Motivation-based Hierarchical Behavior Planning

송웨이°, 조경은*, 엄기현**
Wei Song°, Kyungeun Cho*, Kyhyun Um**
Dept. of Game & Multimedia Engineering, Dongguk University
Dept. of Multimedia, Graduate School of Digital Image & Contents, Dongguk University°, *, **
{songwei°, cke*, khum**}@dongguk.edu

ABSTRACT

This paper describes a motivation-based hierarchical behavior planning framework to allow autonomous agents to select adaptive actions in simulation game environments. The combined behavior planning system is formed by four levels of specification, which are motivation extraction, goal list generation, action list determination and optimization. Our model increases the complexity of virtual human behavior planning by adding motivation with sudden and cumulative attributes. The motivation selection by probability distribution allows NPC to make multiple decisions in certain situations in order to embody realistic virtual humans. Hierarchical goal tree enhances the effective reactivity. Optimizing for potential actions provides NPC with safe and satisfying actions to adapt to the virtual environment. A restaurant simulation game was used to elucidate the mechanism of the framework.

요 약

본 논문에서는 동기 기반의 계층적 행동 계획 시스템을 제안한다. 가상 시뮬레이션 게임 환경에서 에이전트는 행동 계획 시스템을 통해 적합한 행동을 선택하게 된다. 행동 선택 시스템은 동기를 추출하고 목표를 선택하고 행동을 생성하고 최적화를 수행한다. 동기를 평가할 때 갑작스럽게 발생하거나 누적된 이벤트에 대해 계산한다. 동기를 선택할 때는 확률 분포를 사용하여 무작위로 선택한다. 계층적 목표 트리를 탐색한 후에 목표를 실행할 수 있다. 행동들을 비교한 후 가장 적합한 행동을 선택하게 된다. 선택을 할 때 안전도값과 만족도 값을 비교하여 최적화된 행동을 선택한다. 본 연구에서 제안한 시스템을 식당경영 게임에 적용했다

Keyword : autonomous agent, behavior planning, motivation, probabilistic distribution, simulation game

※ Corresponding Author:Kyungeun Cho, Address:(100-715) 26, 3-ga, Pil-dong, Chung-gu, Seoul, Korea, TEL:+82-2-2260-3834, FAX:+82-2-2260-3766, E-mail:cke@dongguk.edu

I Introduction

Artificial intelligence (AI) plays an important role in the field of games. As computer graphics have developed, game AI has also undergone a quiet revolution. Different game genres like RTS (Real-Time Strategy), FPS (First Person Shooting), and RPG (Role-Playing Game) have some common characteristics such as offering an environment in which to test ideas about creating Non Player Characters (NPC) which are modeled after human beings. Interacting with these artificial creatures, the human player can experience how well these computer players display human-like behavior and thinking [1].

Behavior planning is a way of implementing the most basic issue of artificial intelligent systems. It solves the problem what to do next. According to behavioral manners of human beings and animals, the autonomous action selection problem is typically designed with human or animal behavior [2].

Traditional behavior planning uses several basic mechanisms like FSM (Finite State Machine), PSM (Probabilistic State Machine), and Fuzzy algorithm, among others. These conventional methods deal only with a few simple action selections. They are limited in achieving a human-like level. As game AI technology is investigated by both game programmers and virtual reality researchers, high level behavior planning is developed rapidly, such as HTN (Hierarchal Task Network) and search based planning [3].

In simulations and games, autonomous agents do not only need to adapt for a static environment, but must also consider interacting with other agents and human players. Interactive action is considered to be complex behavior.

Based on human performance, this study designed a behavior planning framework combining four processes, which are motivation extraction, goal decision, action generation and behavior optimization.

Because the environment information is continuously changing, agents that motivate action vary with different situations. Immediate motivation selection is necessary for autonomous agents to adapt to dynamic environment. Motivated automata allow the agent to choose the most appropriate action according to its perception of the environmental conditions.

With the aim of providing the autonomous agent with more safety and satisfaction, this paper proposes a motivation-based hierarchical behavior planning system to extend the capability of the traditional hierarchical behavior planning networks.

In Part 2, related works on behavior planning are introduced. In Part 3, a motivation-based hierarchical behavior planning system is designed. In Part 4, the planning structure is tested in a virtual restaurant environment and this paper's conclusion and comparisons with other research are given.

II Related works

Recently, high level behavior planning is developed rapidly. HTN (Hierarchical Task Network) is an exploration algorithm for making a decision by hierarchical sub-task generation. MinMin algorithm helps to make an optimization of selecting an effective action sequence. Search based planning system is a potential prediction structure for avoiding the danger. Hierarchical classifier system is a goal-oriented behavior planning model for action generation.

M. Lozano [4] described a prototype in which character behavior is driven by a real-time MinMin searching algorithm. By selecting an action sequence with less weight, virtual agent can control its behaviors efficiently. But the MinMin searching system is organized without multiple precondition situations.

J. L. Lugrin[5] described an emergent storytelling prototype by means of a search-based hazardous action generation algorithm. In the system, less-dangerous action is intercepted; however, the various determinants of causality should be reconciled to enhance the system.

M. Cavazza [6] describes a method for implementing the behavior of artificial actors in the context of interactive storytelling. The artificial actor generates behavior by the mean of HTN to interact with both other agents and environment.

Etienne de Sevin and Daniel Thalmann [7] have proposed a motivational model of action selection and hierarchical classifier system, in which the behavior is selected according to the motivation. However, the author makes the remark that the behavioral planner is not complex enough and should be replaced by one that manages complex behaviors.

In this paper, the problem of representing a new complex behavior planning system of extracting motivation, deciding goals, generating action and optimization is researched. An autonomous agent behavior planning system is divided into these four parts in order to adapt to the environment more effectively and completely. These human-like actions play the role of generating coherent autonomous behavior in a dynamic environment.

This study tested motivation-based behavior planning frameworks in the restaurant simulation game "Crazy Waitress" so as to clarify how to implement complex action with more satisfaction and safety.

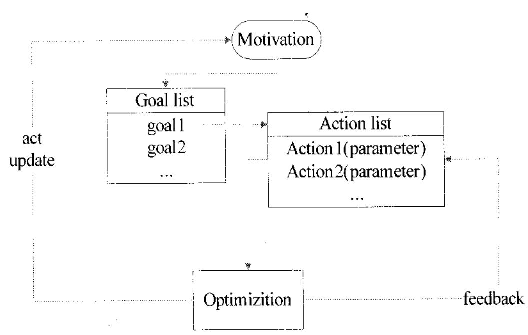
III Extended search-based system

The architecture of motivation-based hierarchical behavior planning is presented in this section. The planning system includes extracting motivation, deciding goals, generating action and optimization. The main network for generating adaptive and interactive behavior is designed as the framework in Fig.1.

The motivation process includes motivation evaluation and extraction. We suggest a motivation evaluation algorithm denoting the effective value of sudden and cumulative events. After extracting one active motivation out of all the defined motivations by sampling from probability distribution, the system will explore subgoals to build hierarchical goal tree.

Then autonomous agent will generate an actions set in order to satisfy the selected subgoal from bottom to top of the goal tree until the motivation is satisfied.

Finally, an optimization planner is added to select most adaptive action. The generated action will update the virtual world which repeatedly activates the autonomous agents.



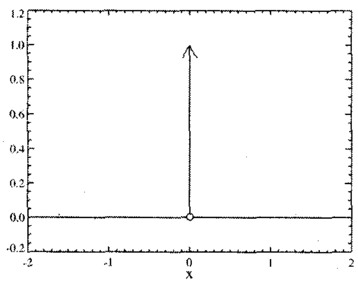
[Fig. 1] Motivation-based planning framework

3.1 Motivation evaluation and selection

Motivated automata allow the agent to choose the most appropriate action according to its proprioception and environmental conditions.

There are many kinds of motivations to be considered in the virtual reality simulation such as food needs, water needs, sleeping needs, danger avoidance and so forth. Because the motivation value is changing with time, we define a variable $M(t)$ to denote motivation value along with different time.

In daily life, there are two kinds of motivation forms: cumulative and sudden situations. In equation (4) we use a integral function $\int_0 f(t)dt + M(t_0)$ to simulate cumulative motivations like hunger and thirst, where $f(t)$ is the rate of motivation increment, and the integral constant $M(t_0)$ is the base value of the motivation happening at time t_0. Impulse function $\int I_i(t)\delta(t-t_i)$ is suggested to simulate sudden events like being beaten by others. Delta function $\delta(t)$ can be loosely thought of as a function on the real line which is zero everywhere except at the origin, where it is infinite (see Fig.2) [8].



[Fig. 2] Impulse function

The mathematical definition is given in Function (1), which is also constrained to satisfy the identity (2).

$$\delta(t) = \begin{cases} \infty & t = 0 \\ 0 & t \neq 0 \end{cases} \tag{1}$$

$$\int_{-\infty}^{\infty} \delta(t)dt = 1 \tag{2}$$

The following property of the delta function can be used to formulize a sudden event. For any continuous function $I(t)$,

$$\int_{-\infty}^{\infty} I(t)\delta(t-t_0)dt = I(t_0) \tag{3}$$

According to Equations (1), (2) and (3), the time the sudden event happens is defined as $t()$ and the amplitude estimation can be fed to the motivation value.

Sudden events happen in unpredictable time so that autonomous agent should detect these real time signals which affect the motivation value. By finding the sum of the affection result and the cumulative motivation function, Equation (4) can evaluate real time motivation. The motivation value should be positive. If the generated value is negative, zero will be returned.

$$M(t) = \begin{cases} \int_{t_0} f(t)dt + M(t_0) + \sum_{i \in I} \int I_i(t)\delta(t-t_i)dt & M(t) > 0 \\ 0 & M(t) \leq 0 \end{cases} \tag{4}$$

After evaluating the motivation value, extraction of the effective motivation is executed by means of probability distribution. According to each motivation value, the probability of the activating index can be calculated by Equation (5):

$$\mathrm{P(M)} = \frac{M(t)}{\sum_i M_i(t)} \qquad i = \{i \mid motivation\_index\} \tag{5}$$

Upon finding the sum of each motivation happening probability, one will be returned. It is satisfied by applying the cumulative distribution function (CDF), defined in Equation (6), to intercept a reasonable motivation randomly.

$$F_X(x) = P[X \leq x] \tag{6}$$

The function *rand()* is defined as a random value from 0 to 1. If the generated random value is between $F_X(x)$ and $F_X(x+1)$, the motivation selection system will generate the index x as the current motivation. Therefore the current motivation selection can be derived by Equation (7),

$$M = F^{-1}(F_X(x-1) < rand() < F_X(x)) \qquad (7)$$

where F^{-1} is the inverse function of CDF; $F(x)$ is the motivation probability distribution value due to different index variable x.

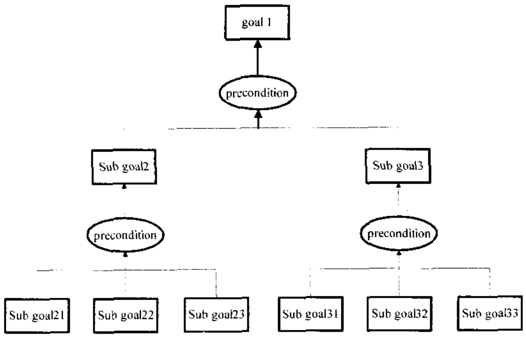
For example, motivation index is defined as i={i|ANGER=1, HUNGER=2, FULL=3} with the probability that $P(M_1)=0.2$, $P(M_2)=0.3$ and $P(M_3)=0.5$. By Equation (6), motivations CDF value is calculated as $F_X(0)=0$, $F_X(1)=0.2$, $F_X(2)=0.5$ and $F_X(3)=1$. If rand()=0.4 satisfying $M = F^{-1}(F_X(1) < rand() < F_X(2))$, motivation index should equal to 2 in this situation and motivation of hunger is generated.

3.2 Goal tree generation

Notice that unlike traditional planning approaches, behavior cannot be specified for motivations, since a motivation is not guaranteed to succeed. Thus, we can only specify which goal a special motivation pursues.

The process part of goal list generation is proposed consisting of the following constructs: motivation solution and sub-goal sequence decision. Sub-goal means that the execution engine must find another behavior that must be executed to satisfy that particular sub-goal [9]. These generated goals can be structured in a specialization hierarchy in order to specify the relationships among them.

Goal attributes are proposed with two parameters which are precondition and sub-goals. The goal can be implemented only when the precondition is satisfied. Meanwhile it is necessary to execute some sub-goal firstly for that precondition to result. This process is shown in Fig.3.



[Fig. 3] An example of goal tree

In the tasks of solving sub goals, the first considered goal will be met last. Therefore based on a last-in-first-out (LIFO) algorithm and search tree principle, a goal list is created as shown in Table.1.

[Table. 1] goal-based planning by LIFO

goal	Precondition	Subgoal
Gn	Current condition	null
…	…	…
G3	c3={c31,c32,⋯,c3i,⋯}	G31,G32,⋯,G3i,⋯
G2	c2={c21,c22,⋯,c2i,⋯}	G21,G22,⋯,G2i,⋯
Motivation	c1={ c2,⋯,ci,⋯}	G2, G3,⋯,Gi,⋯

First of all, motivation as a basic goal is put into the goal stack. To achieve the motivation needs, a basic set of preconditions is essentially. These preconditions lead to some new sub-goals that must be completed first. In this way the new sub goals will be put into the stack till the precondition can be satisfied in the agent current condition.

For instance, if the preconditions of goal g2 are c21 and c22, sub-goals g31 and g32 will be selected to satisfy the preconditions.

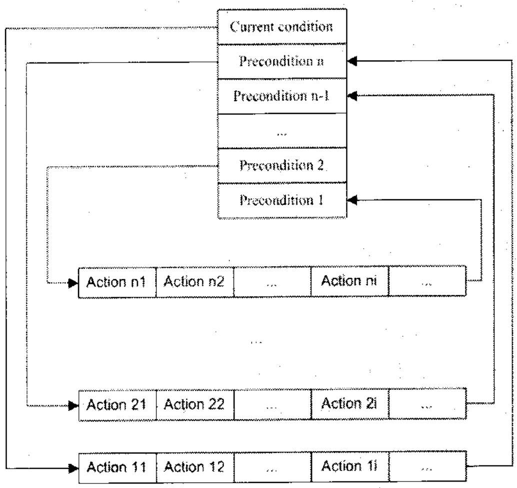
3.3 Multiple action decision

According to the generated goal list or goal stack, some actions caused by relevant goals can be selected as shown in Table.2. There are several ways to achieve the generated goal. For example, if the goal is to eat something, the solution can be to eat, to drink, or even parallel actions like both eating and drinking. Because of the relationship with the goal list, the action list is created to satisfy the preconditions from the current condition to motivation. The first generated action will be operated first, which follows the first-in-first-out(FIFO) algorithm.

[Table. 2] Action list by FIFO

goal	action	result
$g_{n-1} \in G_{n-1}$	A1={a11,a12,⋯,a1i,⋯}	Cn={cn1,cn2,⋯,cni,⋯}
$g_{n-1} \in G_{n-1}$	A2={a21,a22,⋯,a2i,⋯}	C2={c21,c22,⋯,c2i,⋯}
...	...	...
Motivation	An={an1,an2,⋯,ani,⋯}	C1={c11,c12,⋯,c1i,⋯}

Action parameters can be divided into external and internal parts. External parameters, considered as precondition and result sections, are defined to interact with the goal. The precondition section restricts the feasibility of the action and the result section leads to goal achievement. The external parameters are crucial in deciding the goal list for which the working flowchart is shown in Fig.4. Internal parameters design action manners like speed, involved objects and so on. The behavior safety and satisfaction value relates with these internal factors.

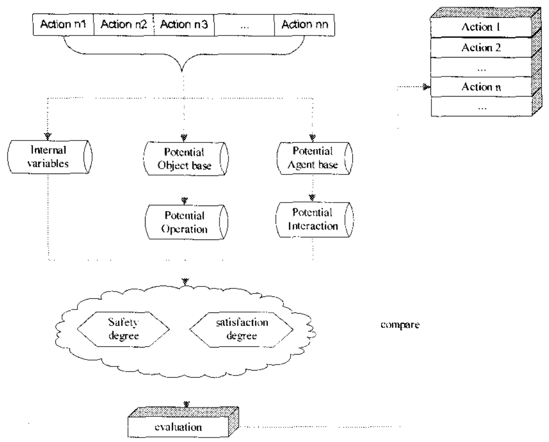


[Fig. 4] Action list generation

3.4 Adaptive action generation

To determine the best action for each sub-goal, a deep search network is designed as in Fig.5. The main purpose of the optimizing process is to evaluate the degree of safety and satisfaction by comparing potential factors.

The potential action attributes are divided into three parts: internal variables, potential involved objects and potential interactive agents. Internal variables are defined as the action parameters which the agent can control autonomously. Taking the speed factor as an example, the autonomous agent should change their behavior as the speed differs but in a limited range. Potential involved objects mean those objects which can be operated by the agent in the next state. Potential interactive agents mean those which will interact with the autonomous agent in next state.



[Fig. 5] Optimizing process

If the action provides the motivation with lower danger and greater satisfaction, the higher value will be returned. After comparing each action in the list, the best-fitting behavior will be selected by Equation (8) and put into the planning stack.

$$A_n = \mathrm{argmax}(e_{safety}(a_i) + e_{satisfaction}(a_i)) \quad (8)$$

Ⅳ. System experiment

To elucidate the mechanism of a motivation-based hierarchical behavior planning system, the problem of building an autonomous agent with intelligent planning is examined in a restaurant simulation game, 'Crazy Waitress.'

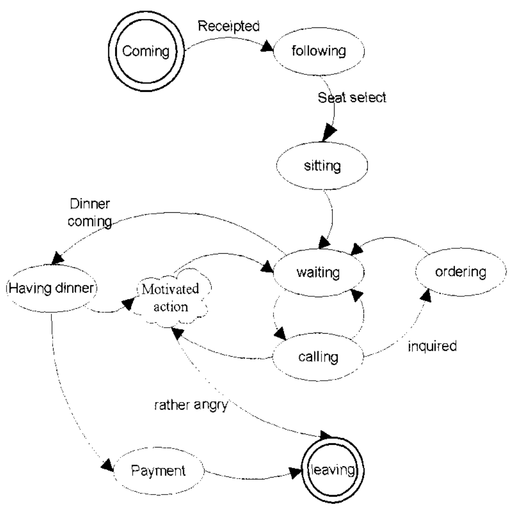
This game is implemented using C++.NET as the main developing environment and G-Blender as the game programming engine.

4.1 Design and implementation

As the aim is creating an autonomous character with interactive and optimum behavior, the motivation-based hierarchical behavior planning structure is implemented in a virtual reality simulation game. The game environment is a restaurant in which the waitress is controlled by a human player and the guests are non-player characters.

In this project we examine the problem of building autonomous behavior planning starting from a dynamic environment and proprioceptive motivation. After comparing all of the motivations value, the most effective motivation is selected to activate the agent behavior.

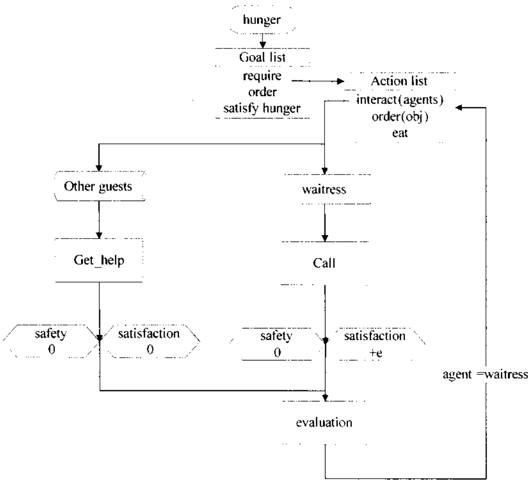
Fig.6 describes the FSM for NPC as a guest. Traditional FSM is limited to a dynamic motivation process. Therefore the hierarchical FSM is used to extend capability for motivation level. Fig.7 gives an example of the dynamic motivation based behavior planning process as a sub-FSM.



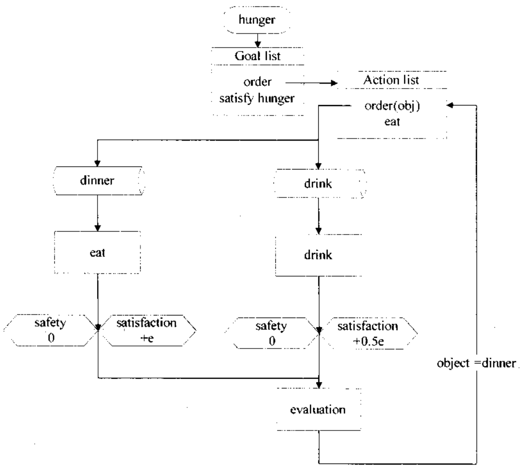
[Fig. 6] Guest FSM design

In different situations, NPC's external and internal motivations are different. The motivations presented in this simulation are considered as food, drink, reply and toilet needs.

During the waiting state, the degree of the need for food (hungry) increases with time. After calling for service, the degree for the need (angry) of a reply will be activated by a positive impulse. If the waitress serves him or her, the degree of reply-need will decrease by a negative impulse. While eating, the need for the toilet (full) increases as it would with humans. According to each motivation value, the motivation probability distribution is calculated by a CDF function defined as Equation (6). To satisfy the sampling motivation, a dynamic search-based FSM is updated (Fig.7) as an instance of hunger motivation.



(a) Hunger motivation



(b) After waitress inquiry

[Fig. 7] Example of dynamic motivation-based planning

In Fig.7 (b), it shows that to have dinner is better to satisfy the motivation of hunger than to have drink. Therefore ordering object should be “dinner”.

4.2 Simulation result

The restaurant simulation game ‘Crazy Waitress’ is presented in Fig.8. The simulation game is a multi-agent serving system in which NPC characters act synchronously. They might follow such an action sequence as coming, following, waiting, calling, eating and leaving. Human players should serve these actions with such parallel behaviors as receiving, leading, inquiring and dinner serving. It’s a complex system for making autonomous planning with different motivations.

(a) A screen shot of experimental game

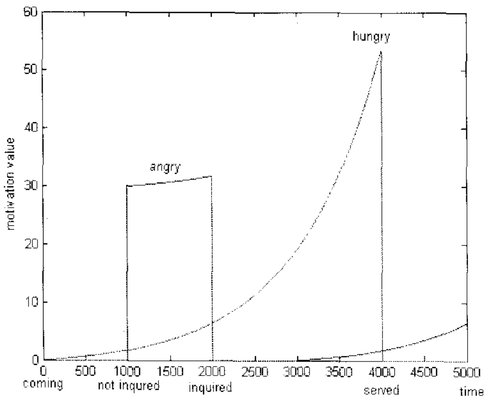
1	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	
2	hunger	thirsty	angry	full	motivation
	1.54	0.59	1.51	0.00	hungry
3	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	
4	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	
5	hunger	thirsty	angry	full	motivation
	2.93	0.98	10.21	0.00	angry
6	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	
7	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	
8	hunger	thirsty	angry	full	motivation
	1.51	0.58	3.32	1000.00	full
9	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	
10	hunger	thirsty	angry	full	motivation
	0.00	0.00	0.00	0.00	

(b) Status information of the shot (a)

[Fig. 8] Simulation environment

In the game, the NPC can generate reasonable actions according to motivation condition. By means of the probability distribution of internal and external motivations, there is the probability that all of the motivations can be extracted. The sampling frequency is proportional to motivation value. The results of the performance of this experiment are more active and logical than traditional, randomly-selecting actions.

When taking hunger and anger as motivations, for example, the coherent motivated result is shown in Fig.9. The two curves stand for the hunger and anger motivation separately. After extracting the motivation value by sampling from probability distribution, sub-goals and actions will be generated.



[Fig. 9] Simulation result

4.3 Discussion

VRlab, led by Prof.Thalmann, researches real time simulations for the virtual world. They have created an algorithm to select motivation due to the motivation value. Because of similar motivation value situations, suitable autonomous planning is difficult to select. Also, game performance will be less changeable for motivations with wide disparity. The motivation with the largest value is always intercepted. To resolve this problem, this study proposes a probability distribution to present parallel action generation. For example, the agent can drink while having dinner through the hunger motivation.

Table 3 gives a mathematical comparison between VRlab's research and the work detailed here. (The evaluation of motivation in the paper [7] is given that $M = T_1 e^{(i-T_1)^2}$, if i < T_1. To make motivation value generation starting from 0, the equation should be $M = ie^{(i-T_1)^2}$ for $i < T_1$. That's a mistake.) The defined threshold T_1 and T_2 of VRlab's algorithm is used to determine the motivation equation in different situations.

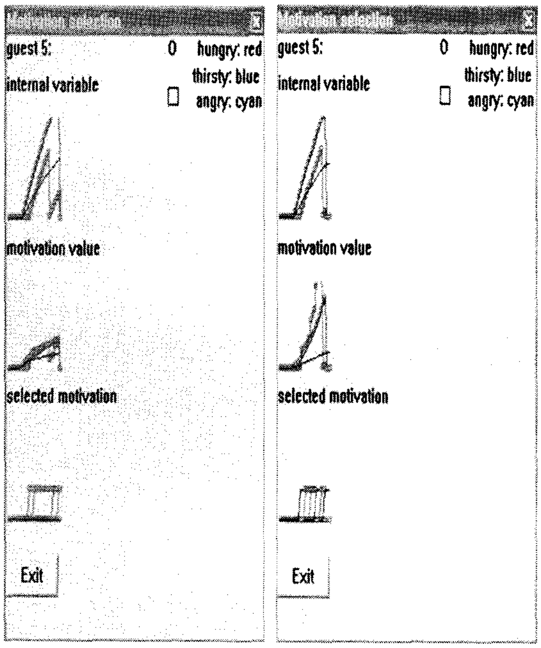
[Table. 3] Algorithm comparison

	VRlab	RISE group
Motivation evaluation	$\begin{cases} M = ie^{(i-T_1)^2} & i < T_1 \\ M = i & T_1 \le i \le T_2 \\ M = \frac{i}{(1-i)^2} & i > T_2 \end{cases}$ $M_t = (1-\alpha)M_{t-1} + \alpha(M + e_t)$	$M(t) = \int_0 f(t)dt + M(t_0) + \sum_{i \in I} \int_0 I_i(t)\delta(t - t_i)dt$
Motivation extraction	maximum value	probability distribution
Variables	T_1: threshold of comfort and tolerance zone. T_2: threshold of tolerance and danger zone. i: internal variable.	$f(t)$: internal increment rate $I_i(t)(\underline{t-t_i})$: feed from sudden event.
Result	Only the motivation with maximum value is considered.	All of the motivations can probably be extracted. The sampling frequency is proportional to motivation value.

Fig.10 (a) and (b) give simulation results of Thalmann's algorithm and my proposed algorithm separately. Both of the simulations are tested with same internal variable.

In the simulation, the full motivation (toilet need), which remains zero before having dinner, is skipped for the clarity of simulation data curves.

The proposed method generates motivation following both sudden events and cumulative stimulation so that agent autonomy is sufficient for a dynamic environment. Because the motivation is intercepted by probability distribution, the generated behavior works in a parallel way instead of singly.



(a) Thalmann' s algorithm (b) Proposed algorithm

[Fig. 10] Performance comparison

In Figure.1 (a) of Thalmann's simulation result, it can be seen that the highest motivation is selected. The motivation with the smaller value is never considered like thirsty. In case of the hysteresis equation application, if the internal variable become 0, we can see that character's anger motivation till keeps αM_{t-1}. That's the shortcoming to imitate human behavior in virtual world.

In Figure.1 (b), the generated motivations are randomly sampled with the frequency directly proportional to the relevant motivation value. The smaller motivation of thirsty is also considered in our designed system. It can reflect the real life more precisely.

Meanwhile, Thalmann's hierarchical decision loop helps to choose the most activated action with the goals as objects and places. However some action should be taken after satisfying some special preconditions. Taking FPS game as an example, if shooting enemy, NPC should carry a gun with more than one bullet. Our proposed goals tree is aimed at satisfying the action's precondition. According to the precondition definition, the goal sequence is built.

In Thalmann's paper, the "compromise behaviors" is modeled to select actions by the compromise action activity estimation. It's the process of calculating the activity of each defined action linked to the goal. We propose that goal is not appointed with special action. There are many potential actions and different action parameters. The NPC should select the most appropriate action with more safety and more satisfaction to the motivation.

Future research on the expression of emotion is proposed to increase the flexibility and sociability of the planning model.

References

[1] Jean Paul van Waveren, "The Quake III Arena Bot", Master of Science thesis Delft University of Technology, June 2001.

[2] C. Brom, J. Lukavský, O. Šerý, T. Poch, P. Šafrata, "Affordances and level-of-detail AI for virtual humans", Proceedings of Game Set and Match 2, 2006.

[3] J. J.Bryson, "Action Selection and Individuation in Agent Based Modelling", Proceedings of

Agent 2003: Challenges in Social Simulation, Argonne National Laboratory, Argonne, IL, USA. pp. 317 - 330, 2003.

[4] M. Lozano, S.J. Mead, M. Cavazza, and F. Charles, "Search-based Planning for Character Animation", ADCOG 2002, Hong Kong, China. pp 41-43, 2002.

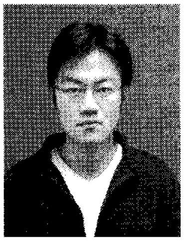
[5] J-L. Lugrin and M. Cavazza, "AI-based world behavior for emergent narratives", Proceedings of the ACM Advances in Computer Entertainment Technology, Los Angeles, USA, article No.25, 2006.

[6] M. Cavazza, F. Charles, and S. J. Mead, "Planning characters' behaviour in interactive storytelling", The Journal of Visualization and Computer Animation, Volume 13, Number 2, pp. 121-131(11), May 2002.

[7] E.d. Sevin and D. Thalmann, "A Motivational Model of Action Selection for Virtual Humans," In Computer Graphics International (CGI), IEEE Computer Society Press, New York, pp 213-220, 2005.

[8] R.D. Yates, D.J. Goodman, Probability and stochastic processes a friendly introduction for electrical and computer engineers, John Wiley & Sons, Inc., 2005.

[9] Santiago Ontañón, Kinshuk Mishra, Neha Sugandh, Ashwin Ram, "Case-Based Planning and Execution for Real-Time Strategy Games," ICCBR 2007, pp 164-178, 2007



Song, Wei

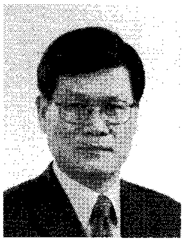
2001. 9~2005. 7 B.S. Software College of Northeastern University, China.
2006. 3~present M.E. Dept. of Multimedia,Graduate School of Digital Image & Contents, Dongguk University.

Research Interests : Artificial Intelligence for Games, Game Algorithm

Cho, Kyungeun

1989. 3~1993. 2 Computer Science, Dongguk University (BS)
1993. 3~1995. 2 Computer Engineering, Dongguk University (MS)
1995. 3~2001. 8 Computer Engineering, Dongguk University (Ph.D)
2002. 3~2003. 2 Dept. of Digital Media, Fulltime Lecturer, Anyang University
2003. 3~2003. 9 Dept. of Game Engineering, Fulltime Lecturer, Youngsan University
2003. 9~2005. 8 present Dept. of Computer Multimedia Engineering, Fulltime Lecturer, Dongguk University
2005. 9~present Dept. of Game & Multimedia Engineering, Assistant Professor, Dongguk University.

Research Interest : Artificial Intelligence for Games, Game Algorithm, Computer Vision.



Um, Kyhun

1971. 3~1975. 2 Dept. of Applied Mathematics, Engineering College, Seoul National University (BS)
1975. 3~1977. 2 Dept. of Computer Science, Korea Advanced Institute of Science and Technology (MS)
1986. 3~1994. 2 Dept. of Computer Engineering, Graduate School, Seoul National University (Ph.D)
1978. 3~2006. 6 Dept. of Computer and Multimedia Engineering, Full Professor, Dongguk University
2006. 7~present Dept. of Game and Multimedia Engineering, Full Professor, Dongguk University
2001. 3~2003. 2 College of Information and Industrial Engineering, Dean, Dongguk University
1995. 3~1999. 2 Information Management Institute, chief director, Dongguk University
1998. 8~2000. 7 Korea Information Science Society, SIGDB Chair
1999. 4~2005. 4 Int. Conf. on Database Systems for Advanced Applications (DASFAA) Steering Committee member
2007. 1~present Korean Multimedia Society, President
2004. 1~present Korean Game Society, Consulting member

Research Interest : Game Systems, Multimedia Applications