kinematics

Inverse kinematics allows for the determination of joint configurations of a humanoid figure given end effector locations. Forward and backward reaching inverse kinematics (FABRIK)(Aristidou & Lasenby, 2011) approach is proposed to be used for solving the full body biped inverse kinematics (FBBIK) problem. This solution treats the problem of finding each joint location as the problem of finding points on a line, which saves time and computational costs, instead of calculating the rotational angles of matrices. This solution is considered the ideal method for pursuing inverse kinematics as it does not suffer from the drawbacks of other inverse kinematics models. Other models such as cyclic coordinate descent (CCD) (Wang, & Chen, 1991), and Jacobian inverse kinematics described in Buss (2004) result in unrealistic postures. The Jacobian method has very high computation costs, produces unrealistic movement when the end effector is far from the target. FABRIK produces smooth motion, and works well with multiple end effectors. It updates the joint position along a line to the next joint, working in a forward and backward iteratively mode it is able to minimize the distance between the target and the end effects at each time. FABRIK can be easily constrained by adding angle rotational restrictions to each joint and results in more realistic postures for humanoid figures.

Given joint positions in the kinematic chain $p_1, \dots, p_n$ where p_1 is the root joint and p_n is the end effector with a target at t. First calculating the distances between each joint position $d_i = |p_{i+1} - p_i|$ for $i = 1 \dots n-1$. Then, check whether the target is reachable by finding out if the distance between the root and the target is within an appropriate distance. The algorithm consists of two stages: a forward iteration and a backward iteration. In the forward stage the algorithm estimates each joint position starting from the end effector p_n and moving inwards to the base p_i. The new position of the end effector is assigned to the target position $p_i' = t$. Then, find the line l_{n-1} which passes through the joint position p_{n-1} and p_n'. The new position of the $(n-1)th$ joint p_{n-1}' lies on that line with distance d_{n-1}. Calculated for all i. The algorithm continues until all joint positions are updated. In the backward stage of the algorithm, the same procedure is repeated, but moving backwards from the manipulator's base to the end effector. This completes one full iteration. By the end of a certain number of iterations, the end effector will be close to the target. The resulting algorithm is simple, flexible, has a low computational cost, and supports biological joint constraints, reaching the target fast and with natural movement.

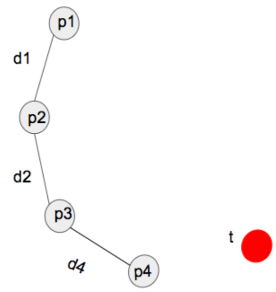

Figure 1: The initial position of joints represented by p_1 to p_4 with p_4 as the end effector and t as the target.

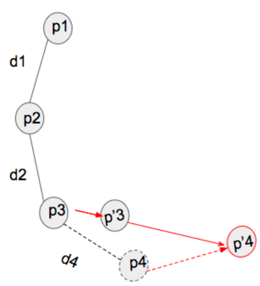

Figure 2: The algorithm begins with p'4 taking on the position of t. A line is drawn from p_3 to p_4' to derive the position of joint p'3. Once this has been completed for all joint positions, it is repeated, but instead starting from moving p_1' to the starting position and performing the steps from p_1' to p_4'. By the end of the last interaction the end effector will be closer to the target.

3.2 Navigation

The footsteps model described in the work by Singh, Kapadia, Reinman, & Faloutsos (2011) creates a footstep path to a goal in an environment that is informed by biomechanics of real human motion to create realistic navigation behaviors. The footstep is determined from the position, velocity, and timing of the agent's center of mass, the location and orientation of the foot and the cost of taking a step. The model uses the concept of a footstep plan to inform movement as it enables realistic avoidance behavior that does not cause agents to change directions in unnatural ways or sliding.

Following the path of the center of mass of a walking human reveals that it resembles a curve over a trajectory of a sphere. Within biomechanics, this is modeled using an inverted pendulum. Projecting this motion onto the ground, offset by shoulder width of an individual, can create a possible trajectory of footsteps informed by this motion. The parabolic curve of this projection of the inverted pendulum can have a certain degree of curvature depending on alpha in equation $y = \alpha t^2$. A lower α represents a straighter walking path and a higher α results in more turning during walking. Given the desired orientation of the parabola, the desired initial speed of the center of mass and the desired time duration of the step, the next footstep action can be derived.

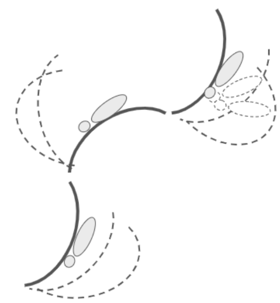

Figure 3: A possible footstep trajectory with possible curvatures of parabolas and possible foot orientations represented by dotted lines. The resulting trajectory represents one that would minimize the cost function.

Footsteps chosen to the goal are determined according to a cost function which is derived from the assumption that humans take low effort paths towards their goal. The cost of transition between two steps is given as $\Delta E_1 + \Delta E_2 + \Delta E_3$. ΔE_1 is the fixed rate of energy the character spends per unit of time, proportional to the amount of time it takes for the agent to reach its goal. ΔE_2 is the cost of effort to choose a certain speed, based on the exertion of additional work on the center mass to make up for the loss of momented caused at the beginning of a new step. Changing of momentum due to abrupt changes in direction have a high cost. ΔE_3 is the cost associated with changing a character's momentum at the beginning of each step. Minimizing these three costs minimizes the time it takes for the agent to reach its goal, prioritizes finding footsteps that require less effort, and results in the agent taking less curved steps, and the character preferring to take smaller steps when changing direction ultimately leading to more natural looking footsteps.

To determine the path to the goal Singh, Kapadia, Reinman, & Faloutsos, (2011) use an A* search path that constrains the search of nodes to a maximum number of nodes that can be expanded, and then a footstep path is decided within the constraints of the A* search path.

Berseth, Kapadia, & Faloutsos (2015) improvement upon the footsteps model by developing a method that is robust to challenging scenario configurations.Their method implements footstep path planning that avoid undesirable initial footstep configurations and optimizing short term path planning between A* waypoints to avoid invalid footsteps.

The intention of this paper is also to explore the further integration of the animation with navigation from multiple agent reinforcement learning described in Hu, et al. (2021). Hu, et al. (2021) exploits the use of multiple agent reinforcement learning (MARL) to solve the problem of steering in crowd simulations, allowing for generality, heterogeneity, and anticipatory reciprocal collision avoidance. The method uses a single reinforcement learning policy that is shared across agents, but can be parametrized which enables heterogeneous behaviors across agents. Agents have individual goals, and preferred velocities, and move according to them. This work makes further steps towards high fidelity humanoid crowds.

The results of this paper are the attempts to combine this method of determining navigation with the animation framework described above.

3.3 Integration Results

Efforts were made towards the integration of the animation works described in Beacco, et al. (2015) and Shoulson, et al. (2013) with the navigation work developed in Singh, et al. (2011) and Berseth, Kapadia & Faloutsos, (2015), as well as the work reinforcement learning techniques described in Hu, et al. (2021). An understanding of the animation and navigation methods was reviewed in order to understand the purpose and desired results from this work, and the feasibility of the future integration was explored. The implementation of the work described in Beacco et al. (2015) and Shoulson, et al. (2013) was done within the Unity game engine (Unity, 2014). Progress was made in understanding components within the animation system, and exploring possible points that would allow for the integration of a footsteps navigation system and inverse kinematics solver.

The animation framework was extracted from [github.com/Rutgers-CG/KADAPT] (Beacco, et al. (2015). The Unity project was made usable in a version of Unity being used to develop a footstep navigation approach from its existing version. Footstep navigation implementation and methods for integration were explored to understand the necessary components, and how integration might be possible. The result of the efforts was a virtual human ready to be integrated and an agent represented by a particle and spheres representing feet that follows a footstep trajectory that exist in a scene together, showing that the animation capabilities are able to exist in the same environment as the footsteps model, as well as the environment in which the MARL approach is being developed.

Figure 4: A virtual human in an updated version of Unity from the work described in Beacco, et al. (2015). A footstep planning path to be implemented the agent should follow is represented as the parabolic projection and footstep placement shown in figure, with a pink bar representing the difference between the footstep position in the plan and the actual agent footstep placement. This difference can be further corrected using the inverse kinematic method described, and any resulting differences can be evaluated by measuring the resulting distance between the footstep position and the agents' foot placement. The updated version is available at: github.com/juliaputko/KADAPT/tree/test-unity-.

The capabilities of the available code was explored, and basic movement was created using speed, velocity and angle that were estimated from a footstep navigation script described in Singh, et al. (2011). This estimated speed, angle and velocity were fed into a locomotion function in a navigation script. This provided the virtual human with an estimate of navigation, however did not make progress into following a footstep trajectory.

The availability of the inverse kinematics solver described in Aristidou & Lasenby, (2011) within the available code base was also explored.

4. Evaluation

Verifying the success of the resulting project can be done in several different stages. Evaluation techniques for specific components of the integrated result as well as comparative evaluation techniques can be used to determine if the completed integration is appropriate for use cases associated with crowd simulations.

Success of the chosen inverse kinematic solver should be evaluated qualitatively and quantitatively. The solver can be quantitatively examined by calculating the distance between the target position and the end effector. In this case the end effector will be the actual foot position of the virtual human, and the target will be the footstep position derived from the footstep trajectory described in Singh, et al. (2011). After a successful implementation of the inverse kinematics solver described, the distance between the actual foot position and the position of the output of the next footstep from the footstep trajectory should be minimized as much as possible. Performance of the inverse kinematic solver proposed can also be evaluated qualitatively by observing the resulting movement of the animated virtual human. An ideal inverse kinematic solution will result in smooth movement in order to reach the target. Smoother movements result in higher realism in animations.

Producing crowd simulations that model realism are computationally complex. Increasing the number of agents increases the computational efficiency decrease rapidly (Karamouzas, Sohre, Narain, & Guy, 2017). The feasibility of implementing these techniques into a crowd model can be examined by recording the computational performance of the model after the integration of techniques as the number of agents added increases. Realistic movement should be balanced with the ability to model multiple agents within the same model for the ability to study collision avoidance between multiple agents.

Qualitative evaluation can be used to determine whether navigation trajectories resemble those of real humans for individual agents. Comparing the result of the completed integration with the qualitative performance of other models can be useful in determining whether the completion of this integration would result in a more realistic model of human navigation.

5. Future work

Future work should be done to finalize the integration of the animation and the navigation models described and evaluation techniques described should be used to validate the results, for eventual integration with the multiple agent reinforcement learning technique described in Hu et al. (2021). Within real crowds, people do not always walk with the same motivations, abilities and speeds through environments, and there are variations in how humans reach their goals. Consideration should be made into other factors that support realism in human navigation within crowds. Realistically, real human crowds consist of individual differences in locomotion, which can reflect differences in ability that would be reflected in walking gait or speed, which influence crowd behaviors. Other biological aspects of bipedal locomotion could be explored in order to improve the footstep trajectory, as well as the function of the humanoid animation including the ankle and knee joints, the muscle action, and the shifting of weight in the body during locomotion.

There are other factors beyond footsteps that guide human navigation. Further work can be done in this area of determining how human cognition may have an impact on motivations behind steering and navigation decisions within humans, and how this can be implemented within simulations in order to appropriately reflect the subtleties that motivate real humans in crowds. Cognitive differences can also be considered when simulating diverse crowds. cognitive differences as well that may affect how agents decide to navigate to their goal, including the circumstances, individual biases, emotional states, energy levels. This motivation is especially important related to the goals of crowd simulation in evacuation scenarios, for example. Individual as well as social cognition and how it affects the behaviors of crowds should also be investigated to understand how it can be reflected in crowd modeling.

Conclusion

This paper explores an animation technique, an inverse kinematic technique, and a navigation technique to begin an integration in order to make progress towards producing a crowd simulation of agents that can perform locomotion within the constraints of human motion. Justification behind each technique was provided in the context of choosing methods that were successful in producing natural human motion. Separate evaluations concluded that, separately, the techniques described produce movement that resembles natural looking movement, and can be implemented on scales needed for crowd simulation. Progress was made to attempt integration of these techniques with the assumption that a combination of the techniques within a single crowd simulation would also produce results that align with the goal of producing a crowd simulation that resembles natural movement. Evaluation techniques were explored in order to validate a working integration in the future.

References

Aristidou, A., & Lasenby, J. (2011). FABRIK: A fast, iterative solver for the Inverse Kinematics problem. *Graphical Models*, *73*(5), 243-260.

Beacco, A., Pelechano, N., Kapadia, M., & Badler, N. I. (2015). Footstep parameterized motion blending using barycentric coordinates. *Computers & Graphics*, *47*, 105-112.

Berseth, G., Kapadia, M., & Faloutsos, P. (2015). Robust space-time footsteps for agent-based steering. *Computer Animation and Virtual Worlds*.

Buss, S. R. (2004). Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, *17*(1-19), 16.

Choi, M. G., Lee, J., & Shin, S. Y. (2003). Planning biped locomotion using motion capture data and probabilistic roadmaps. *ACM Transactions on Graphics (TOG)*, *22*(2), 182-203.

Diankov, R., Ratliff, N., Ferguson, D., Srinivasa, S., & Kuffner, J. (2008). Bispace planning: Concurrent multi-space exploration. *Proceedings of robotics: Science and systems IV*, *63*.

Egges, A., & van Basten, B. (2010). One step at a time: animating virtual characters based on foot placement. *The Visual Computer*, *26*(6), 497-503.

Geijtenbeek, T., Van De Panne, M., & Van Der Stappen, A. F. (2013). Flexible muscle-based locomotion for bipedal creatures. *ACM Transactions on Graphics (TOG)*, *32*(6), 1-11.

Haworth, M. B. (2019). Biomechanical locomotion heterogeneity in synthetic crowds.

Helbing, D., & Molnar, P. (1995). Social force model for pedestrian dynamics. *Physical review E*, *51*(5), 4282.

Hu, K., Haworth, B., Berseth, G., Pavlovic, V., Faloutsos, P., & Kapadia, M. (2021). Heterogeneous crowd simulation using parametric reinforcement learning. *IEEE Transactions on Visualization and Computer Graphics*.

Karamouzas, I., Heil, P., Beek, P. V., & Overmars, M. H. (2009, November). A predictive collision avoidance model for pedestrian simulation. In *International workshop on motion in games* (pp. 41-52). Springer, Berlin, Heidelberg.

Karamouzas, I., Sohre, N., Hu, R., & Guy, S. J. (2018). Crowd space: a predictive crowd analysis technique. *ACM Transactions on Graphics (TOG)*, *37*(6), 1-14.

Karamouzas, I., Skinner, B., & Guy, S. J. (2014). Universal power law governing pedestrian interactions. *Physical review letters*, *113*(23), 238701.

Karamouzas, I., Sohre, N., Narain, R., & Guy, S. J. (2017). Implicit crowds: Optimization integrator for robust crowd simulation. ACM Transactions on Graphics (TOG), 36(4), 1-13.

Kapadia, M., Wang, M., Singh, S., Reinman, G., & Faloutsos, P. (2011, August). Scenario space: characterizing coverage, quality, and failure of steering algorithms. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation* (pp. 53-62).

Kuffner, J. J. (1998, November). Goal-directed navigation for animated characters using real-time path planning and control. In *International Workshop on Capture Techniques for Virtual Environments* (pp. 171-186). Springer, Berlin, Heidelberg.

Kuffner, J., Kagami, S., Nishiwaki, K., Inaba, M., & Inoue, H. (2003, September). Online footstep planning for humanoid robots. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)* (Vol. 1, pp. 932-937). IEEE.

Kuffner, J. J. (1998, November). Goal-directed navigation for animated characters using real-time path planning and control. In *International Workshop on Capture Techniques for Virtual Environments* (pp. 171-186). Springer, Berlin, Heidelberg.

Mahmood, I., Haris, M., & Sarjoughian, H. (2017, May). Analyzing emergency evacuation strategies for mass gatherings using crowd simulation and analysis framework: Hajj scenario. In *Proceedings of the 2017 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation* (pp. 231-240).

Pelechano, N., Allbeck, J. M., & Badler, N. I. (2007). Controlling individual agents in high-density crowd simulation.

Pettré, J., Laumond, J. P., & Siméon, T. (2003, October). 3d collision avoidance for digital actors locomotion. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)* (Vol. 1, pp. 400-405). IEEE.

Sharma, D., Bhondekar, A. P., Shukla, A. K., & Ghanshyam, C. (2018). A review on technological advancements in crowd management. *Journal of Ambient Intelligence and Humanized Computing*, *9*(3), 485-495.

Shoulson, A., Marshak, N., Kapadia, M., & Badler, N. I. (2013). Adapt: the agent development and prototyping testbed. *IEEE Transactions on Visualization and Computer Graphics*, *20*(7), 1035-1047.

Singh, S., Kapadia, M., Reinman, G., & Faloutsos, P. (2011). Footstep navigation for dynamic crowds. *Computer Animation and Virtual Worlds*, *22*(2‐3), 151-158.

Unity, Unity—game engine,⟨http://unity3d.com/⟩; 2014.

Usman, M., Lee, T. C., Moghe, R., Zhang, X., Faloutsos, P., & Kapadia, M. (2020). A social distancing index: Evaluating navigational policies on human proximity using crowd simulations. In *Motion, Interaction and Games* (pp. 1-6).

van Basten, B. J., Peeters, P. W. A. M., & Egges, A. (2010). The step space: example‐based footprint‐driven motion synthesis. *Computer Animation and Virtual Worlds*, *21*(3‐4), 433-441.

Van den Berg, J., Lin, M., & Manocha, D. (2008, May). Reciprocal velocity obstacles for real-time multi-agent navigation. In *2008 IEEE international conference on robotics and automation* (pp. 1928-1935). Ieee.

Wang, L. C., & Chen, C. C. (1991). A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *IEEE Transactions on Robotics and Automation*, *7*(4), 489-499.

Wong, S. K., Wang, Y. S., Tang, P. K., & Tsai, T. Y. (2017). Optimized evacuation route based on crowd simulation. *Computational Visual Media*, *3*(3), 243-261.

Xiong, M., Lees, M., Cai, W., Zhou, S., & Low, M. Y. H. (2009, September). A rule-based motion planning for crowd simulation. In *2009 International Conference on CyberWorlds* (pp. 88-95). IEEE.

Zhang, L., Pan, J., & Manocha, D. (2009, November). Motion planning and synthesis of human-like characters in constrained environments. In *International Workshop on Motion in Games* (pp. 138-145). Springer, Berlin, Heidelberg.