Crowd Simulation using Discrete Choice Model

Wenxi Liu* City University of Hong Kong
Rynson Lau† City University of Hong Kong
Dinesh Manocha‡ The University of North Carolina, Chapel Hill

ABSTRACT

We present a new algorithm to simulate a variety of crowd behaviors using the Discrete Choice Model (DCM). DCM has been widely studied in econometrics to examine and predict customers' or households' choices. Our DCM formulation can simulate virtual agents' goal selection and we highlight our algorithm by simulating heterogeneous crowd behaviors: evacuation, shopping, and rioting scenarios.

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Animations

1 INTRODUCTION

Crowd simulation has been regarded as an important research topic in computer graphics, virtual environments, and related areas. Various approaches have been proposed to simulate large, heterogeneous crowds and they are widely used on realtime applications. A simple approach used to simulate a variety of crowd behaviors is goal-driven crowd simulation [13] [17]. In this approach, goals could be time-dependent and may change dynamically in the environment. Given a goal position, the simulation algorithm uses global navigation and local collision avoidance techniques to compute velocity. But it is tedious to manually assign goals for hundreds of agents. Ideally, we need good techniques that automatically computes appropriate goal positions, and thereby generate realistic crowd behaviors.

We present a crowd simulation algorithm based on our novel goal selection mechanism. The contributions of our work include:

- We present an algorithm for goal-driven crowd simulation based on Discrete Choice Model, which provides proper goals for agents on the run.
- We present a dynamic feedback model that takes into account the positive and negative feedbacks from the environment during goal selection.
- Our formulation can be used to generate real-world scenarios. We highlight the evacuation, shopping and rioting scenarios.

Our DCM formulation has a number of advantages. It can be easily integrated with existing local collision avoidance schemes. It is simple to implement and has a small runtime overhead. It can also simulate crowd motions composing of hundreds of agents at interactive rates.

The rest of the paper is organized as follows. Section 2 summarizes existing works on goal selection. Section 3 presents our DCM formulation. Section 4 demonstrates how we model different crowd behaviors with our DCM formulation. Section 5 discusses and evaluates some experimental results.

*e-mail: wenxiliu2@cityu.edu.hk
†e-mail:rynson@cs.cityu.edu.hk
‡e-mail:dm@cs.unc.edu

IEEE Virtual Reality 2012
4-8 March, Orange County, CA, USA
978-1-4673-1246-2/12/$31.00 ©2012 IEEE

2 RELATED WORKS

There is extensive literature on crowd simulation and we refer the reader to some nice recent surveys [7] [14]. Here we survey on the works most related to goal selection. We classify them into two types: those considering geographical points or areas as goals and those defining goals as motor skill or actions.

As a representative work of the first type, Treuille et al. [17] consider goal selection as an external parameter set by the animator and a large number of agents need to share a few common goals. Many other works use a similar strategy [8] [10] [6].

The second type of works defines an *action* or a *behavior routine* as a goal. Most of them integrate individual personalities into the procedure of action selection, and construct a complete decision scheme for agents. They use a high-level decision scheme to direct agents' behaviors [11] [9] [19]. Terzopoulos et al. [12] simulate a school of fish and model their behaviors using mental states and a finite state machine. Funge et al. [1] propose a sophisticated cognitive framework to model agents' decision making by including environmental knowledge and logic. In all these works, a suitable decision making scheme needs to be designed specifically for each scenario.

3 OUR DCM FORMULATION

In Section 3.1, we first introduce our DCM formulation, which considers different factors in computing the utility values. In order for our DCM formulation to consider useful environmental factors during the decision making process, we introduce our dynamic feedback model in Section 3.2.

3.1 The Discrete Choice Model (DCM)

The discrete choice model (DCM) [16] is used to model choices of a decision maker from an alternative set. DCM have been used in the study and prediction of business decisions, including pricing and product development [16]. It has also been used to predict demands for a transportation system being planned [15] and in energy usage and environmental study [2].

There are different types of discrete choice models. The prominent types are *binomial choice models*, which deal with two alternatives, and *multinomial choice models*, which are able to select goals from more than two alternatives. For more details of these models, we refer the readers to [16]. Among different models, we apply the Logit model [4] in our work. Its major advantage is that the probability computation can be expressed as a closed-form formulation. This closed-form solution is relatively cheap to compute, making it possible to perform interactive crowd simulation on large crowds.

In our DCM formulation, when an agent (or a decision maker), a, faces a choice of *N potential goals* (i.e., N potential targets that a may choose from), a net benefit or utility, U_i, of each potential goal is computed. Agent a would choose the potential goal that provides the largest utility, i.e., potential goal i given $U_i > U_j, \forall j \neq i$, as his/her next goal. We model the utility of each goal by two components: an *observable component* and an *unobservable component*. Agent a could only observe some attributes of potential goal i, x_i, and some attributes of a itself, s_a, generally denoted as X. We denote the utility of all these observable attributes as a *representative utility*. For agent a, the representative utility of potential goal i is, $O_{i,a} = O(x_i, s_a) = \beta \cdot X$, where β is the set of corresponding weights to the set of observable attributes X. Hence, the utility of potential goal i to agent a can be expressed as:

$$U_{i,a} = O_{i,a} + \overline{O}_{i,a} \tag{1}$$

where $\overline{O}_{i,a}$ represents the unobservable component of i's utility.

It could be shown that the final formulation used in our framework is a closed-form expression of the probability as shown below [16]:

$$P_i = \frac{e^{O_i}}{\sum_j e^{O_j}} = \frac{e^{\beta X_i}}{\sum_j e^{\beta X_j}} \tag{2}$$

With this formulation, we may compute the probability of an agent selecting potential goal i as his/her goal. The set of attributes, X, and the set of corresponding weights, β, differ in different scenarios, which are discussed in following subsections.

3.2 The Dynamic Feedback Model

According to a cognitive study on crowd behaviors [5], self-organized behavior relies mainly on positive feedback loop and negative feedback loop. Let us consider shopping as an example, those shops with lots of customers always attract more shoppers to visit. As pointed out by Moussaid et al. [5], this kind of positive feedback may lead to “explosive amplification”, since the positive feedback runs recursively and leads to a non-linear propagation of the existing feedback. Hence, we suggest to model it as an exponential function:

$$f_{pos} = v_{pos} e^{k_{pos}(n/n_{max})} \tag{3}$$

where v_{pos} is a constant to scale the expression result within [0,1]. k is constant to calibrate the positive feedback. n is the current number of agents inside a potential goal, while n_{max} is the maximum number of agents that the potential goal can accommodate.

On the other hand, shoppers would be discouraged to enter a shop if it is crowded with too many people. This is a negative feedback and we may define it as follows:

$$f_{neg} = -v_{neg} e^{k_{neg}(1-\sqrt{n_{max}/n})} \tag{4}$$

where v_{neg} and k_{neg} are constants related to the degree of influence of the negative feedback.

Our dynamic feedback model is obtaining by integrating the effects of positive and negative feedbacks as follows and as shown in Figure 1:

$$\begin{aligned} f &= f_{pos} + f_{neg} && (5) \\ &= v_{pos} e^{k_{pos}(n/n_{max})} - v_{neg} e^{k_{neg}(1-\sqrt{n_{max}/n})} && (6) \end{aligned}$$

The dynamic feedback model of Eq. (6) can now be considered as a function of variable n, and applied to our DCM formulation of Eq. (2) simply by treating f as an attribute of vector X as defined in Section 3.1.

4 Modeling Goal Selection

In this section, we show how our DCM formulation with dynamic feedbacks can be used to simulate goal selection of different crowd behaviors, including evacuation, shopping and rioting.

4.1 Evacuation

When evacuating, people often have to make choices, such as selecting an exit to escape. Without loss of generality, we focus here on exit selection in evacuation. The representative utility of choosing an exit is determined by the following factors:

- $d_{i,a}$: Normalized distance between agent a and exit i.
- A_i: “Attractiveness” of exit i to model the blocking status of i.

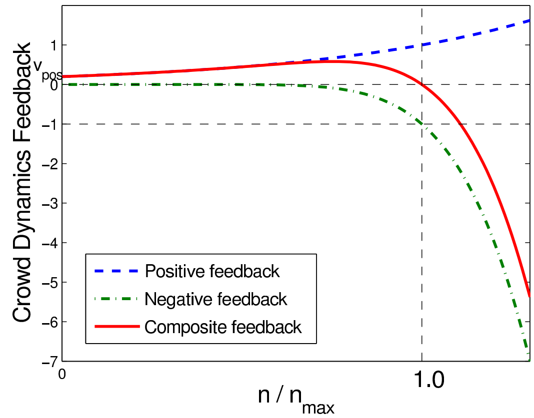

Figure 1: Positive and negative crowd dynamic feedbacks (Eq. 6).

We may incorporate these factors into our DCM formulation by forming the attributes of X_i in Eq. (2) as follows:

$$X_i = [\frac{1}{d_{i,a}} \qquad A_i] \tag{7}$$

Now, the probability of agent a choosing exit i becomes:

$$P_{i,a} = \frac{e^{\frac{\beta_0}{d_{i,a}} + \beta_1 A_i}}{\sum_j e^{\frac{\beta_0}{d_{j,a}} + \beta_1 A_j}} \tag{8}$$

where β_0 and β_1 are the weights of the first and the second attributes, respectively, in X_i. To model the attractiveness of exit i, we apply the dynamic feedback model of Eq. 6. We have:

$$A_i = v_{pos} e^{k_{pos}(n_i/n_{i,max})} - v_{neg} e^{k_{neg}(1-\sqrt{n_{i,max}/n_i})} \tag{9}$$

where n_i is the number of agents near exit i and $n_{i,max}$ is the number of agents that could block exit i.

4.2 Shopping

Shopping is a popular activity that we frequently engage in our daily life. A shopping trip scenario is often composed of a sequence of shop visits, which could be modeled by goal selection. The representative utility of each shop consists of the following factors:

- $d_{i,a}$: Normalized distance between agent a's current location and the location of shop i.
- A_i: General attractiveness of shop i.
- $I_{i,a}$: Agent a's personal interest towards shop i.

The attributes of X_i in Eq. (2) for this scenario are:

$$X_i = [\frac{I_{i,a}}{d_{i,a}} \qquad I_{i,a} A_i] \tag{10}$$

The probability of agent a choosing shop i to visit becomes:

$$P_{i,a} = \frac{e^{I_{i,a}(\frac{\beta_0}{d_{a,i}} + \beta_1 A_i)}}{\sum_j e^{I_{j,a}(\frac{\beta_0}{d_{j,a}} + \beta_1 A_j)}} \tag{11}$$

where β_0 and β_1 are the weights of the first and second attributes, respectively, in X_i. To model the attractiveness of shop i, we use Eq. (9) in the evacuation scenario to compute A_i.

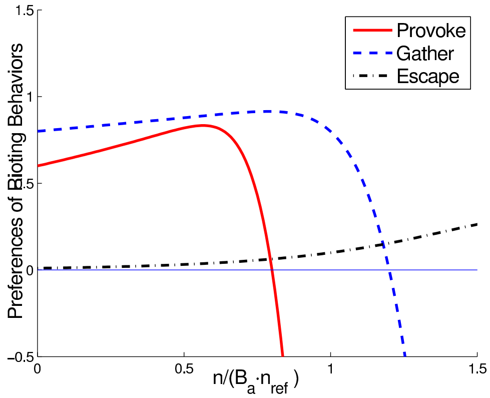

Figure 2: These curves are generated based on our dynamic feedback model. When rioters outnumber the police, the chance of gathering or being provoked is relatively high. When there are more police agents than rioters, most of rioters choose to escape.

4.3 Rioting

Rioting is a type of interaction between rioters and the police. Facing the police, rioters will provoke, gather or run away. We may define the factors to model rioters' representative utility for choosing a specific behavior as follows:

- B_a: Braveness or aggressive value for rioter a.
- d_a: Normalized distance between rioter a and the nearest group of police agents.
- $A_{i,a}$: Preference of a specific behavior i (provoke, gather or run away) to agent a.

The attributes of X_i in Eq. (2) for this scenario are:

$$X_i = [k_{i,d}(B_a d_a) \qquad A_{i,a}] \tag{12}$$

Here, $k_{i,d}$ is set to +1 for provoking and gathering, and -1 for escaping. Now, the probability of rioter a choosing a rioting behavior i becomes:

$$P_{i,a} = \frac{e^{\beta_0 k_{i,d}(B_a d_a)+\beta_1 A_{i,a}}}{\sum_j e^{\beta_0 k_{j,d}(B_a d_a)+\beta_1 A_{j,a}}} \tag{13}$$

To model the preference of a behavior i, we use the dynamic feedback model to compute $A_{i,a}$ as follows:

$$A_{i,a} = v_{i,pos} e^{k_{i,pos}(n/(B_a n_{i,ref}))} - v_{i,neg} e^{k_{i,neg}(1-\sqrt{(B_a n_{i,ref}/n)})} \tag{14}$$

where n is the ratio of the numbers of police and rioters. $B_a n_{i,ref}$ is the rioter a's preferred ratio of the police and the rioters for a specific rioting behavior i. The preferences of these three rioting behaviors are set differently, which is demonstrated in Fig. 2.

5 Implementation and Results

Our algorithm can serve as a high-level decision making scheme on top of global navigation and local collision avoidance, as shown in Fig. 3. We have integrated it with two well-known local collision avoidance models: the Social Force model [3] and Reciprocal Velocity Obstacle (RVO) [18]. Experiments are performed on a PC with an Intel Core 2 Quad 2.83GHz CPU and NVIDIA GeForce GTX 280. The real-time simulation performance of goal selection performed on CPU is shown in Fig. 6.

We discuss the implementation details and highlight the results on evacuation, tradeshow tour and rioting scenarios in the following subsections.

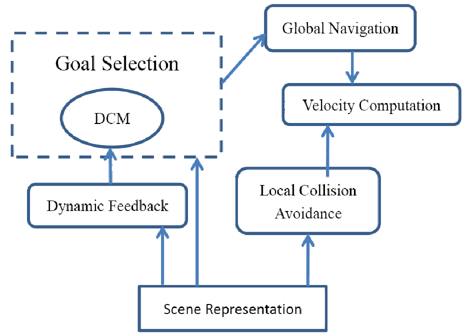

Figure 3: Crowd Simulation using DCM: The DCM formulation is used as a goal-selection scheme to determine goal positions to direct agents' navigation.

Para.	Evacuation	Tradeshow	Rioting		
			Provoke	Gather	Escape
n_{max}	25	25	0.8	1.2	2.0
v_{pos}	0.4	0.1	0.6	0.8	0.01
k_{pos}	0.92	2.30	0.51	0.22	4.61
v_{neg}	1	1	1	1	1
k_{neg}	18.9	18.9	18.9	18.9	18.9
$beta_0$	2.0	7	1	1	1
$beta_1$	-0.1	6	1	1	1

Table 1: Parameter settings in the three scenarios, for reference.

5.1 Evacuation

We simulate agents escaping from a square two-exit and four-exit room. The parameter setting refers to the table 1.

We first build our formulation upon the social force model (SF). Our experiment compares the DCM formulation and the Shortest-path strategy (SP) to the exit with 200 agents. SF leads to inter-penetration when the crowd density is high so we evaluate this pair of techniques by counting the number of inter-agent collision times. The following result shows that the DCM formulation performs better in collision and congestion avoidance:

Group	Collision Times per run	Collision Times per agent
SF+SP	4363.8	21.82
SF+DCM	2064.2	10.32

The second experiment is based on RVO. We compare the DCM formulation with the Shortest-path strategy and the Random Selection strategy, by evaluating the speed of evacuation from both 2-exit and 4-exit rooms with 400 agents as shown in Fig. 4.

5.2 Tradeshow Tour

We would like to demonstrate the advantage of the dynamic feedback model over the Random Selection strategy. The scene is a small section of the tradeshow hall, where the agents choose to visit different booths or to leave the hall. We record the visiting times during 1 minute simulation as shown in Fig. 5. We observe that our feedback model offers an easy way to control the number of visitors by changing the maximal number of visitors. Since the maximum number of visitors is set to 25, the curve is maintained around 25. On the contrary, random selection often leads to drastic fluctuation. Another advantage of our method is that it provides a tradeoff between positive and negative feedbacks while random selection does not. In Fig. 5, once our method has reached the maximal number limit, the curve drops immediately and then later recovers.

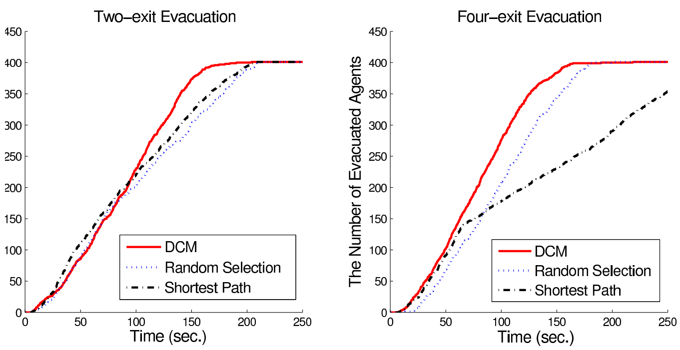

Figure 4: Evacuation: In a dense crowd, random goal selection possibly causes agents to collide with each other, which slows down the evacuation. The simulation using shortest path strategy easily fails due to congestions near the exits. Our DCM formulation performs better than the other two approaches. In the early stage, three methods perform similarly since the congestions exists. In the following stage, our approach gradually reduces the congestions and make the evacuation faster.

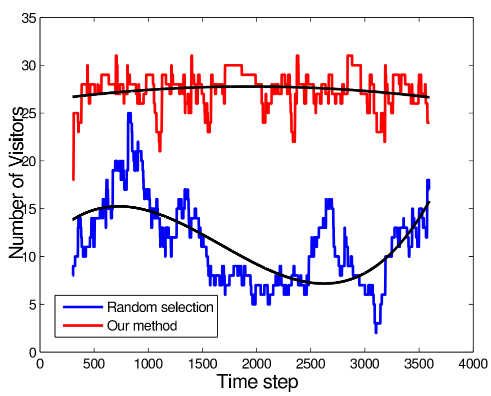

Figure 5: Tradeshow: Our method produces a more stable curve and better tradeoff between positive and negative feedbacks when compared with random selection.

5.3 Rioting

In a rioting scenario, we would like to show that our algorithm can generate heterogenous crowd motions by performing dynamic goal selection. The scenario shows that a large number of riot agents gather together and try to break through police blockade until more police agents come to reinforce. To our knowledge, there is no simple high-level decision making model such as the social force model or the random selection strategy that can simulate such crowd behaviors without manually determining behavior rules.

6 Conclusion

We propose a new technique for simulating agents' decision making based on the Discrete Choice Model integrated with a dynamic feedback model. It can simulate heterogeneous crowd behaviors and work well with local collision avoidance models. Yet, it has limitations including being constrained to certain crowd scenes and providing many control variables.

There are many avenues for future work. Current model only considers short-term goal selection. Many behaviors require long-term plans. Additionally, the formulation can be applied to motion synthesis.

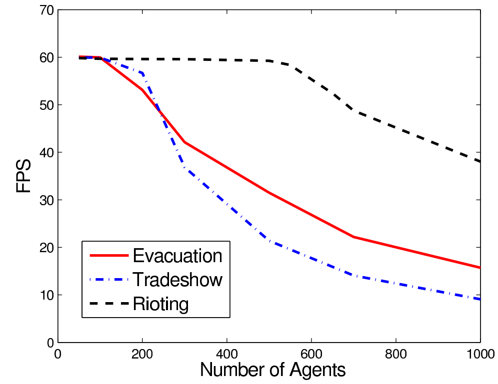

Figure 6: Real-time performance of the DCM formulation in three scenarios.

References

[1] J. Funge, X. Tu, and D. Terzopoulos. Cognitive modeling: knowledge, reasoning and planning for intelligent characters. In *Proc. ACM SIGGRAPH*, pages 29–38, 1999.

[2] A. Goett, K. Hudson, and K. Train. Customers' choice among retail energy suppliers: The willingness-to-pay for service attributes. *The Energy Journal*, **21**(4):1–28, 2000.

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

[4] D. McFadden. Conditional logit analysis of qualitative choice behavior. *Frontiers and Econometrics*, 1974.

[5] M. Moussaid, S. Garnier, G. Theraulaz, and D. Helbing. Collective information processing and pattern formation in swarms, flocks, and crowds. *Topics in Cognitive Science*, **1**(3):469–497, 2009.

[6] S. Patil, J. van den Berg, S. Curtis, M. Lin, and D. Manocha. Directing crowd simulations using navigation fields. *IEEE Trans. on Visualization and Computer Graphics*, **17**(2):244–254, 2011.

[7] N. Pelechano, J. Allbeck, and N. Badler. *Virtual crowds: methods, simulation and control*. Morgan and Claypool Publishers, 2008.

[8] N. Pelechano, K. O'brien, B. Silverman, and N. Badler. Crowd simulation incorporating agent psychological models, roles and communication. In *Proc. Int'l Workshop on Crowd Simulation*, 2005.

[9] W. Shao and D. Terzopoulos. Autonomous pedestrians. In *Proc. ACM Symp. on Computer Animation*, pages 19–28, 2005.

[10] A. Sud, R. Gayle, E. Andersen, S. Guy, M. Lin, and D. Manocha. Real-time navigation of independent agents using adaptive roadmaps. In *Proc. ACM VRST*, pages 99–106, 2007.

[11] M. Sung, M. Gleicher, and S. Chenney. Scalable behaviors for crowd simulation. **23**(3):519–528, 2004.

[12] D. Terzopoulos, X. Tu, and R. Grzeszczuk. Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world. *Artificial Life*, **1**(4):327–351, 1994.

[13] D. Thalmann. Crowd simulation. *Wiley Encyclopedia of Computer Science and Engineering*, 2007.

[14] D. Thalmann, C. O'Sullivan, P. Ciechomski, and S. Dobbyn. *Populating virtual environments with crowds*. Eurographics Tutorial Notes, 2006.

[15] K. Train. A validation test of a disaggregate mode choice model. *Transportation Research*, **12**(3):167–174, 1978.

[16] K. Train. *Discrete choice methods with simulation*. Cambridge University Press, 2003.

[17] A. Treuille, S. Cooper, and Z. Popovic. Continuum crowds. In *ACM SIGGRAPH*, pages 1160–1168, 2006.

[18] J. van den Berg, M. Lin, and D. Manocha. Reciprocal velocity obstacles for real-time multi-agent navigation. In *IEEE Int'l Conf. on Robotics and Automation*, pages 1928–1935, 2008.

[19] Q. Yu and D. Terzopoulos. A decision network framework for the behavioral animation of virtual humans. In *Proc. ACM Symp. on Computer Animation*, pages 119–128, 2007.