When a Couple Goes Together: Walk along Steering

Markéta Popelová, Michal Bída, Cyril Brom, Jakub Gemrot, and Jakub Tomek

Charles University in Prague, Faculty of Mathematics and Physics,
Malostranské nám. 2/25, Prague, Czech Republic

Abstract. Steering techniques are widely used for navigation of single agents or crowds and flocks. Steerings also have the potential to coordinate movement of human-like agents in very small groups so that the resulting behavior appears socially believable, but this dimension is less explored. Here, we present one such "social" steering, the Walk Along steering for navigating a couple of agents to reach a certain place together. The results of a believability study with 26 human subjects who compared the new steering to the known Leader Following steering in eight different scenarios suggest the superiority of the Walk Along steering in social situations.

1 Introduction

Various types of entities can move in applications featuring 3D virtual reality, ranging from inanimate objects, such as rockets and vehicles, to virtual animals and humans. For controlling bodily movement of animate entities, called intelligent virtual agents (IVAs) henceforth, a three-layer architecture can be employed to a great advantage [15]. The top-layer, responsible for high-level action selection and searching for a path in a large environment using global information, a kind of map, is usually supplemented by a middle layer, which refines the high-level path employing a reactive approach that takes local information into account. The lowest layer, a physical/animation engine, is then responsible for actual locomotion.

A favorite reactive mechanism used to implement the middle layer is steering techniques (steerings) of Craig Reynolds. His first techniques served to steer large groups of virtual birds, herds and fish [14]. Later, these techniques were extended for general virtual agents [15]. Even though these techniques are simple, they can be used to steer large groups of agents and the resulting behavior looks naturally. At the same time, these techniques are deterministic, which helps with debugging, and computationally inexpensive. For these qualities, Reynolds' steerings and their derivations are useful in various applications: crowd simulations, safety modeling, traffic planning, computer games, films, educational applications etc.

Steerings may serve well for navigating single agents or agents in a crowd or a flock. Especially steerings for avoiding collisions with other IVAs or static obstacles are researched a lot, e.g., [4, 5, 6], or steerings for crowd/flock behavior, e.g., [2, 8].

However, steerings can also be used for other purposes than for purely mechanistic navigation: they can help a designer to express relations between agents, their personality or mood and other social traits. Even though some steerings are used for controlling human-like IVAs in intrinsically social situations, such as Seek, Flee, and

J.M. Allbeck and P. Faloutsos (Eds.): MIG 2011, LNCS 7060, pp. 278–289, 2011.

Leader Following, the social dimension of steerings has not been explored much. To our knowledge, only few works explored this line of research: the authors of [9, 10, 11] investigated group conversation dynamics and human territorial behavior during social interactions, and the authors of [7, 12] researched behavior of small groups.

We are interested in this second way of using the steering approach: for social expressions of human-like IVAs. One of main motivations is the usage of steerings in our educational micro-game Cinema Date [1]. The game features two virtual teenagers dating on their way to the cinema. The player influences the course of the date by shaping behavior of one of the characters with a particular game goal.

In this paper, we present one result from this ground – a new steering named Walk Along (WA), which we designed and implemented. The goal of this steering is to steer two virtual agents who go to a certain place together. Ideally, the agents should go side by side. It may happen that the IVAs would need to avoid static and dynamic obstacles on their way, pass narrow corridors consecutively etc. However as they return to free space, they should walk side by side again. Also, it may happen that one of the IVAs will get stuck or delayed along their way (e.g., that it stops and watches a shop-window, lets a car go by, etc.). The second IVA should notice this situation and wait for the first IVA or return for it.

In general, the WA steering is supposed to navigate agents in open spaces, not inside small enclosed rooms. At the same time, its main purpose is to create single pairs of people, not crowds composed of pairs of people (although such usage could also be possible). To create a perfect impression of two friends going together to a certain place, it would be necessary to use appropriate animations expressing what they are talking about, their mood, emotions and other social behavior. These issues are out of scope of this paper (but see [1]).

A possible solution of the walk along assignment is the steering Leader Following (LF) by Reynolds [15], extended for following the leader at a certain position – next to the leader in our case. One of the two walking IVAs would be the leader and the other would follow it on its side. This solution has certain disadvantages. In the basic form of that steering, the leader does not wait for its follower(s). For instance, if a follower gets stuck, the leader keeps going to its target, which is not a very plausible behavior (unless the leader is angry at the follower). In the WA steering, both agents have the same role and give the impression of more balanced relationship.

In our evaluation with human subjects, we compare the WA steering to the modified LF in eight different scenarios with the intention to demonstrate possibilities and limitations of the new steering. For the purposes of the evaluation, we also implemented six other steerings which may be combined together and with the WA. These steerings are based on the Reynolds' seminal work [15].

The paper proceeds as follows: general architecture and six implemented steerings are briefly described in Section 2. The Walk Along steering is detailed in Section 3. Section 4 describes the evaluation results and Section 5 concludes.

2 General Architecture and Implemented Steering Behaviors

We implemented the three-layer architecture for controlling motion of IVAs designed by Reynolds [15]. It is composed of an *action selection* layer, a *steering* layer and a *locomotion* layer. At the action selection layer, it is decided which steerings are active and how they are parameterized. The steering layer computes the velocity of a steered agent in the next tick (the time in our simulation is discrete). The locomotion layer moves the agent according to the given velocity.

Every steering has the following information: the current location and velocity of the agent, and locations and velocities of all other visible agents (in 180° range). Inanimate obstacles are detected by five rays: one front ray, two short lateral and two long front-lateral rays (note that we tested the WA steering also with an agent with seven and nine rays). The result of computation of each steering in one tick is a single force vector. By combining all steering force vectors with the velocity vector from the previous tick, a new velocity vector is computed and passed to the locomotion layer.

As a means of combining the force vectors of the steerings, we chose a weighted average of all nonzero steering vectors (with the previous velocity vector). The weight of the previous velocity vector can influence smoothness vs. prompt reactions ratio (the higher the weight is, the smoother the motion is, but reactions may be slower).

In our application, we have implemented the following six steerings: Target Approaching, Obstacle Avoidance, Path Following, Leader Following, Wall Following, and People Avoidance. The first five are based on [15] while the last one uses a similar approach as in [5]. The LF steering allows for setting the agent's relative position to the leader, which is our innovation to Reyonolds' version of this steering. All steerings are detailed in [13]. The seventh steering, detailed in Section 3, is Walk Along. We have used UnrealEngine2Runtime as a 3D engine, our own 0.25 km^2 large virtual city as a virtual environment, and the Pogamut platform for developing AI of the virtual characters [3].

3 Walk along Steering Behavior

The goal of the Walk Along steering is to steer an agent in such a way that it will, together with another agent (a *partner*), reach a certain place. Both of them should walk alongside each other keeping a certain distance between them. They should be capable of catching up with their partner or slowing down so that their partner will reach them sooner, if necessary. As opposed to the Leader Following steering, no agent is in the leading position and both know their target.

The steering has six parameters:

1. *Partner* – the agent, with whom the steered agent should walk along to the target.
2. *Target* – the target location.
3. *Distance* – the ideal distance between partners. They will keep it if the environment does not make it impossible.
4. *Force Scale* – a scaling constant; scales the magnitude of the resulting force vector to units that can be interpreted in the given virtual environment (cm_{UE2R}, m_{UE2R},...).
5. *Give way* – a boolean parameter which helps to make sure that partners do not get too close to one another. It solves situations where one of the agents is between the other agent and the target location.
6. *Wait for Partner* – a boolean parameter with which a steered agent that is much closer to the target location than the partner waits for the partner instead of going towards it and then returning back together with it. This influences only certain kinds of topological arrangements of partners and their target.

The WA steering steers only one of the agents and does not need to communicate with the other agent except of knowing the joint target, and the position and the velocity vector of the partner. To achieve the right behavior, both partners have to be steered by the steering Walk Along with the same or similar parameters.

The resulting force of the steering has three components: a_T (attractive force to the target), a_p (attractive force to the partner) and r_p (repulsive force from the partner). The first handles reaching the target place and the other two keep correct distance between partners. At every moment, either a_p or r_p is zero. We now explain how these force components are calculated, starting with introducing the naming conventions.

The *Target* parameter is just one location and both partners can not stand on this location together. Therefore each partner will have its own location: T_m (my target) and T_p (the partner's target).

Figure 1 shows an outline of the situation: T stands for the *Target* parameter, L_m is the location between the partners, and *axis* is the straight line going through L_m and T. The locations T_m and T_p lie on the straight line going through T that is perpendicular to *axis*. The distance between T_m and T is the same as the distance between T_p and T and equals the half of the parameter *Distance*. T_m is the location nearer to the steered agent and T_p is the other location. We will further need three distances: d_p is the current distance between partners, D_p is the distance between the partner and its target T_p, and D_m is the distance between the steered agent and its target T_m.

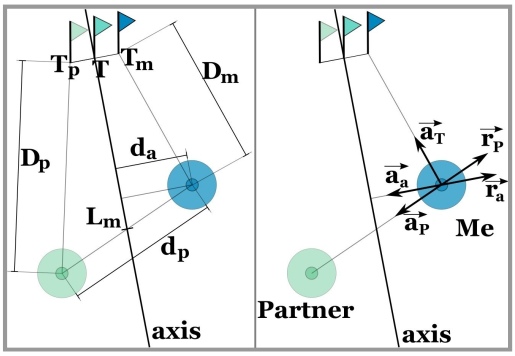

Fig. 1. A drawing of partners, their targets, distances (left) and forces (right) of the Walk Along steering for the darker agent

The first component of the resulting steering force is the vector a_T, the attractive force to the target. Its direction is from the location of the steered agent to the T_m location. Its magnitude is determined by the equation:

$$F \cdot 2^{\min(1.5,\ D_m - D_p)} + c, \tag{1}$$

where F is the value of *Force Scale*, and c is a small positive constant *iff* $D_m > D_p$, or 0 otherwise. The purpose of c is to increase the agent's speed when it is farther from the target than its partner (c is approximately $F \cdot 1/7$ in our implementation). Note that when the agents walk next to each other ($D_m = D_p$), a_T equals F. When they are not in a pair, the magnitude of a_T for the farther agent increases at least by $F+c$ but it is truncated at the upper bound $F\cdot 2^{1.5}+c$ (which corresponds to the maximal agent's speed; the value 1.5 can be changed when a different maximal speed is needed). At the same time, the magnitude decreases for the nearer agent to zero, which means that the agent will go increasingly slower (with increasingly higher distance between the agents). The lower speed of the nearer agent and the higher speed of the farther agent allow the farther one to catch up with the nearer one.

The other two components of the resulting steering force (a_p and r_p) are calculated according to the value of d_p, the actual distance between the partners. The direction of a_p, the attractive force to the partner, is from the location of the steered agent to its partner, and magnitude of a_p is determined by the equation:

$$F \cdot \frac{\max(0, d_p - D)}{D}, \tag{2}$$

where D is the value of the parameter *Distance* and the other parameters are as above. The higher the actual distance between the agents is, the higher the attractive force is. When the actual distance is the same as *Distance* or lower, the a_p's magnitude is zero.

The vector r_p leads to the steered agent away from its partner and the vector's magnitude is determined by the equation:

$$F \cdot \frac{\max(0, D - d_p)}{D}, \tag{3}$$

where the parameters are as above. This vector is nonzero *iff* the actual distance between the partners is lower than *Distance*. The lower the actual distance is, the higher the repulsive force is. Note the maximal magnitude of r_p equals to F.

Eventually all three steering components, a_T, a_p and r_p, are combined. If the magnitude of the resulting steering force exceeds a maximal value, it is truncated. Because the a_p component is the only one without its own upper bound, its contribution to the resulting steering force can be substantial relative to the other two components. This becomes useful when the IVAs are far apart from each other.

Due to these three force components, the partners are able to walk smoothly together to the target. If they are far apart from each other at the beginning of the simulation, they run to each other at first. Subsequently, they go to the target and try to keep the right distance between each other. If one of them gets stuck, the other slows down (if the first is not too far away), or returns back to the first one (if the first one is far away). If they get close to the target, the steering (in our implementation) returns the requirement for stopping the agent and if no other steering returns a nonzero force, the agent stops.

3.1 Advanced Walk along Behavior – The *Give Way* Parameter

The WA steering behavior described so far has one problem. When the agents are approaching each other, it may happen that they both end up at or very close to *axis*. Now, it will take some time until both partners form a pair perpendicular to *axis*, even if they have enough space around them. The reason is that r_p and a_T forces of the partner farther from the target have nearly opposite direction and they may nearly cancel each other out. Therefore, the farther partner slows down not to get too close to the agent closer to the target. However, the farther agent should have sped up to get next to its partner to form a pair. At the same time, the repulsive force of the agent closer to the target has the same direction as this agent's attractive force to the target. Therefore, the closer agent speeds up, but it should have slowed down.

The parameter *Give Way* solves this problem. Instead of detecting that partners are too close to each other, the steering uses d_a, the distance from *axis*. Instead of the force component r_p, we now use the force component r_a, the repulsive force from the *axis* (see Figure 1). The magnitude of the r_a force is determined by the equation:

$$F \cdot \frac{\max(0, D - 2 \cdot d_a)}{D}, \tag{4}$$

where F and D are as above. If d_a is higher than half of *Distance*, the force r_a is 0. Otherwise, the lower the actual distance is, the higher the repulsive force r_a is. When the partners form a couple staying perpendicular to *axis*, r_a equals to r_p from the basic WA variant. Otherwise, each partner moves to its side (away from *axis*) making a way for the second partner. Thus, both can get next to each other.

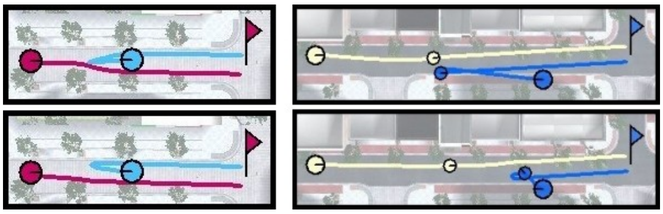

Fig. 2. The steering Walk Along without (top left) and with (bottom left) the parameter *Give Way*, and the WA steering without (top right) and with (bottom right) the parameter *Wait For Partner*. The heading of the agents is denoted by a short line, the flags denote the targets.

Figure 2 compares the same situation without (top left) and with (bottom left) the parameter *Give Way*. On the left top figure, the agents get too close to each other and it lasts too long until they form a pair staying perpendicular to *axis*. On the left bottom figure, they give way to each other and quickly get next to each other.

3.2 Advanced Walk along Behavior – The *Wait for Partner* Parameter

The steering Walk Along offers yet other functionality by means of the *Wait for Partner* parameter. It becomes useful in situations where one partner (Partner A) is approximately between the second partner (Partner B) and their target, and Partner B is quite far away. The basic steering behavior or the behavior with the *Give Way* parameter switched on steers Partner A to run to Partner B. When they meet, they continue together to the target, which means that Partner A, after a while of walking together with partner B, returns back to where it already was. This behavior may be natural in certain situations, e.g., when two young lovers run to each other to be with each other as soon as possible. However, the person closer to the target would often simply wait for the other person so that it does not have to walk the same route twice. With the parameter *Wait for Partner* switched on, Partner A is steered to wait until Partner B comes to it and only then they continue together to the target location.

We operationally define above described situation by two conditions: Firstly, the current distance between partners is higher than the *Distance* parameter. Secondly, the difference between D_p and D_m is higher than triple of *Distance* (the exact multiple can be changed). If both conditions are satisfied, the following calculation is used:

1. If d_a, the actual distance from *axis*, is higher than half of *Distance*, the sum of a_p and a_a forces is used instead of the force component a_p. The force a_a is the attractive force to *axis* and its magnitude is determined by the equation:

   $$F \cdot \frac{d_a}{D}, \tag{5}$$

   where the parameters are as above. The farther the steered agent is from the axis, the higher this force component is, which attracts the agent to *axis* and the partner.
2. If d_a is lower than or equal to *Distance*, the steering returns the requirement for stopping the steered agent and turning it to the partner.

Figure 2 compares a same situation with the *Wait for Partner* parameter switched off (top right) and on (bottom right). On the top, the agents run directly to each other. On the bottom, the darker agent just gets closer to the lighter one and waits there until the lighter one comes. Subsequently, they both continue together to the target.

4 Evaluation

We have tested the six steerings mentioned in Section 2 and the WA steering described in Section 3 in several dozen of scenarios in [13]. Moreover, to demonstrate usefulness of the WA steering, we have designed a formal evaluation study. Recall that the WA task can be also addressed by the extended Leader Following steering, where the follower walks next to the leader. The purpose of this study was to compare believability of the two solutions.

4.1 Method

We designed 8 different scenarios, which can be solved by both steerings. All scenarios have following common features: a) two virtual characters (friends) go together to the same place, b) they know about each other from the beginning and c) both of them know about the target place from the beginning. In some of the scenarios, characters meet at first and then continue together to the target.

We created videos[1] of the eight scenarios. Each video had two variants demonstrating how the LF or WA steering, respectively, solves the scenario. One scenario had

---

[1] All steering videos we have created and used are publicly available at: `http://artemis.ms.mff.cuni.cz/emohawk/doku.php?id=emohawk_steering_videos`.

four variants (see below). Twenty-six human subjects (20 males, 6 females, average age = 27, all except of four having a high previous experience with 3D VR applications) had to judge believability of every scenario using a six point Likert scale and add verbal comments on the believability of the videos. In particular, the subjects were introduced every scenario, stressing the point that the IVAs knew each other and should have walked along to the target, and should have judged to what extent the motion of IVAs appears believable (from 5 – "totally unbelievable" – to 0 – "this cannot be improved further"). The subjects were also asked to focus on the IVAs' trajectories, speed, heading, distance towards each other and pauses in movement, but not at the quality of animations. The order in which the two (four) variants of every scenario were showed to a viewer was randomized. The viewers did not know which steering was used in which variant. The scenarios are depicted on Figure 3 and listed below. Note that the last three were inspired by SteerBench [16].

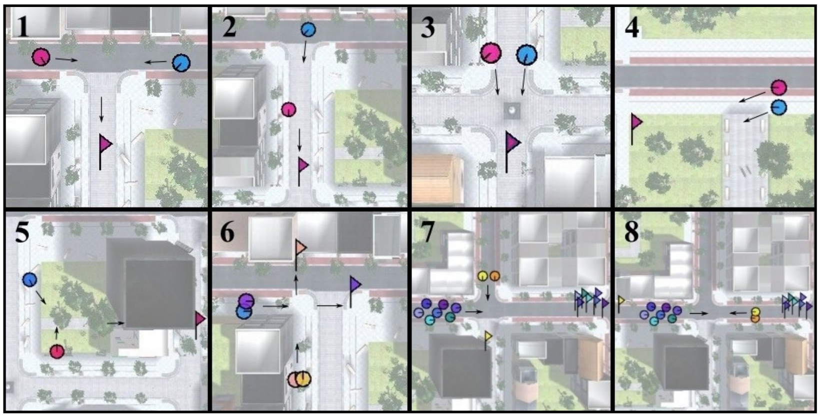

Fig. 3. The eight tested scenarios. The circles denote the starting locations of agents, the flags denote their targets, and arrows indicate direction of the agents' trajectories

1. Two friends meet at a crossway and continue together on the street.
2. Two friends meet on a street and continue together on the street. This scenario had four variants: WA with the *Wait for Partner* parameter switched (i) on and (ii) off, and LF where the leader is (i) the agent nearer to the target and (ii) the agent farer from the target.
3. Two friends walk together on the street and avoid an obstacle (a spherical statue).
4. Two friends walk together in a park and one of them suddenly stops. In the tested scenario, he gets stuck in front of a low bench, which is intentionally so low that agent's rays do not detect it. The viewers were instructed to imagine, that the stuck agent, e.g., laces its shoes, and judge the believability of the other agent's behavior.
5. Two friends meet in a park and walk together through a passage to a street.
6. Two friends go on a pavement along a building. A different pair of friends goes from the other side of the building and both pairs meet at the corner of the building. The pairs see each other at the very last moment. They should avoid all collisions and continue in the previous direction.

7. Two friends have to avoid collisions with the crossing group of people.
8. Two friends have to avoid collisions with the oncoming group of people.

4.2 Results

The scores of every two variants were compared using a paired T-test since the data were not grossly non-normal. For the second scenario, we compared the two WA variants against each other, and the better WA variant (i.e., with the lower mean of score) to the better LF steering variant. The results are showed in Table 1.

Table 1. Results of the Walk Along steering evaluation. The scores are scaled to 0 (the best) – 1 (the worst). The effect sizes were calculated using classical Cohen's *d*, where the classification is negligible (<0.2), small (<0.5), medium (<0.8), and large (over 0.8). In scenario 2, LF ii scored worse than LF i, thus, we used the latter for the comparison.

Scenario	Variant	Mean	STD	P-value	Cohen's *d*
1	WA	0.33	0.25	0.088 +	0.48 small
	LF	0.45	0.26		
2	WA i	0.25	0.21	0.0022 **	0.73 medium
	WA ii	0.42	0.25		
2	WA i	0.25	0.21	< 0.001 ***	1.10 large
	LF i	0.52	0.32		
3	WA	0.15	0.21	0.2 -	0.28 small
	LF	0.22	0.23		
4	WA	0.42	0.27	0.0032 **	0.91 large
	LF	0.68	0.31		
5	WA	0.20	0.20	< 0.001 ***	1.66 large
	LF	0.58	0.25		
6	WA	0.12	0.14	< 0.001 ***	4.57 large
	LF	0.85	0.18		
7	WA	0.53	0.30	0.95 -	-0.01 negligible
	LF	0.52	0.30		
8	WA	0.20	0.28	0.32 -	0.25 small
	LF	0.27	0.27		

We see that in the scenarios 2, 4, 5, and 6 the WA solution significantly outperformed the LF solution, and in the scenario 1, the difference is not significant but there is a clear trend favoring the WA steering. Arguably, concerning scenarios 1, 2, 4, and 5, this is because the partners wait on each other, or a partner basically reflects the existence of the other agent (see the supplementary videos). Additionally, concerning 5 and 6, the ignorance of the follower by the leader in the LF approach leads to severely limiting the movement possibilities of the follower.

On the other hand, there are no significant differences in scenarios 3, 7, and 8. This could be caused by the social aspects of the WA steering (waiting on the partner, reflecting its existence) not being important for solving situations 3, 7 and 8.

In general, the scores of both steerings suggest that the steerings are not perfect for most of the situations. Thus, the subjects' comments on the believability limitations are important. We now list the most recurring complaints for both steerings:

1. When the agent nearer to the target waits for the farther partner, or goes towards it, this nearer agent then rotates in the unnatural direction when the partner arrives and they both continue to the target. In particular, it rotates from the partner instead of to the partner. (Scenarios 2 and 4 – see the supplementary video).
2. Some respondents said that it is unnatural that the agents run to each other at first, but later they just walk. Few respondents noted that it could look naturally if the characters represented a pair of lovers who had not seen each other for a long time. But then they would have to walk very close to each other. (Scenarios 1, 2 and 5.)
3. It would look nice, if the agents stopped for a while and exchanged greetings etc. when they met. (Scenarios 1, 2 and 5.)
4. Partners could both go around an obstacle from the same side. (Scenario 3.)
5. Partners could both go around a group of other characters from the same side. (Scenario 8).
6. In scenario 2, when the parameter *Wait for Partner* is switched on, the waiting agent waits almost until its partner comes to it but it makes several steps to the partner just shortly before this moment and then it turns and continues with the partner. Some respondents emphasized this feature as a very natural detail whereas others regarded it more natural to wait until the partner comes.

4.3 Discussion and Possible Improvements

In general, the results demonstrated that the WA steering is useful. However, further improvements would add to its believability. Possible improvements can be divided intro three groups: a) at the level of the WA steering, b) at the action selection layer during combining the force vectors of individual steerings, c) by implementing a new steering. Concerning Group (a), we have already made three improvements[1] based on the results of our evaluation. In particular, we fixed:

- Unnatural direction of turning (Point 1 above). Briefly, we added yet another force vector to the vectors a_p, a_a, r_p, r_a, and a_t (the solution can not be detailed here due to the space constraints).
- Running towards each other (Point 2 above). We added a parameter for truncating the maximal velocity. Consequently, the partners do not run to each other, even if they are far away from each other.
- Waiting until the partner comes (Point 6 above). We implemented optional behavior for the *Wait for Partner* parameter that the agent waits until its partner really comes to it and then it just turns in place and continues with the partner.

Points 3, 4 and 5 belong to Group (b). In our opinion, these issues can be solved relatively easily at the action selection layer. In general, at this layer, additional behavior may be implemented by changing the WA steering parameters in runtime. For instance, various social relations between the partners may be expressed by setting *Distance* and *Wait for Partner* parameters (switching *Wait for Partner* on may be suitable for general use, but it may be better to switch it off for small children or a pair in love). Another possibility is combining WA with a different steering, for instance, for walking in a pair along a street on a pavement.

WA is currently able to steer a pair of agents. A future extension could include the ability to steer small groups of three or four people. Finally, note that the described WA steering is intended for open spaces. For closed space with narrow corridors and many obstacles, more changes of the WA steering, or even a new steering, would be necessary, which represents the Group (c).

5 Conclusion

In this paper, we have presented the Walk Along steering, which steers a pair of agents to reach a certain place together, ideally walking alongside each other. The steering is used to control their movement, not their animations. We have conducted an evaluation of believability of the WA steering and compared it to an extended version of the Leader Following steering. The evaluation has shown that the WA steering is useful and may lead to more believable behavior than the extended LF steering. In particular, the WA outperforms (in terms of believability) the LF solution in situations where a social behavior is expected by a human observer and/or the LF approach limits the movement possibilities of the follower. On the other hand, there are negligible or small differences between the solutions in situations in which the social aspects of the WA steering are not important. We have also already solved the most problematic issues related to the WA about which the respondents complained.

Our method keeps the advantage of Reynolds' steerings: it is simple and computationally inexpensive. In our opinion, the WA steering may be used in applications where a pair of virtual agents walking together to a common target is needed, such as computer games, educational applications or human crowd simulation, where pairs of virtual agents could contribute to natural impression of the crowd. Finally, the WA steering shows that steerings may not only control low-level navigation, but they may also be used to express social relations between agents.

Acknowledgments. This research was partially supported by project P103/10/1287 (GACR), by a student grant GA UK No. 0449/2010/A-INF/MFF, by SVV project number 263 314 and the research project MSM0021620838 of the Ministry of Education of the Czech Republic.

References

1. Bída, M., Brom, C., Popelová, M.: To Date or Not to Date? A Minimalist Affect-Modulated Control Architecture for Dating Virtual Characters. In: Vilhjálmsson, H.H., Kopp, S., Marsella, S., Thórisson, K.R. (eds.) IVA 2011. LNCS, vol. 6895, pp. 419–425. Springer, Heidelberg (2011) (accepted, to appear)
2. Ho, C.S., Nguyen, Q.H., Ong, Y.-S., Chen, X.: Autonomous Multi-agents in Flexible Flock Formation. In: Boulic, R., Chrysanthou, Y., Komura, T. (eds.) MIG 2010. LNCS, vol. 6459, pp. 375–385. Springer, Heidelberg (2010)
3. Gemrot, J., Kadlec, R., Bída, M., Burkert, O., Píbil, R., Havlíček, J., Zemčák, L., Šimlovič, J., Vansa, R., Štolba, M., Plch, T., Brom, C.: Pogamut 3 Can Assist Developers in Building AI (Not Only) for Their Videogame Agents. In: Dignum, F., Bradshaw, J., Silverman, B., van Doesburg, W. (eds.) Agents for Games and Simulations. LNCS, vol. 5920, pp. 1–15. Springer, Heidelberg (2009), `http://pogamut.cuni.cz` (6.7.2011)

4. Guy, S.J., Lin, M.C., Manocha, D.: Modeling collision avoidance behavior for virtual humans. In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Richland: International Foundation for Autonomous Agents and Multiagent Systems, vol. 2, pp. 575–582 (2010)
5. Karamouzas, I., Heil, P., van Beek, P., Overmars, M.H.: A Predictive Collision Avoidance Model for Pedestrian Simulation. In: Egges, A., Geraerts, R., Overmars, M. (eds.) MIG 2009. LNCS, vol. 5884, pp. 41–52. Springer, Heidelberg (2009)
6. Karamouzas, I., Overmars, M.: A Velocity-Based Approach for Simulating Human Collision Avoidance. In: Safonova, A. (ed.) IVA 2010. LNCS, vol. 6356, pp. 180–186. Springer, Heidelberg (2010)
7. Karamouzas, I., Overmars, M.: Simulating the Local Behaviour of Small Pedestrian Groups. In: Proc. Of the 17th VRST, pp. 183–190 (2011)
8. Olfati-Saber, R.: Flocking for multi-agent dynamic systems: Algorithms and theory. IEEE Transactions on Automatic Control 51(3), 401–420 (2006)
9. Pedica, C., Vilhjálmsson, H.H.: Social Perception and Steering for Online Avatars. In: Prendinger, H., Lester, J.C., Ishizuka, M. (eds.) IVA 2008. LNCS (LNAI), vol. 5208, pp. 104–116. Springer, Heidelberg (2008)
10. Pedica, C., Vilhjálmsson, H.: Spontaneous Avatar Behavior for Human Territoriality. In: Ruttkay, Z., Kipp, M., Nijholt, A., Vilhjálmsson, H.H. (eds.) IVA 2009. LNCS (LNAI), vol. 5773, pp. 344–357. Springer, Heidelberg (2009)
11. Pedica, C., Högni Vilhjálmsson, H., Lárusdóttir, M.: Avatars in Conversation: The Importance of Simulating Territorial Behavior. In: Safonova, A. (ed.) IVA 2010. LNCS, vol. 6356, pp. 336–342. Springer, Heidelberg (2010)
12. Peters, C., Ennis, C.: Modeling Groups of Plausible Virtual Pedestrians. In: IEEE CGA, pp. 54–63 (2009)
13. Popelova, M.: Knihovna steering technik pro virtualni agenty. Bachelor thesis. Charles University in Prague. (2011) (in Czech), `http://amis.mff.cuni.cz/emohawk/` (6.7.2011)
14. Reynolds, C.: Flocks, herds and schools: A distributed behavioral model. In: Proceedings of the 14th Conference on Computer Graphics and Interactive Techniques, pp. 25–34 (1987)
15. Reynolds, C.: Steering behaviors for autonomous characters. In: GDC, pp. 763–782 (1999)
16. Singh, S., Naik, M., Kapadia, M., Faloutsos, P., Reinman, G.: Watch Out! A Framework for Evaluating Steering Behaviors. In: Egges, A., Kamphuis, A., Overmars, M. (eds.) MIG 2008. LNCS, vol. 5277, pp. 200–209. Springer, Heidelberg (2008)