

Title	Survey of How Human Players Divert In-game Actions for Other Purposes: Towards Human-Like Computer Players
Author(s)	Temsiririrkkul, Sila; Sato, Naoyuki; Nakagawa, Kenta; Ikeda, Kokolo
Citation	Lecture Notes in Computer Science, 10507: 243-256
Issue Date	2017-08-25
Type	Journal Article
Text version	author
URL	<a href="http://hdl.handle.net/10119/15417">http://hdl.handle.net/10119/15417</a>
Rights	© IFIP International Federation for Information Processing 2017. Sila Temsiririrkkul, Naoyuki Sato, Kenta Nakagawa, Kokolo Ikeda, Lecture Notes in Computer Science, 10507, 2017, 243-256. The final publication is available at Springer via <a href="http://dx.doi.org/10.1007/978-3-319-66715-7_27">http://dx.doi.org/10.1007/978-3-319-66715-7_27</a>
Description	International Conference on Entertainment Computing, ICEC 2017: Entertainment Computing – ICEC 2017, 16th IFIP TC 14 International Conference, Tsukuba City, Japan, September 18-21, 2017, Proceedings

# Survey of How Human Players Divert In-game Actions for Other Purposes: Towards Human-like Computer Players

Sila Temsiririrkkul<sup>1</sup>, Naoyuki Sato<sup>2</sup>, Kenta Nakagawa, Kokolo Ikeda<sup>3</sup>

Japan advanced institute of science and technology Nomi Asahidai 1-8 923-1211 Japan  
{temsiririrkkul<sup>1</sup>, satonao<sup>2</sup>, kokolo<sup>3</sup>}@jaist.ac.jp

**Abstract.** Human-like behaviors are an important factor in achieving entertaining computer players. So far, the target of human-like behavior has been focused on actions in a game with the goal of winning. However, human behaviors might also be performed with other purposes not directly related to the game's main objective. For example, in FPS games, some human players create illustrations or graffiti with a weapon (i.e., gun). In co-operative online FPS, when chat is not allowed in-game, some players shoot the nearest wall to warn an ally about danger. This kind of action for an indirect purpose is hard to reproduce with a computer player, but it is very important to simulate human behavior and to entertain human players. In this article, we present a survey of the possible actions in a game that are not directly related to the game's main objective. Study cases of these behaviors are collected and classified by task and intention (i.e., warning, notification, provocation, greeting, expressing empathy, showing off, and self-satisfaction) and we discuss the possibility of reproducing such actions with a computer player. Furthermore, we show in experiments with multi-agent Q-learning that such actions with another purpose can emerge naturally.

**Keywords:** Human-likeness·Entertainment·Computer players·Q-learning·multi-agent.

## 1 Introduction

Nowadays, computer game players (game-AIs) are strong enough in term of performance to surpass human players in many domains, especially in classical board games such as chess or the game of Go [1]. In video games, which are more complex, DeepMind showed that a computer player was able to play 49 games, among which 29 were at the level or surpassing human record levels or scores [2-3]. Game-AIs are now strong enough to be a human opponent or partner in terms of performance.

However, performance is not enough to entertain human players. In recent years, generating “human-like” behavior has become an important target among game researchers [4].

For example, one attempt to generate an entertaining game-AI by Ikeda and colleagues presented a method to entertain human players in the game of Go by letting

them win without allowing the players to notice their advantage, by choosing suboptimal actions but avoiding obviously bad ones [5].

Togelius and colleagues introduced the idea of “believability,” which refers to the ability of a character or bot to make someone believe that the character is real or being controlled by a human being [6]. Many approaches were proposed to obtain believability and produce human-like behavior, such as in Fujii and colleagues, where a human-like computer player is obtained by simulating biological constraints. This approach considers human-likeness only in relation to the game’s main objective [7].

In some games, the human player can perform actions “outside” of the game itself, such as bluffing or using facial expressions in poker games, or natural language communication via VOIP programs such as Skype. This communication is performed “outside” of the game itself in order to achieve the game’s main objective. Such actions are also an important target for human-likeness.

However, sometimes, human player actions are not directly related to the game’s main objective. For example, in FPS games, some players try to use their guns to create illustrations with bullet holes; in racing games, some players stop just before the goal, wait until another player comes closer, and then reach the goal. Such actions can be observed in many types of games, such as action games, RPGs, MMORPGs, puzzle games, and racing games. They are sufficiently frequent and significant to be a target (or even possibly a necessity) for obtaining human-likeness in computer agents.

In this research, we focus on human players’ actions in-game that are not directly related to the game’s main objective. We collect study cases from several types of games and classify them into seven types (i.e., warning, notification, provocation, greeting, expressing empathy, showing off, self-satisfaction). We also discuss the context in which these types of actions appear. In addition, we present an experiment that shows how multiple Q agents in an easy hunting game learn to divert game actions from their original goal in a way that we believe is similar to humans.

## **2 Human-like Behavior: Previous Literature and Target Area**

Recently, there are a number of approaches that aim for creating entertaining computer players. Specifically, computer players with human-like mannerisms are a very popular subject of interest among researchers. The idea of human-like AI was originally proposed by Alan Turing in the Imitation Game, which was the starting point of the Turing test [13]. Togelius and colleagues defined “believability” as the ability to make someone believe that the character/bot is being controlled by a human player [6]. So far, believability can be assessed by conducting a Turing test, which is mainly conducted by observing in-game behavior from a third person perspective.

Many computer players/bots in competitions were assessed according to believability assessment in order to indicate their performance in the believability aspect. For example, an assessment of the competition of computer players in an FPS (first person shooter) game was based on an Unreal tournament [10]; another assessment was based on competition in an action side-scrolling game, Super Mario Bros [11].

In an FPS game, finite state machine which represent state by information of combat and collected item, and another approach using behavior tree combine with Neuro

Evolution, performed good performance to play game with human-like behavior [10]. In a Turing test tracking a Mario AI competition in 2012, top ranking human-like computer players used artificial neural network, influence map, and nearest neighbor methods [12]. Also, in recent years, Fujii and colleagues presented approaches to creating believable computer players by using biological constraints, which are applied to Q-learning and the A\* algorithm [7]. These earlier methods are improved based on action/behavior in-game directly concerning the main objective of the game ((1) in Table 1) (e.g., Super Mario Bros: reach the goal at the right-most end of the screen). However, there are many behaviors that are produced which indirectly related to game main objective.

	Directly concerned with main objective	Indirectly concerned with main objective
Inside a game	(1) Normal play	(3) provoking, reminding, etc.
Outside a game	(2) Bluff, Skype, etc.	(4) Screaming, leaning, etc.

**Table 1.** Human player's action categories

Shiratori and colleagues showed another aspect of human-like behavior: communication outside the game, such as through facial expression and bluffing [8]. This behavior is important in some board games or card games such as poker and mahjong. They presented computer players in a fighting game with facial expressions outside the game that match the in-game state. Such "outside game" actions are related to a game's entertainment value, similar to shouting when a game character is being attacked ((2) in Table 1).

Current human-like behavior or believability research is focused on action in-game or outside the game that is mostly based on the intention to clear the game's main objective. However, human players also take some actions with intentions that are indirectly related to the main objective of the game. For example, outside the game, human players might scream when their characters get attacked in-game or may lean in the direction of their characters when turning in a racing game. Human players often take these unnecessary actions in order to immerse themselves in the game environment.

In-game, human players might show behaviors with intentions other than to clear the main objective ((3) in table 1). For example, some players make illustrations using bullet holes in FPS games. These in-game actions are used for reasons other than reaching the game objective, for purposes such as provoking, reminding, or warning the other player (for example, punching when not in attack range to provoke the opponent in a fighting game).

We created Table 1 to briefly explain the behavior of human players in response to a game. These behavior have been widely discussed in the field of games and culture; for example, Tylor studied the behavior of players in the game World of Warcraft [12]. However, in the study of computer gaming AIs, these behaviors still receive less attention, especially (3), which is the main subject of discussion in this article.

### 3 Classification of Human Behavior Not Directly Related to The Game's Main Objective

In section 2, four types of action were presented. When playing a game, human players do not only aim to clear the game's main objective (such as getting high scores, clearing the stage, or defeating the enemy), but they also divert game actions for purposes not directly related to the game's main objective. In a game where cooperation with another player is necessary, some actions might be used to transmit a message, such as a notification or warning about something, or to provoke an enemy.

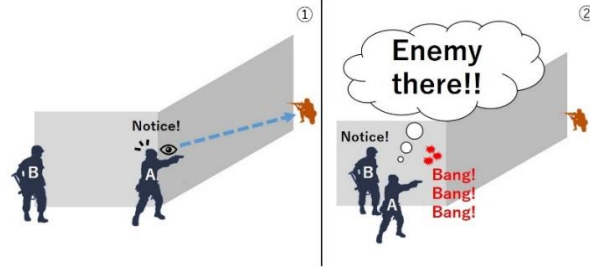
For human players, it is also possible to notify or warn about something with natural language by using an in-game chat system or VOIP programs such as Skype. However, in some situations, these communication channels are unavailable (e.g., chat is not available in the game or "too busy to chat, chat is difficult"), and in that case, actions inside the game itself can be diverted from their original use and used to represent different meanings or intentions.

In this chapter, 50 typical cases are introduced where human players seem to select their actions not to win, but for another purpose. We have viewed many videos of human gameplay and selected 50 typical cases. We do not claim that they are a complete or representative set of such behaviors; there might be other interesting and important examples, but we believe these 50 cases are valuable to show. These cases were then grouped into seven classes according to the purpose of the action, such as warning, provocation, or greeting. Some of these action types, such as the warning type, are highly related to the main objective of winning, and some of them, such as greeting, are less related. The following subsections explain the seven types of indirect action, from highly-related ones to less-related ones.

#### 3.1 Warning

In cooperative games, warning is important for developing a strategy to clear the main objective of the game. Vocal warnings, alarms, or simple signals are part of some games. However, in cases where these functions are not available or the player is unavailable at that time, players often use other actions to transmit their messages. Actions with the intention of warning are strongly related to the main objective of the game. For humans, these actions facilitate a feeling of cooperation, so it is important for game-AI to produce and understand such actions in order to increase the satisfaction of human players with game-AI. We show two study cases of warning actions.

- Study case <1> (MOBA: League of Legends): Warning a team member about an incoming enemy or enemy action by using the "?" mark available in the game itself instead of the in-game chat, which can be used but consumes more time.
- Study case <2> (FPS: Sudden Attack): When the player notices a sniper, he/she shoots the nearest wall or corner in order to warn allied players (Fig. 1).

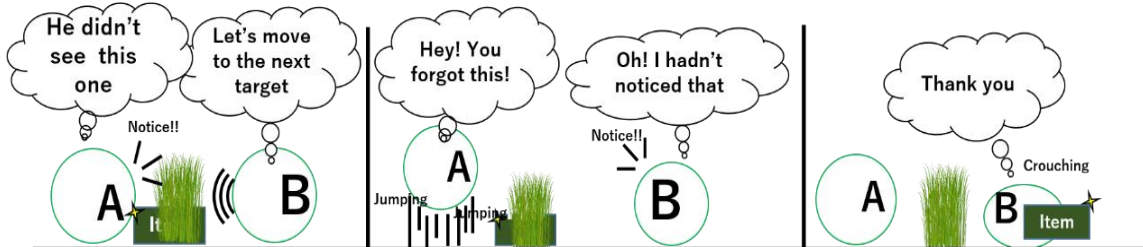


**Fig. 1.** Warning action in FPS Game: (1) Player A moves out from the corner of the building and finds an enemy, then (2) Player A tries to warn player B by shooting the nearest corner.

### 3.2 Notification

A notification action is defined as an action where the intention is to tell something to an opponent or ally, such as “Let’s start the match!”, “Please surrender!”, or “Hey, come here!” Notification actions are strongly related to the game’s main objective, thus the implementation of them in game-AI might be not difficult. The following study cases show some examples of the notification action.

- Study case <3> (MMO: Maple Story): To notify another player that there is a forgotten item on the floor, the player jumps repeatedly over an item with one hand over the item or in the direction of the other player. The intent of this action is to tell another player “there is an item here” and “please pick it up hurry” (Fig. 2.).
- Study case <4> (Action fighting: Super Smash Bros): In the character selection lobby in online matching, the match will start when all players press the ready button. In this phase, it is possible to press ready and cancel repeatedly to notify another player to “hurry up.”
- Study case <5> (MMO: Dungeon & Fighter): In a dungeon while the team is co-operating, using attack actions at the door or passage that a player wants to explore indicates the destination to the other teammate.
- Study case <6> (Board Game: Go): In a situation where one player has an advantage on the other player, a clearly suboptimal move is chosen on purpose in order to transmit “I can beat you even if I choose this kind of suboptimal move. Surrender now!”

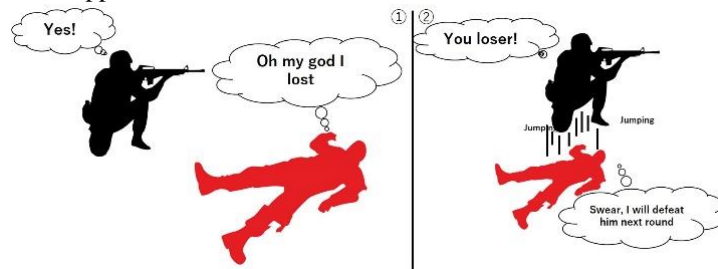


**Fig. 2.** Notification action in MMORPG game: (upper) Player A finds an item that Player B didn’t notice; (mid) Player A jumps repeatedly so that Player B notices the item; (lower) Player B picks up the item and expresses his gratitude by crouching.

### 3.3 Provocation

Provocation (or “trolling”) is an action that tries to frustrate the opponent when the player is in an advantageous situation; this gives some small impediment to the opponent with malice, or a player might put him/herself at disadvantage on purpose. This action often occurs when a player is able to keep superiority in the game continuously. Normally, human players do such actions in order to satisfy themselves. However, sometimes, the goal of provoking or trolling is to lure the opponent into a mistake and might be strongly related to the game’s main objective. Reproducing such actions with Game-AIs might not increase human players’ satisfaction, but these actions are important for human-likeness.

- Study case <7> (FPS: Call of Duty): Moving, jumping, and crouch-standing repeatedly around a defeated opponent character’s (dead body) location to aggravate the defeated player (Fig. 3.).
- Study case <8> (Fighting game: Super Street Fighter II): In fighting games, after a round is finished, a player can punch or kick the dead body of the losing player to provoke the opponent.



**Fig. 3.** Provoking action: In an FPS game, moving, jumping, and crouch-standing repeatedly around a defeated opponent character to provoke the enemy.

### 3.4 Greeting

Greeting refers to an action where players communicate something like “Hello”, “Nice to meet you”, “Thank you”, or “My bad” (this category often includes apologizing). Normally, greeting (or apologizing) is done via VOIP programs such as Skype or built-in chat systems in the game. However, in some cases when chat or VOIP are not available, actions in-game are used to express these sentiments. Greeting has a weak relationship with the game’s main objective. In action games where a crouch action is available (normally for evasion from an attack), it is often used to perform greeting or apologizing.

- Study case <9> (Action fighting: Super Smash Bros): Players use crouch-standing repeatedly to express “Nice to meet you” when creating a team battle. The meaning of an action can change depending on when it is performed. After doing something considered as bad manners, a player can apologize to other players by crouch-standing repeatedly.

### 3.5 Expressing Empathy

Expressing empathy refers to actions that expect some response from an opponent or are used to provoke some action from an allied player; these express a “Let’s have fun together” feeling. These actions are done without malice.

- Study case <10> (Fighting game: Super Street Fighter II): Some players enjoy using attack actions or jumping outside attack range for fun, expecting the opponent to do the same thing in response.
- Study case <11> (MMORPG: Final Fantasy XIV): Sometimes, mass numbers of players come together and try to use in-game actions called “emotes” to express some movement or dance at the same time.

### 3.6 Showing Off

Showing off is an action based on appearances rather than performing to meet the game’s objective. Sometimes, this action might be conducted in order to provoke an opponent, but many players try to perform this type of action seriously.

- Study Case <12> (3D Fighting: Soul Caliber, Fate/Stay night) In some fighting game, the combo (series of action) which difficult to perform, afford cost is higher than performance (damage), but the appearance is good, such combo is exist. Some player tries to perform such showoff combo.
- Study Case <13> (Street fighter III) Counter Attack or Blocking are existing in many fighting game. Using such attack allowed player to who encounter the attack avoid and strike back without taking any damages. However, blocking action has to be perform suddenly after opponent perform attack which is very difficult.

### 3.7 Self-Satisfaction

Self-satisfaction actions include actions taken to pursue curiosity or to bind or constrain play. To bind play is to play the game with extra rules stated by the player him/herself, such as clearing the game at a low level, limiting item uses, or clearing a level or area without damage. Bind play is also performed for creating new styles of play.

Another type of action that players take to satisfy themselves is a creativity action. In games with a high degree of freedom such as Minecraft and Mario Maker, players can try to create innovative stages in their own style.

- Study case <14> (Action RPG: The Elder Scrolls V: Skyrim): Some players might enjoy exploring locations in the game that are normally hard to reach, such as the top of a mountain.
- Study case <15> (Action: Resident Evil): Some players implement extra rules such as “clear with limited equipment or weapons” or “stay alive at a low level of life or hit point throughout the game.”



#### **4 Appearance Conditions of Actions Not Directly Related to The Game's Main Objective**

In the previous chapter, we provided examples of study cases and classified them in seven categories. However, the conditions that cause this behavior to appear are very significant.

In-game behavior that is indirectly related to the main objective can be observed in many types of games. However, some of these actions require minimum knowledge of or skill at the game or the type of game in order to be interpreted correctly. For example, in study case <7>, the action of moving around the dead body of a defeated player might not be understood by a beginner. However, along with the improvement of player skill, action comprehension becomes deeper. When reproducing such actions in a computer player's behavior, it is necessary to take into account the skill and knowledge of the human player.

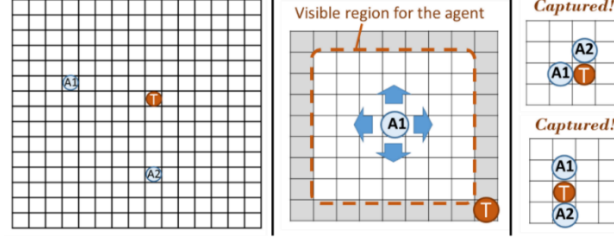
Limiting the information that players have affects their comprehension of a behavior's intention. For example, in card games, we can help or impede an opponent by discarding a card. However, if the opponent only knows that a card has been discarded and not for what purpose it can be discarded, it may be difficult to distinguish the other player's intention. Thus, when information is limited, these actions rarely appear. On the other hand, in games with complete information such as fighting games (where health, time, and/or a special attack energy gauge are shown), the intention of actions is easier to understand.

However, in some kinds of games, even when information is limited, human players can estimate the situation by using action information. For example, in FPS games, if enemy attacks suddenly stop or weaken, then the player might assume that opponents have changed their strategy. In this situation, the player is able to fill in the missing information and estimate the intentions of actions.

Another factor is the amount of spare time a player has in game, which relates directly to the difficulty of the game and the situation in game. In a game in which chat is allowed and the player has spare time to use a chat, the game's main objective may become temporarily irrelevant while this action is performed. The degree of freedom of action regarding game tasks directly affects the appearance of this type of action. In games with busy tasks or games where every action in-game affects the score or victory such as Go or Tetris, this type of action will not be performed.

#### **5 Emergence of Actions Not Directly Related to The Main Objective**

Most of the actions we introduced in this paper are unique to humans, though it is possible for some of them to emerge from systems without humans, as we mentioned in section 4-5. Thus, we carried out two experiments to observe how actions not directly related to the game's main objective emerge from interactions between reinforcement learning agents.



**Fig. 4.** Overview of the environment. Two agents try to catch a target. Each agent has a limited view. In this case, when both of the agents touch the target, the task is accounted a success.

### 5.1 Setting

The two experiments share a common setup. Two reinforcement learning agents with limited views try to catch a target while co-operating with each other, as illustrated in Fig 4. We expected these (limited sight) agents to substitute their sequential movement actions for a signal that the target is located near the agent.

- **Environment:**
  - The field consists of 15x15 grid spaces in which agents can locate the target.
  - Two agents and one target are randomly arranged on the grid initially.
  - Each agent can move to an adjacent grid space in a compass direction each turn.
  - The field has a torus structure. Agents will appear in the right-most when they go left on the left-most space and appear on the top side when they go down on the bottom of the grid.
  - More than one character (agent or target) cannot be located in the same grid space at the same time.
  - The target does not move.
  - In the case that both agents are located in grid spaces adjacent to the target before 100 turns passes from the initial state, the search and chase task is accounted a success. The task is accounted a failure otherwise. In both cases, the game state will be re-initialized.
- **Agent:** Each agent decides its action using a one-step Q-learning algorithm. The game state observed by each agent is a combination of feature values as below.
  - (F1) Coordinate of the target relative to the agent. The agent can find the target only when it is located inside the 7x7 grid area whose center coordinate is the agent; therefore, 49 values are possible for this feature.
  - (F2) Coordinate of the other agent relative to the agent. The limited eyesight of agents does not affect this information; thus, 224 ( $= 15 \times 15 - 1$ ) values are possible for this feature.
  - (F3) The number of turns during which the agent does not see the target. Once the agent finds the target, this value is set to zero.  $\{MaxT + 1\}$  values are possible for this feature, where  $MaxT$  is a parameter value of the agent (if the number of such turns becomes greater than  $MaxT$ , this feature value is set to  $MaxT$ ).
  - (F4) The last  $MaxH$  actions taken by the other agent, where the  $MaxH$  is a parameter value. The number of possible actions an agent can take is five (go up, down, left, right, or stay), therefore, 5  $MaxH$  values are possible for this feature.

Our agents have two parameter values for observing game states,  $MaxT$  and  $MaxH$ , as stated above. Additionally, there are other parameter values related to the learning algorithm. The reward is 100 for reaching a terminal state by succeeding in the catching task, 0 for failure.

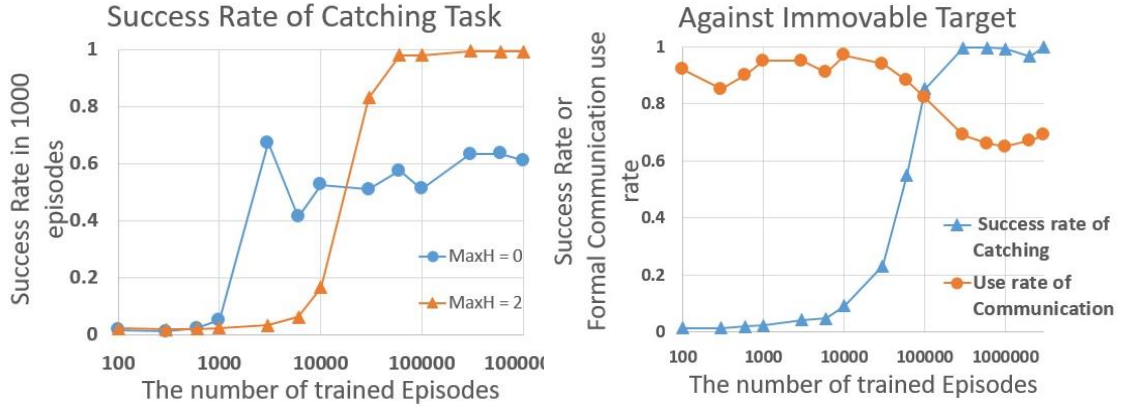
Discount factor: 0.8.

Learning rate:  $0.1 \times \frac{1000000}{1000000 + \{\text{Number of episodes trained}\}}$

The agent adopts the  $\epsilon$ -greedy policy with ( $\epsilon = 0.1$ ) as its behavior policy.

## 5.2 Experiment 1: Observing Emergence of Action Substitution

- Setting: We compared the movement patterns of agents under two parameter settings, that is  $MaxT=0, MaxH=0$  and  $MaxT=0, MaxH=2$ . In the  $MaxT=0, MaxH=0$  case, each agent must decide its action according only to the current positions of the other agent and the target (if it is located within eyesight). Therefore, agents cannot give their partner any clues to find out the target's location. On the other hand, in the  $MaxT=0, MaxH=2$  case, each agent is able to pay attention to the movement patterns of its partner. Therefore, agents have a chance of telling their partner the location of the target by showing the partner some characteristic movement patterns.



**Fig 5.** (left) Success rate of the task. The success rate was enhanced by 40% when each agent memorized the last two moves of the other agent. (Right) Success rate and the increasing use rate of the communication function to signal the immovable target. Through the whole training process, agents came to rely on the function at a rate of 70% (i.e., any agent used the function in more than 700 episodes out of the last 1000 episodes on each plot).

- Result: We observed the movement patterns in both settings after 3,000,000 training episodes. In the case of  $MaxH:0$ , each agent moved chaotically until the target came into sight. After that, the agent rushed at the target. In the case of  $MaxH:2$ , each agent used regular zigzag movement patterns for exploring (e.g., goes {up, left, up, left, . . .}) until the target came into sight. Whenever an agent found the target, it

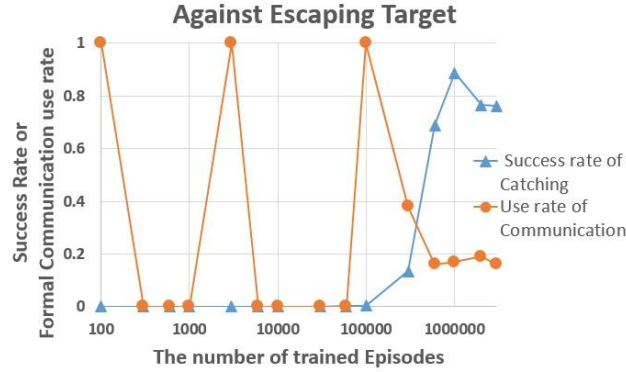
changed its movement pattern, avoiding moving away from the target (e.g., goes {up, down, up, down, . . .}) to inspire its partner to approach. Fig. 5. shows the performance of agents under these conditions. Introducing the information about the action history of the partner agent enhanced the success rate by 30% in the end, even though that information does not contain any direct clue for the target location. Therefore, we think the enhanced performance was caused by the emergence of an action substitution, that is, agents substituting their movement action to signal their partner.

### 5.3 Experiment 2: Encouraging Action Substitution

In the experiment described below, we aimed to show that even if there is a formal way to communicate with partners, agents prefer to communicate through action substitution if the situation is urgent and the formal method requires more time.

- Setting: We added two rule options for the system.
  - Escaping target: The target can also move in this rule. It moves away from agents once every four turns and in the case that any agent catches the target in its (limited) field of view.
  - Formal communication: Each agent can inform the other agent of the precise current location of the target at the cost of becoming immovable during the following four turns.

We compared the movement patterns of agents under two option settings: Escaping target: Off, Formal communication: On and Escaping target: On, Formal communication: On. In either case, the agent parameter setting is  $MaxT=21$ ,  $MaxH=2$ . That means agents can use both action substitution and formal communication to tell their partner the location of the target.



**Fig. 6.** Success rate and use rate of the communication function while chasing the escaping target. Eventually, agents began to avoid using the communication function in all but 20% of episodes.

Compared with action substitution, formal communication requires a larger number of turns to communicate and inform the partner of the more precise location of the target. Thus, the formal method imitates text chat or Skype in actual multiplayer video

game situations. In this experimental setting, we observed how the frequency of formal communication adopted by the agents varies.

- Result: The success rate of capturing the target and frequency of use for formal communication are shown in Fig. 5. (Left) and 5. (Right). In the Escaping target: Off case, the rate of use for formal communication is around 70%. We think this is because the communication method is useful in capturing the target. Meanwhile, in the Escaping target: On case, agents adopt the communication method in only 20% of episodes. We think the reason why is that the “capturing escaping target task” does not allow agents enough time for formal communication. On the other hand, the success rate of tasks after training is similar in both settings. This means that agents use action substitution more frequently in the scenario with an escaping target than with an immovable target, but they are able to use this method with comparable effectiveness to capture the target. Otherwise, the success rate would have largely dropped off in Fig. 6.

#### 5.4 Conclusion of Experiments

Experiment 1 showed the emergence of action substitution, in which movement actions are used as signals between agents. The agents obtained this method automatically through reinforcement learning, without any specific if-then routines for action substitution emergence. Therefore, we insist that agents in a system without humans can automatically obtain a type of in-game action that does not directly achieve the main goal (or at least an action pattern that appears to fall in such a category).

Experiment 2 demonstrated how the degree of urgency affects the probability of action substitution emergence. A higher degree of urgency makes agents less likely to use a formal method of communication and encouraged them to use substituted actions for their communication.

### 6 Conclusion

In this research, we showed new aspects of human-like behavior that it is possible to categorize in two sets of categories and in four ways: inside or outside the game, related or not directly related to the game’s main objective. We focused on actions without the intention to clear the main objective of the game, which we think are a significant behavior specific to humans and necessary to achieve human-like computer players. So far, 50 study cases of human actions were collected. These study cases were classified into seven types of behavior (i.e., warning, notification, provocation, greeting, expressing empathy, showing off, and self-satisfaction) and we discussed the occurrence conditions and possibility of reproduction by computer players. Furthermore, we conducted an experiment that shows the natural emergence of such behavior by learning between multiple Q-learning agents. This experiment successfully demonstrates the emergence of actions that are similar to the communication of humans.

### References

1. David Silver, et al, Mastering the game of Go with deep neural networks and tree search, Nature 529,484489 (2016)

2. Volodymyr Mnih, et al, Playing Atari with Deep Reinforcement Learning, NIPS Deep Learning Workshop 2013 (2013)
3. Volodymyr Mnih, et al, Human-level control through deep reinforcement learning, Nature 518, 529-533 (2015)
4. Soni, et al, Bots trained to play like a human are more fun, Neural Networks, 2008. IJCNN 2008 (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference, pp.363-369 (2008)
5. Kokoro Ikeda, Simon Viennot, Production of Various Strategies and Position Control for Monte-Carlo Go - Entertaining human players, IEEE Conference on Computational Intelligence and Games (CIG2013), pp. 145-152, (2013)
6. Togelius, J. et al, Assessing Believability Believable Bots: Can Computers Play Like People?, Springer Berlin Heidelberg, 215-230 (2012)
7. Fujii N., et al, Evaluating Human-like Behaviors of Video-Game Agents Autonomously Acquired with Biological Constraints. In: Reidsma D., Katayose H., Nijholt A. (eds) Advances in Computer Entertainment. Lecture Notes in Computer Science, vol 8253. Springer, Cham (2013)
8. Shiratori et al, 感情表現生成を行う仮想ゲームプレイヤー, The Society for Art and Science Vol. 7, No. 2, P65-74 (Japanese) (2008)
9. The 2k bot prize, <http://botprize.org/>, last accessed 2017/04/20.
10. <http://www.marioai.org/turing-test-track>
11. N. Shaker et al., "The Turing test track of the 2012 Mario AI Championship: Entries and evaluation," 2013 IEEE Conference on Computational Intelligence in Games (CIG), pp.1-8.(2013)
12. T.L. Taylor., "The Assemblage of Play" 2009 Game and Culture, pp. 331-339 (2009)
13. Turing, A. M. "Computing Machinery and Intelligence." Mind, vol. 59, no. 236, 1950, pp. 433-460.