Int j simul model 15 (2016) 4, 585-596
ISSN 1726-4529 Original scientific paper

MODELLING GOAL SELECTION OF CHARACTERS IN PRIMARY GROUPS IN CROWD SIMULATIONS

Jayalath, D. D. A. C. J.*; Wimalaratne, S. P. W.* & Karunananda, A. S.**
* University of Colombo, School of Computing, No. 35, Reid Avenue, Colombo 0007, Sri Lanka
** Department of Computational Mathematics, University of Moratuwa, 10400, Moratuwa, Sri Lanka
E-Mail: pog666@gmail.com, spw@ucsc.cmb.ac.lk, askarunananda@gmail.com

Abstract
This work presents an agent based behavioural model that simulates decision making ability of individual characters of primary social groups in crowd simulations. The proposed approach includes a decision function which is used by each character in order to select the most appropriate goal from a list of predefined goals. This decision function selects a goal for an agent by evaluating attractions to goals, distance to goals, decisions of fellow companions in the agent's group, and attractions towards such companions. The behavioural realism of this model was evaluated through a series of experiments that compared parameters of real world scenarios with their simulated counterparts. The level of realism at which the model simulates characters is suitable for crowd simulations in entertainment related applications such as video games and movies. Further, the profiling of the algorithm shows that the approach is capable of being deployed in real-time applications such as video games.
(Received in March 2015, accepted in May 2016. This paper was with the authors 7 months for 1 revision.)

Key Words: Artificial Intelligence, Multi-Agent Systems, Virtual Reality, Crowd Simulation, Primary Groups, Social Groups

1. INTRODUCTION

Crowd simulation systems bring realism to applications such as movies and games [1, 2] while it is also used for solving serious problems such as floor planning [3]. Realism of a crowd simulation depends on several factors such as visual realism of characters and behavioural characteristics. Much research is conducted in order to improve visual realism and immersion of crowd behaviours within simulated environments [4, 5] and for formally specifying real world crowd behaviours [6, 7]. While realistic behavioural models are used in serious applications such as urban planning [3], simplified behavioural models are used in simulations such as games and movies [1, 2].

Groups of humans such as friends and colleagues are frequently found within real world crowds. They are referred to as Social groups in the domain of sociology. Social groups are categorised into two types as *primary groups* and *secondary groups* by the Cooley's classification [8]. Each person of a primary group interacts with all other persons in the group directly. Family, friends and colleagues are some examples for primary groups. Each person in a secondary group doesn't need to interact with all other persons but, all members share a specialty or interested in a common thing. Asians, Teachers and Americans are some examples for secondary groups. Despite their strong relationship with each other, members of a primary group may have different interests and different goals [9]. Moreover, strength of relationship among companions may differ even within the same primary group [10]. Despite these differences and similarities among members of a primary group, behaviour of a person is greatly affected by his/her primary group companions [8]. Therefore, consideration of primary social groups is of great importance to crowd simulations.

Much research has been done on simulating groups and their collaborative behaviours in virtual crowds. Nevertheless, most of those researches are based on groups of crowd characters that have the same goals [11]. Therefore, a lack of research in simulating groups of characters with different goals was observed.

This research was focused on simulating behaviours of groups of characters who have different goals. Although social groups could be of diverse types, the research focuses only on primary groups due to their aforementioned importance. The primary application of the research is real-time simulations such as computer games. The scope was kept within the tactical level of decision making [12] of virtual characters. Therefore, pre-known groups of characters with pre-known goals were only considered. Interactions between characters of separate groups were not considered.

Scalar values could be used for modelling dynamic interpersonal relationship strengths, as used by Bècheiraz & Thalmann [13] for simulating non-verbal communication of humans. They proposed of using a normalised value of fixed range from zero to one, in which zero and one represent hostile attitude and friendly attitude respectively. Therefore, the strength of attraction towards an individual could be modelled as a scalar value which depicts the level of attraction according to the value.

Action points and interest points were introduced for modelling path nodes and goal locations by Musse et al. [11, 14, 15]. These points allow agents to walk and interact with the environment. While interest points help in path planning and collision avoidance, action points store further information such as preferred orientation at the location and information about the actions that are to be taken at the goal location. The mentioned series of research were conducted on simulating groups of characters who have only common goals (i.e. a shared goal list). Therefore, all characters in a group always walk together as a flock. Further, these characters were modelled as rule based agents.

Later research on crowd simulations started using utility based agents [4] and decision networks [16] for achieving more realistic results. The characters were able to generate goals according to the mental state and use utility based decision making to decide upon selection of goals. Moreover, decision networks allowed modelling how behaviours of one agent affect another agent. However, communication between agents was not utilized for any decision making.

Cellular automata had been used on simulating group motions of characters by Dadova [17]. The method involved in counting the number of activated neighbouring cells of a character for determining its next move. This method was found to be hard to use due to difficulties in adjusting a cellular automata.

Multi agent communication has been used for distribution of knowledge for simulating evacuation behaviours by Pelechano et al. [18, 19]. While each agent was assigned a specific role (trained personnel, leader or follower), and the communication between characters of different roles helps all characters to evacuate the danger zone.

The related works described above were related to the problem of simulating groups of characters although none of them has specifically addressed the problem of having different goals for each character. However, the techniques used in those work are adapted to this research as needed.

In summary, this research attempts to address the lack of adequate research on simulating groups of characters with different goals. As a result, it contributes an agent based approach which could simulate decisions of groups of characters with approximate realism and adequate computational efficiency. Moreover, it introduces a novel function for deciding the next goal from a set of remaining goals of a crowd character.

2. METHOD

Primary groups are considerably small compared to secondary groups [8, 10], and pedestrians tend to walk in groups of six or less, for most of (more than 80 % of) the time [6]. Therefore, a microscopic approach is most preferred for simulating such groups [18].

The problem addressed by the research requires representing different attributes and individual goals for different characters. Further, it is required that the characters communicate with each other in-order to model their strong connections with each other as primary social groups. Cellular Automata and Agent based modelling are the two major categories of microscopic approach. While cellular automata allow local interaction with the environment, it doesn't allow character entities to communicate from a distance [18]. Nevertheless, agent based approaches not only allow characters to have different attributes and individual goals, but also it allows characters to communicate using various methods [18]. Further, agent based approaches could be easily executed in parallel for achieving greater performance. Therefore, agent based modelling is selected as the optimal approach for the solution.

Communication among agents could be done using message passing (unicast/multicast/broadcast) or a blackboard system. Message passing requires the agents to be listening at the receiving end while a message is being sent, in-order to receive the message. Therefore, message passing is a synchronized communication approach which requires both parties to be ready at the same time for communication, thus adds synchronization overhead to the communication procedures. On the contrary, blackboard systems utilize shared memory for communication among agents. This allows asynchronous communication by letting agents to read any required data from the blackboard only when they need. Therefore, due to the communication overhead of message passing, blackboard systems are more efficient than message passing. Considering the entire above, blackboard approach is selected as the optimal communication method for this solution since, computational efficiency matters for real-time crowd simulations.

Each group of agent characters (i.e. primary group) is assigned a blackboard. Each agent is able to read or write-to the blackboard. The relationship between a pair of characters is considered asymmetric. The strength of attraction towards a character is modelled as a numerical value. Therefore, each character in a group maintains a set of numerical values which represents attractions towards other characters in the group. An example is shown in Fig. 1, which shows how a group of three characters is formed.

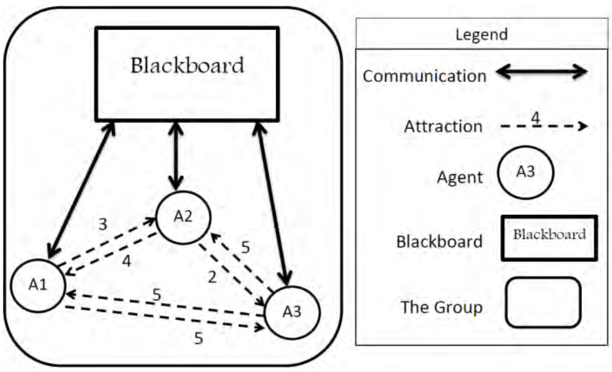

Figure 1: Group of three characters.

Further, it is allowed for each character to have any number of different goals. A goal is a location of interest for the character such as an observable artefact in a museum. For each goal, a utility is considered. Utility of a goal is a numerical value which represents the motivation of the character towards achieving the goal.

2.1 Functional flow of agents

Each agent keeps executing the main loop, which is illustrated in Fig. 2. Execution of this main loop is started immediately at the beginning of the simulation and continues until the loop is exited. At the beginning the agent checks whether there is any goal to be achieved. If there is at least one such goal, then control flows to decide step. At this step, the agent selects a goal from the remaining goals set and names it as the current goal. Then in the next step, the agent broadcasts the selected current goal to the blackboard of its group. After that in the next two steps, the agent moves towards its current goal and once reached interacts with the goal as needed. (E.g. take an item from a stall.) Once the current goal is interacted with, it is marked as achieved. Then the agent comes to the beginning of the loop and continues executing the loop until it goes to exit step.

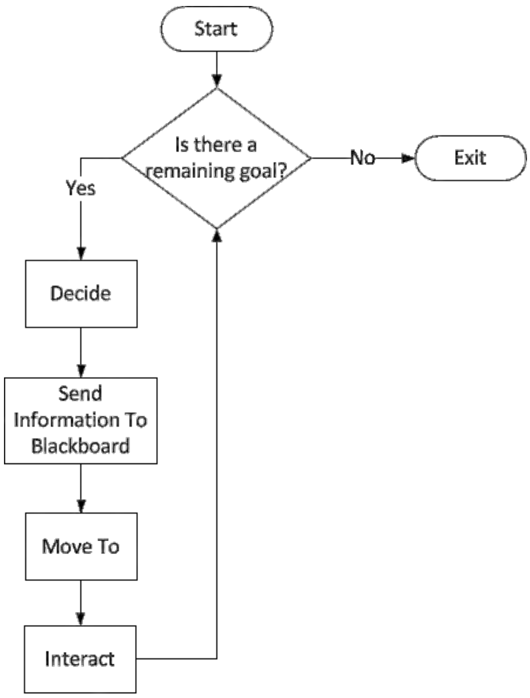

Figure 2: Process flow.

2.2 Decision function

Most important part of the described main loop is the decide step. Realism of this model largely depends on this step.

At this step, the agent calculates a value called *Total Utility* for each remaining goal. *Total Utility* of a goal is dependent on three values: utility of the goal, sum of attractions towards companions who have it as the current goal, and distance towards the goal. *Total Utility* of a goal g for character i (denoted as $T_i(g)$) is calculated as given by the equation below:

$$T_i(g) = K_G^i . U_i(g) + K_A^i . \sum_{c \in H_g} A_i(c) - K_D^i . D_i(g) \quad (1)$$

For a character i, the utility of a goal g, the attraction towards another character c, and the distance from the character to a goal g are given by functions $U_i(g)$, $A_i(c)$ and $D_i(g)$ respectively. The set of companions who has g as the current goal is denoted by H_g. Multipliers denoted by K_G^i , K_A^i , and K_D^i are constants which are attributes of each agent. A goal which gives the maximum *Total Utility* is selected as the current goal. The set H_g is found by reading the blackboard.

3. IMPLEMENTATION

A prototype of the system was implemented for evaluating the model. Unreal Development Kit® (UDK) was used for implementing the prototype.

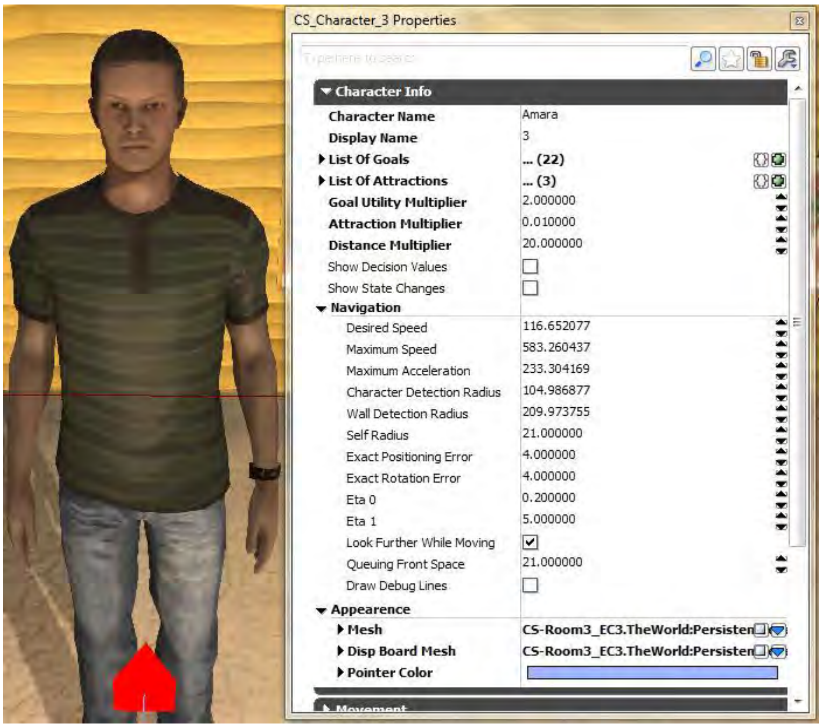

Figure 3: Configuration of Character Properties.

Figure 4: Simulated Environment with Blackboard on top left.

A screenshot of simulation engine with character properties window is shown in Fig. 3, where the character is configured. A screenshot of a running simulation is shown in Fig. 4.

A-star algorithm was used for path finding. Euclidean distance was used as the heuristic function for A-star algorithm. Distance calculations in decide step was also done by path finding with the A-star algorithm and calculating distance along the found path. Distances to remaining goals were normalized before calculating *Total Utility* values of goals at each decide step. The algorithm used for implementing the decide step is shown in Fig. 5.

```
CurrentGoal ← Φ // nothing
G ← ∅ // empty set
NormOfDists ← 0
For each remaining Goal g Do:
│ dist ← CalculateDistanceTo(g)
│ NormOfDists ← NormOfDists + dist²
│ G ← G ∪ {< g, dist >}
NormOfDists ← √NormOfDists
MaxUtility ← −∞
For each < goal, distance > pair in G Do:
│ distance ← distance ÷ NormOfDists
│ utility ← GetUtiliity(g)
│ attraction ← 0
│ For each Companion C Do:
│ │ If Current Goal of C is goal Then:
│ │ │ attraction ← attraction
│ │ │                 +GetAttraction(C)
│ TotalUtility ← K_G × utility + K_A × attraction
│                 −K_D × distance
│ If MaxUtility < TotalUtility Then:
│ │ MaxUtility ← TotalUtility
│ │ CurrentGoal ← goal
return CurrentGoal
```

Figure 5: Decision algorithm.

An array of character vs. goal pairs was used for implementing the blackboard of each group. Navigation component was written by adapting the navigation model suggested by Chraibi et al. [20].

4. EVALUATION

The prototype was used for evaluating the realism and efficiency of the proposed model.

4.1 Behavioural realism

Behavioural realism of the model was evaluated by comparing decisions of subjects (real world characters) with their simulated counterparts. The Museum of Colombo was used as the venue for conducting the experiments. Three rooms were selected from the museum and modelled in the prototype simulator. A total of forty-one volunteered human subjects were taken for the experiment. Twenty three groups of characters were created using these subjects. Each group consists of four characters. The ages of subjects were between 18 to 25 years where all of them were Sri Lankans. Only three subjects out of the forty-one subjects were female.

At each trial of the experiment, a group was taken into a selected room in the museum and they were allowed to observe the artefacts of the museum briefly enough to get an idea about the room. A map of the room was also given to each of them. Then the group of subjects were taken outside and were asked to list down the artefacts which they would like to observe in their data sheets. Observing these artefacts was to be their goals when they enter the room to observe them. Further, they were asked to rate each artefact with a numerical value (ranging from 0 to 10) to indicate the utility of each goal. They were also asked to write the attraction towards each companion in the group using a numerical value similarly. After collecting these data, they were allowed to enter the room through the same door as a group and observe the artefacts which they marked. While they are observing the artefacts, a separate group of observers was used to observe each one of them and write down the order which they visited their goals.

Data collected from the subjects (goals, utilities and attractions) were used to model each subject as a virtual character (i.e. an agent) and simulate the same scenario in the simulator. For each character, the multiplier values K_G^i, K_A^i, and K_D^i were determined experimentally. The order which virtual characters visited their goals was recorded. This ends one trial. These trials were done with all 23 groups, one at a time.

At the end of each trial, two orders were retrieved: real order and virtual order. These orders (i.e. sequences) are permutations of each other. Real order is the order which real character visited his goals. Virtual order is the order which virtual character visited the same goals. The two orders will be similar only if the proposed approach is realistic. Levenshtein distance and Median displacement were taken as measures for calculating degree of similarity. Definition of median displacement is given below:

> *Median displacement:* Measured by calculating the absolute displacement of each element in a pair of sequences and taking the median value of those values. (For each element, find the two indexes at which the element occurs in both sequences and calculate the difference. Afterwards, calculate the median of those difference values.)

Both measures (Levenshtein distance and Median displacement) become zero when the compared orders are the same and increase when the compared orders are different. The number of mismatching goals in a pair of orders is depicted by Levenshtein distance. Median displacement shows the distance between occurrences of the same goal, in a pair of orders.

Since there were no similar models to compare with, the approach was compared with a model which had a randomizing decision function (i.e. a model which selects a random goal from the remaining set of goals as the current goal.). Simulating a subject using this randomizing model yields a random order of same length as the output. Therefore, the output of simulating a subject could be retrieved from a random sequence generator. The similarity between random order and real order was measured using two measures described above.

To make the randomization fairer, a set of random sequences were generated for each subject and the median values of measurements were calculated as the measurements for similarity between random order and real order of the subject. Virtual orders and random orders are compared with their real order to calculate the measurement pairs (i.e. measurements for random-vs.-real and virtual-vs.-real orders) for each subject. A total of 92 measurement pairs were retrieved from the twenty-three groups, since each group had four characters.

The quartiles of the distributions of Levenshtein distances and median displacements are given in Table I. These distributions show that the median (Q_2) Levenshtein distances of virtual-vs.-real orders are lower than the random-vs.-real orders. Same is true for the distributions of median displacements. Further it is evident that the inter quartile range (Q_3-Q_1) of distributions of virtual-vs.-real orders are less than or equal to their random-vs.-real counterparts. This suggests that the proposed model gives more realistic results than the random algorithm. However, these distributions were further analysed for drawing better conclusions.

Goodness of fit tests attested that distributions of both Levenshtein distances and median displacements are not normal. Therefore, a Wilcoxon signed rank test was performed for both Levenshtein distances and Median displacements as a hypothesis test. The tested alternative hypotheses were that distance/displacement between real and virtual orders are less than distance/displacement between real and random orders. The results of the tests rejected the null hypothesis at 0.01 level of significance, for both Levenshtein distance and Median displacement. Therefore, the data provides sufficient evidence to conclude that, the virtual orders given by the proposed solution are more similar to real orders than the random orders.

Table I: Virtual-vs.-Real order and Random-vs.-Real order measurements.

N = 92		Min	Q_1	Q_2	Q_3	Max	Q_3-Q_1
Levenshtein distances	**Virtual-vs.-Real**	0	5	8	11	19	6
	Random-vs.-Real	1	7	10	14.25	23	7.25
Median displacement	**Virtual-vs.-Real**	0	1	2	3	5	2
	Random-vs.-Real	1	3	3.5	5	7.5	2

The simulated and the real order of goals of a group are illustrated in Fig. 6, in which paths of the four persons of the group are shown in four distinctive lines. Calculated values for the same group are shown in the Table II.

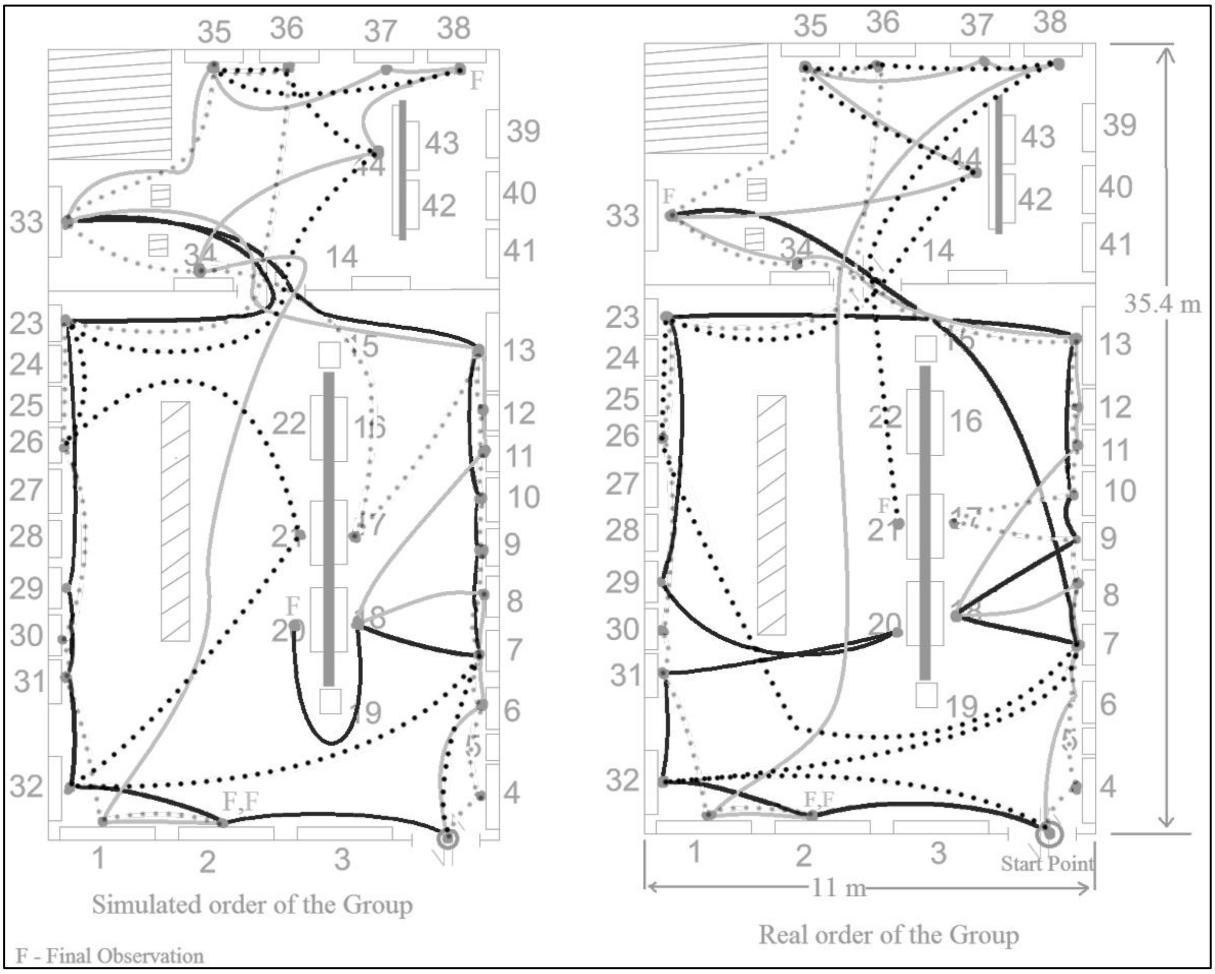

Figure 6: Paths with the order of goals for Simulated and Real World Human Groups.

Table II: Calculated measurements of the group illustrated in Fig. 6.

Person Index	Median displacement for proposed model	Median displacement for randomised model	Levenshtein distance for proposed model	Levenshtein distance for randomised model
0	0.5	3.5	5	10
1	0	5	2	16.5
2	1	3	5	7
3	0	4	4	13

4.2 Computational efficiency

Efficiency was measured against varying number of groups, number of characters per group, and number of goals per character. Performance (i.e. efficiency) is measured by using the update rate of the simulation. Update rate is the rate at which the virtual world evolves (It is not the same as drawing frame rate of the simulation). Update rate of a trial was measured by calculating the median of a series of 300 recorded update rate values (each update rate value is recorded once per 3 updates).

The machine used for the experiment had a dual core processor of type Intel® Pentium®, 2×2.9 GHz and 4GB of RAM. It was equipped with an NVIDIA® GeForce 8400 GS (512 MB) graphics unit with a 64-bit version of Windows® 7 operating system.

Update rate was measured against varying number of goals per character as shown in Fig. 7. According to the graph, the number of goals per character doesn't affect the performance by any significant amount.

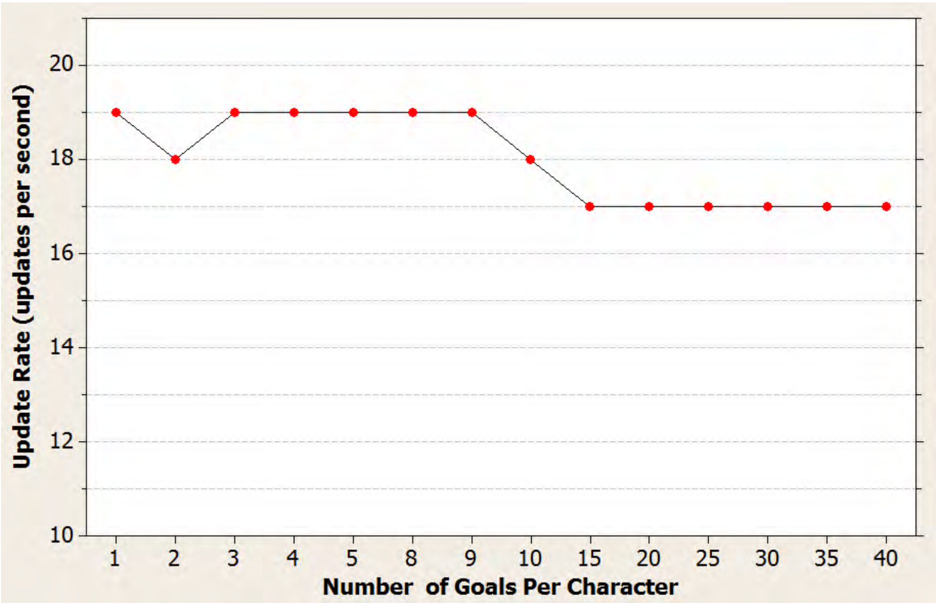

Figure 7: Update rate against Number of goals per character.

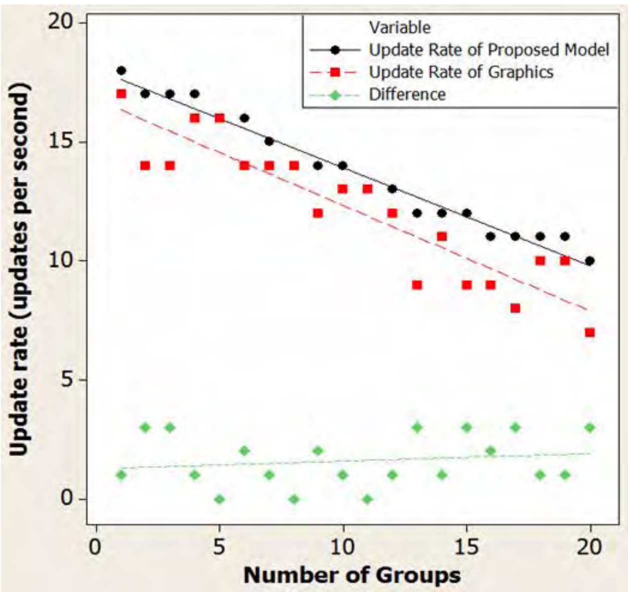

Figure 8: Update rate against Number of groups.

Update rate variation was measured against varying number of groups as illustrated in a scatter plot with regression lines in Fig. 8. The difference of update rates of control and experimental data is also drawn in the same graph. There is a slight grow in this difference as the number of groups is increased. Therefore, it could be concluded that the number of groups which use the behavioural model has an effect on the update rate compared to the effect caused by graphics and other necessary functions. Thus it could be concluded that the performance decreases when the number of groups is increased.

Update rate was measured against varying number of characters per group as shown in Fig. 9. Number of characters per group doesn't seem to be affecting the update rate according to the data since, both control and experimental graphs are giving approximately the same values Therefore it could be stated that, increasing number of persons per a group doesn't drop the performance of simulation.

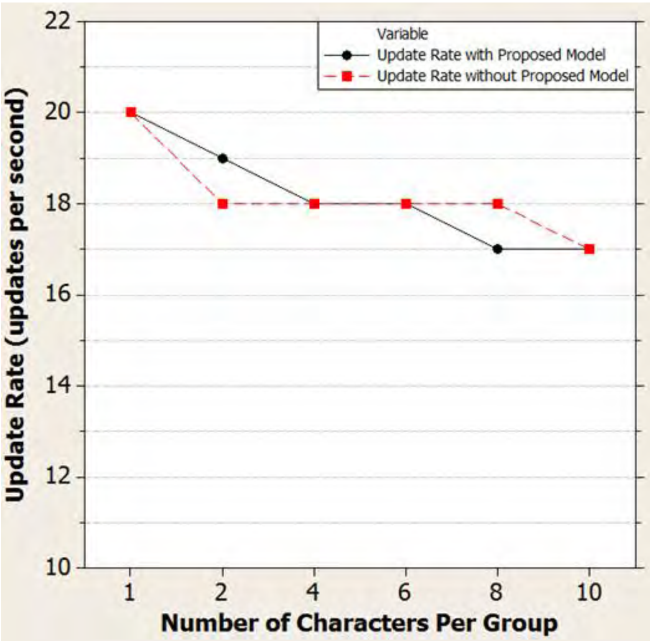

Figure 9: Update rate against Number of characters per group.

4.3 Discussion

Values for multipliers K_G^i, K_A^i, and K_D^i of each character are hard to be determined, while the rest of inputs such as goals, walking speed, and acceleration are obtainable by observation and measurement. Therefore, values for these multipliers were experimentally estimated. However, perfectly accurate input values are required in-order to simulate a given scenario accurately. Therefore, the expected outcome of the evaluation is an approximately similar simulation of the real world scenario.

Conclusion of hypothesis tests on evaluation of behavioural realism confirms that the proposed model is more realistic than a randomizing algorithm. Some of the values in Table I could be reinterpreted to provide further explanations as follows.

The median value of median displacement is 2 for virtual-vs.-real orders. Thus implies that whenever a real world scenario is modelled, if a goal pursued by a real person as his n^{th} goal will be pursued by the respective simulated character as its m^{th} goal, then the difference between n and m will be 2 in most cases. Therefore, it is evident that the virtual order of goals resembles the real order of goals with slight alternations, although it doesn’t perfectly match. The median value of Levenshtein distance is 8 for virtual-vs.-real orders, meaning that the number of mismatching decisions made by the simulated character is 8 at most of the time.

Even though there are considerably many mismatching decisions, as they are such, in most cases these mismatches are just slight alternations. Therefore, it could be concluded that even though it is still far from perfection, the proposed model simulates the given input scenarios with moderate similarity.

Further analysis on the gathered data suggests that some people are more driven by their interest while some are more attached to their companions. Further there are those who prefer to walk short distances. All these types of persona could be easily created by adjusting the K_G^i, K_A^i, and K_D^i multipliers. Therefore, the proposed model allows flexibility in defining different behaviours for characters. However the values of those multipliers are considered to be static in this model.

The evaluation of computational efficiency suggests that the computational cost of the proposed model is considerably low and it is mostly dependent on the number of primary groups in the simulation. Therefore the model could be easily used in real-time software.

5. CONCLUSION

As found in previous research, majority of pedestrians walk with companions. Therefore, simulating social groups is important for crowd simulations. Among social groups, primary social group is the category of groups that mostly affects the decisions of its member. In this research we have addressed the problem of simulating primary social groups in crowd simulations.

The proposed model could simulate groups of characters who have different personal goals with moderate realism. It was also observed that different personalities could be created by changing the multiplier values (K_G^i, K_A^i, and K_D^i). Furthermore, it was found that these values change not only from person to person, but also for the same person according to different environmental conditions. It was found that number of characters per group and number of goals per character doesn't affect the performance. Nevertheless, a decrease in performance is noticed when number of groups is increased.

The simplicity of the proposed model allows flexibility for it to be used in applications such as video games. Further, the computational efficiency of the model is suitable for simulating large crowds.

Further investigation in more complex decision functions could be performed using statistical and machine learning approaches. The proposed model could be adapted to consider negative attractions towards companions and certain locations of the map, and attractions towards characters that are not in the same group could also be considered. Moreover, lot of work could be done on simulating dynamic attractions towards goals, companions and other characters while decisions of a character could be used to alter these attractions of other characters. Such a system would bring further complexity into the simulation and would be curious to study.

ACKNOWLEDGEMENT

The authors would like to thank all the people who provided help in this research including volunteers in the experiment. Special thanks go to Dr. M. Maheesh, Mr. R. Abeygunawardana, and Mr. B. Chandrarathna.

REFERENCES

[1] Silverman, B. G.; Johns, M.; Cornwell, J.; O'Brien, K. (2006). Human behavior models for agents in simulators and games: Part I: Enabling science with PMFserv, *Presence: Teleoperators and Virtual Environments*, Vol. 15, No. 2, 139-162, doi:10.1162/pres.2006.15.2.139

[2] Davies, N. P.; Mehdi, Q. (2006). BDI for intelligent agents in computer games, *Proceedings of CGAMES'2006, 8th International Conference on Computer Games: Artificial Intelligence and Mobile Systems*, 104-107

[3] Smedresman, G. Crowd simulations and evolutionary algorithms in floor plan design, from *http://www.smedresmania.com/wp-content/uploads/2009/08/cs_ea_fpd_2006.pdf*, accessed on 01-01-2016

[4] Shao, W.; Terzopoulos, D. (2007). Autonomous pedestrians, *Graphical Models*, Vol. 69, No. 5-6, 246-274, doi:10.1016/j.gmod.2007.09.001

[5] Karamouzas, I.; Overmars, M. (2012). Simulating and evaluating the local behavior of small pedestrian groups, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, No. 3, 394-406, doi:10.1109/TVCG.2011.133

[6] Moussaid, M.; Perozo, N.; Garnier, S.; Helbing, D.; Theraulaz, G. (2010). The walking behaviour of pedestrian social groups and its impact on crowd dynamics, *PLoS ONE*, Vol. 5, No. 4, e10047, doi:10.1371/journal.pone.0010047

[7] Fridman, N.; Zilka, A.; Kaminka, G. A. (2011). *The impact of cultural differences on crowd dynamics in pedestrian and evacuation domains*, Technical Report MAVERICK 2011/01, The MAVERICK Group, Computer Science Department, Bar Ilan University, Ramat-Gan

[8] Abraham, M. F. (2014). Types of societies and groups, *Contemporary Sociology*, 2nd ed., Oxford University Press, India, 97-98

[9] Wegge, J.; Haslam, S. A. (2003). Group goal setting, social identity and self-categorization, Haslam, S. A.; van Knippenberg D.; Platow, M. J.; Ellemers, N. (Eds.), *Social identity at work: Developing theory for organizational practice*, Psychology Press, Hove, 43-59

[10] Rao, C. N. S. (1995). Social groups, *Sociology*, S. Chand & Co. Ltd., New Delhi, 282-294

[11] Musse, S. R.; Thalmann, D. (1997). A model of human crowd behavior: Group inter-relationship and collision detection analysis, *Computer Animation and Simulations '97, Proceedings of Eurographics Workshop*, 39-51

[12] Schadschneider, A.; Klingsch, W.; Klüpfel, H.; Kretz, T.; Rogsch, C.; Seyfried, A. (2009). Evacuation dynamics : Empirical results, modeling and applications, Meyers, R. A. (Ed.), *Encyclopedia of Complexity and System Science*, Springer, New York, 3142-3176

[13] Bècheiraz, P.; Thalmann, D. (1996). A model of nonverbal communication and interpersonal relationship between virtual actors, *Proceedings of the Computer Animation '96*, 58-67

[14] Musse, S. R.; Thalmann, D. (2001). Hierarchical model for real time simulation of virtual human crowds, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 7, No. 2, 152-164, doi:10.1109/2945.928167

[15] Musse, S. R.; Babski, C.; Capin, T.; Thalmann, D. (1998). Crowd modelling in collaborative virtual environments, *Proceedings of the ACM Symposium on Virtual reality software and technology (VRST '98)*, 115-123, doi:10.1145/293701.293716

[16] Yu, Q.; Terzopoulos, D. (2007). A decision network framework for the behavioral animation of virtual humans, *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 119-128

[17] Dadova, J. (2012). *Cellular Automata Approach for Crowd Simulation*, Ph.D. Dissertation, Faculty of Mathematics, Physics and Informatics, Comenius University, Bratislava, (http://www.sccg.sk/~dadova/phd/rigorozka_dadova_final.pdf)

[18] Pelechano, N.; Allbeck, J. M.; Badler, N. I. (2008). *Virtual Crowds: Methods, Simulation, and Control* (Synthesis Lectures on Computer Graphics and Animation), Morgan & Claypool Publishers, Williston, doi:10.2200/S00123ED1V01Y200808CGR008

[19] Pelechano, N.; Badler, N. I. (2006). Modeling crowd and trained leader behavior during building evacuation, *IEEE Computer Graphics and Applications*, Vol. 26, No. 6, 80-86, doi:10.1109/MCG.2006.133

[20] Chraibi, M.; Freialdenhoven, M.; Schadschneider, A.; Seyfried, A. (2013). Modeling the desired direction in a force-based model for pedestrian dynamics, Kozlov, V. V.; Buslaev, A. P.; Bugaev, A. S.; Yashina, M. V.; Schadschneider, A.; Schreckenberg, M. (Eds.), *Traffic and Granular Flow'11*, Springer-Verlag, Berlin, 263-275, doi:10.1007/978-3-642-39669-4_25