ISSN: 2350-1022
International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 1, Issue 2, pp: (10-19), Month: October 2014 – March 2015, Available at: www.paperpublications.org

LANGUAGE ASSIMILATION THROUGH TYPE-2 FUZZY GRAMMARS: AN APPLICATION IN COMPUTER ENTERTAINMENT

Juan Paulo Alvarado-Magaña[1], Antonio Rodriguez-Diaz[2], Juan R. Castro[3], Olivia Mendoza[4], Oscar Castillo[5], Guillermo Licea[6]

[1,2,3,4,6]Autonomous University of Baja California, Faculty of Chemical Sciences and Engineering, Tijuana, Baja California, Mexico
[5]Tijuana Institute of Technology Tijuana, Baja California, Mexico

Abstract: **In computer entertainment and video games users interact with machine controlled agents often referred to as Non-Player Characters (NPC). An NPC is generally intended as an antagonist for the user, but frequently helpful NPCs designed to aid the player are employed. These types of NPCs are referred to as companions. This paper presents a novel method for improving companion NPC by way of Language Acquisition through the use of Fuzzy Set theory in the form of Type-2 Fuzzy Grammar, which is an extension of the canonical Fuzzy Grammar by Lee and Zadeh. Type-2 Fuzzy Grammar captures the ambiguity of natural language and can easily model the level of expertise with said language. A case study is presented where humans participate in a multi player Predator-Prey pursuit game. The text messages that players exchange during the game are used to teach the language to a companion NPC so that it may directly interact with the players to test its communication capability. Results show how the companion NPC acquired language and its impact on the game.**

Keywords: **Fuzzy Grammar, Fuzzy Language, Fuzzy Sets, Intelligent Agents, Language Acquisition, Natural Language Processing, Video Games.**

I. INTRODUCTION

The ability to communicate with machines through natural speech is an important milestone in man-machine interaction. A more ambitious goal is to have the computer learn a language by being exposed to it, similar to how a child would learn to speak.

Mechanisms, such as joint attention [1], contrast [2], and corrective feedback [3] have been used by researchers to give the machine the means to understand language by being constantly exposed to it [4] [9] [10] [11] [22].

The case was made that in order for a robot to communicate correctly with a human its speech must be based on grammar [5]. Grammar can provide the means to categorize words, segment a sentence properly, and even offer a certain level of prediction, thus granting the robot a way to "finish" another person's sentences and anticipate what is required of it.

In computer entertainment, video games in particular, companion agents are frequently used to enhance the play experience. They are computer controlled Non-Player Characters (NPC) that provide assistance to the player. This paper presents the initial results of a case study consisting of training a Companion NPC with messages used by players participating in a cooperative multiplayer game.

Fuzzy grammar [6] is used to teach language to a Companion NPC. To obtain a Fuzzy Grammar a canonical formal grammar is fuzzified by treating each production rule as a member of a fuzzy set. Each rule is assigned a degree of membership, thus giving certain rules higher membership than others which in turn grants more flexibility to the resulting fuzzy language. In order to capture the language expertise of our Companion NPC, we introduce Type-2 fuzzy grammar. Much like Type-2 fuzzy sets, the production rules in Type-2 fuzzy grammar have a secondary membership to capture uncertainty.

By combining the formality of a classic grammar with the flexibility of fuzzy sets, it is our goal to create a Companion NCP that can learn language from the players it interacts with and adjust it accordingly by manipulating the degrees of membership of each production rule. The Companion NPC will have the ability to learn language from many sources and even adapt colloquialisms of a particular group without sacrificing formal definitions. Furthermore, fuzzy grammar gives it the ability to quickly resolve ambiguity and avoid misunderstandings with the players.

Type-2 fuzzy grammar is applied in a case study, a cooperative multiplayer game based on the Predator-Prey Pursuit Game described in [7]. A Companion NPC uses InductiveCYK [8] to deduce the original set of production rules of the language so that it may change their grades of membership through gameplay.

This paper shows the initial results of a project which will conclude with an agent robust enough to learn language by interacting with humans. The agent will learn and adapt its language due to the flexibility of fuzzy sets; it will also achieve a form of multilingualism as a result of the ability Fuzzy Grammar has of combining multiple rules from different languages through grades of membership. This ability will also allow the agent to adapt colloquialisms from users around it and apply them in conversations without the limitation of having to break hard grammar rules.

This paper is organized as follows. Section 2 presents previous work done in language acquisition, in which work with the goal of developing a machine capable of adopting language (artificial in some cases [7] [9] [10] and natural in others [5] [11] [12] [13]) is reviewed. Section 3 defines the Predator-Prey Pursuit Game; a simulation that was used by Jim [7] to measure the effect language has on the success rate of communal problem solving. The game is modified for this paper's case study. Section 4 details the definition of fuzzy sets, which is the theory that drives Fuzzy Grammar suggested by Lee and Zadeh [6] wherein a formal grammar is expanded by adding grades of membership to the production rules. Section 5 presents the multiplayer Predator-Prey Pursuit Game case study, in which the game definition presented by Jim is implemented as an instance where real human players take roles Predators attempting to capture a computer controlled Prey; the game is used to create a set of training sentences for a Companion NPC and it's also used to test Companion's efficiency with language. Finally, results are shared in Section 6.

II. PREVIOUS WORK

A. *Artificial Companions*

Safaripour et al. [13] propose a multi-layer framework that can accelerate the development of robots for assisted living; these companion robots improve the quality of life of elderly and disabled persons by helping them with day to day tasks or by monitoring changes in their health or environment.

Work was done in producing companions capable of carrying out conversations, for example, the "How Was Your Day" companion from [12] is capable of talking about typical work related issues by using a multi-modal architecture to handle verbal and non-verbal inputs and outputs.

Companions are also included in computer based learning systems as a way of improving the user's learning experience. In [14], the researchers create an agent called "DwestAgent" to test their proposed General Companion Modelling approach which is used to describe both domain and learning competencies, as well as behaviour as a peer tutor or pupil. Varying these aspects produce vastly different learning companions that can be applied to different learning models.

Companions are also used in leisure activities like travel. The work in [15] describes a method for integrating information from various sources to help users that require several means of transportation in order to get from point A to point B. The companion aids the user by planning and managing a trip and making the information available through a smart phone.

These are only a handful of examples of artificial companions applied for the improvement of a user's welfare. Similarly, the user experience in video games can be improved by implementing a Companion NPC that's contextually aware of it and others, such that its interaction with the user is satisfactory.

B. Artificial Companions in Video Games

Amongst most recent cases there's Elizabeth from Bioshock Infinite [16], she assists the user with exploration and combat, she is also a central character in the game's storyline and she reacts appropriately to the player's actions. An example of a Companion NPC that is less involved in the storyline is Dogmeat from Fallout 3 [17]. He is a pet dog that the player can obtain during the game. Dogmeat will help the player by finding hidden objects such as ammunition or food supplies and by fighting alongside the player. These companions, and others like them have limited interaction with the user, in other words, the user has limited control over them; many times this can lead to unwanted or destructive behavior from the Companion NPC.

Companions are also common in multiplayer games, although the interaction with the players is even more limited. In Titanfall [18], a multiplayer First-Person Shooter, two teams of six players each battle over control of the map. NPC soldiers populate the game map on both sides and they help with the flow of the game by pointing the player towards objectives. Although they do not pose a danger to the player, the player has to fight with them and against them in order to win a match. Titanfall's main characters are the Titans, giant armed robots that the player can pilot. The player can use the Titans in two ways, either taking direct control of them by entering the cockpit, or leaving them in an autonomous mode while they play the game as foot soldiers. While the Titan is in autonomous mode the player can have it guard an area (attack any enemy that enters a zone) or follow the player to defend him.

Likewise, in games such as League of Legends [19] and Dota 2 [20], Companion NPC help friendly players by attacking enemy objectives and other players or opposing NPC. In all of these examples, planning and executing a strategy is key to a team's success, this is done by careful coordination of the players through the use of in-game text or voice chat, which is why having Companion NPC that can understand the interaction between the players will lead to a more satisfying experience for those players.

C. Work on Machine Language Acquisition

Outside the realm of video game applications, researchers have already succeeded in teaching a machine to talk through constant exposure to language. In [5], researchers trained a childlike robot named Kaspar to attach meaning to words and holophrases; by exploiting joint attention with a human teacher, the robot can associate the teacher's words with visual and audio perceptions. Training is done using actual dialogues of mothers talking to their children; located in the Child Language Data Exchange System (CHILDES) corpora [21]. Thanks to the use of grammar, Kaspar can plan ahead and categorize words.

Another example is [11], where the authors present an experiment in which a robot learns the meaning of words and grammar through a method of action and effect. The robot is presented with different objects described by three characteristics: shape, colour, and size. During the discovery phase, the robot manipulates the objects to determine which actions are possible and to see the effect on the object while a person verbally describes what the robot is doing. The robot learns the language and represents it using a Bayesian Network. Once the discovery phase is over, a human instructs the robot to do certain actions with the objects and it attempts to perform them. The authors mention cases where ambiguity was detected in the instructions, and, in those cases, it was difficult to assess the actions of the robot objectively, which is why a human judge was required to determine if what the robot did was expected.

There is also research done in more abstract ways of language acquisition. These experiments use an artificial language that is only understood by the simulated participants, for example, the agents used by Jim and Giles in [7]. They use the Predator-Prey pursuit game as a case study. During a game, four predator agents are placed in a square grid with the goal of catching a single prey agent. These predators can only sense where the prey is and do not know where their predator companions are. In order to coordinate their attack, the predators must tell each other their positions. They do so by writing a string of ones and zeros to a message board. To determine the next move, once all messages have been posted, each agent reads all the strings and concatenates them into a single input that is passed to a finite state machine. In order to develop the language, each predator is encoded in a chromosome. The initial generation usually doesn't capture the prey, but as generations advance string lengths increase and improve agent performance. Jim and Giles found that in order to solve the problem, the language must grow to a certain size.

Another interesting example is [10] by Marocco, where the experiment has a group of robots evolve a language to solve a collective navigation problem. The experiment consists of four robots randomly placed in an area with two gray circles marked on the floor; the robots must find the circles and ensure that each circle has two robots in it. Each robot has a neural controller with the following inputs: eight infrared sensors to determine proximity to other robots, one ground sensor to detect the gray circle, and four sound sensors to detect signals from other robots. The neuronal controller outputs signals to two motors and sound device. The researchers identified four distinct signals that were autonomously established by the robots: signal A was used by robots outside the gray areas, signal B was used by robots alone in the gray areas, signal C by those that were accompanied in the gray areas, and signal D by those outside a gray area but that were approaching a robot within the gray area. What's interesting about this study is that the robot's fitness wasn't evaluated in terms of its communication capability; even so the community produced a complex communication system to solve the problem.

In [22] the authors expand the previous work by using just two robots such that both have to be in the zone at the end of the experiment. While in the previous experiment signal synchronization was achieved only when the robots were within target areas, the robots in this experiment also achieve synchrony when both were outside target areas. The authors conclude that the robots are presenting “joint attention" behaviour to equalize internal states. Joint attention is when the speaker and a listener establish focus on the same object usually through nonverbal means.

The case that mostly resembles a video game is the NewTies Project [9]. It is a research project that simulates the trials faced by early human communities in which these are overcome individually and collectively by agents that form the artificial society. During the simulation, many agents are placed in a virtual world. These agents are described by a collection of attributes that define how they will act during the simulation, for example, an agent with a high socialness attribute is more likely to seek out other agents to interact with. Each agent can perform many operations, including eating, talking, moving, mating, etc. A Decision Q-Tree determines the agent's behavior, and it is adjusted as the agent acquires new knowledge. The Decision Q-Tree specifies which action an agent will take in any given time step. Of particular importance to the study is the language game agents play when they decide to talk to each other, since having a society that develops a common language is an objective of the NewTies project. During a language game, communication is started by a speaker agent who selects a target object located in the virtual world to talk about. The speaker decides which concepts will describe the target object and constructs an expression that is then passed to the listener agent (the speaker can invent novel words and add them to its personal lexicon). Upon receiving the expression, the hearer will attempt to interpret the message's meaning or learn the meaning if an interpretation with its current lexicon is impossible [4]. After several simulation time steps, the agents will convergence on a shared language.

D. Discussion

In order to expand the usefulness of Companion NPC, a process by which a companion can actively learn the player's language similar to how a small child learns to speak is proposed. A companion capable of learning and adapting a language can react to the needs of the player and work to fulfil them. With enough advances, Companion NPC could even replace a player in case of network malfunctions. For example, in a Titanfall match, if one player suddenly disconnects from the game, then one team has an advantage in numbers. But if a Companion NPC has learned the team's language it could momentarily substitute that player and still carry out verbal commands. The proposed method of applying Type-2 Fuzzy Sets to grammar will give robustness and flexibility to the language, as well as bridging the gap between the rigidity of formal languages and the ambiguity of natural speech. A grammar model provides tools that make communicating with humans possible [5], such as categorization, segmentation, and planning. These tools can prove advantageous to a Companion NPC. For example, it could accurately predict what the players are attempting to communicate by analyzing incomplete utterances.

III. PREDATOR-PREY PURSUIT GAME

As a case study, a simple multiplayer game based on the Predator-Prey pursuit game by Jim and Giles in [7] was developed. The game consists of four Predators attempting to catch a Prey. Predators and Prey are placed randomly on a

ISSN: 2350-1022
International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 1, Issue 2, pp: (10-19), Month: October 2014 – March 2015, Available at: www.paperpublications.org

square grid (Fig. 1), each turn they take one step, either up, down, left or right. If the Prey cannot move because it is blocked by the Predators then the Prey has been caught and the Predators have won (Fig. 2).

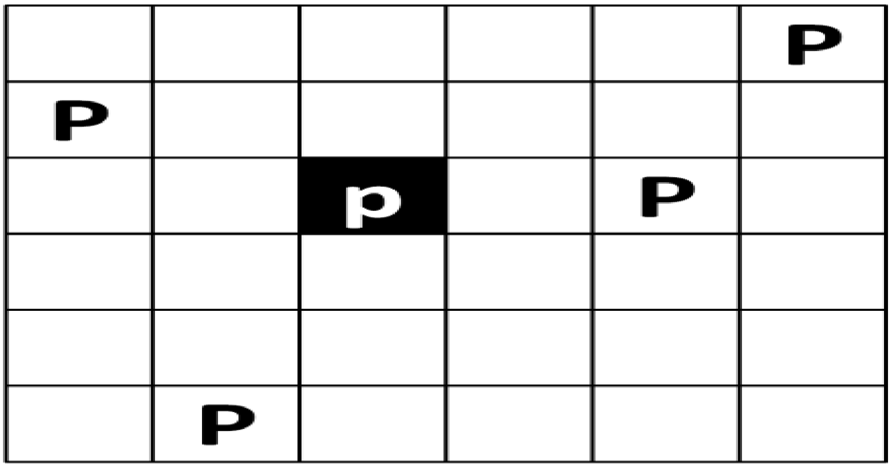

Fig. 1: Predator-Prey Pursuit Game

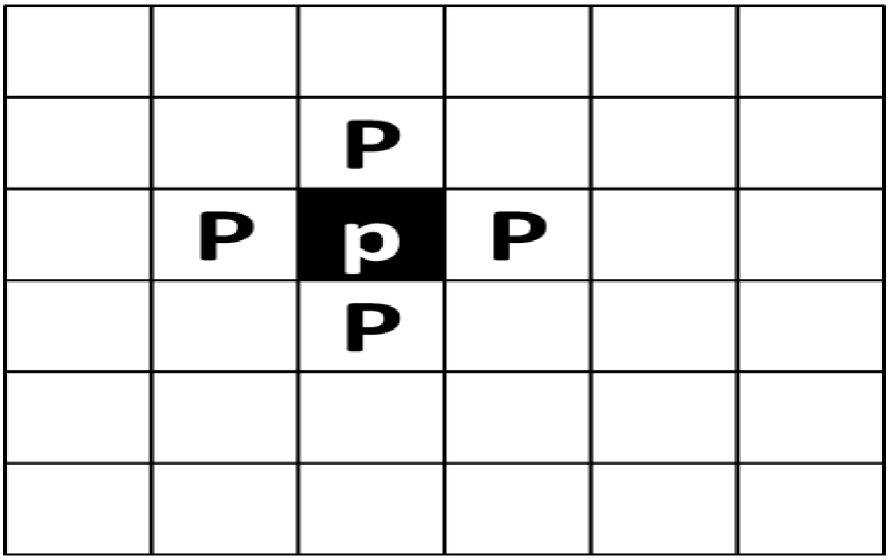

Fig. 2: Predator Win Condition

Jim and Giles add a restriction to the game in order to test the impact language has on Predator’s success rate. Each Predator only knows its own location and the location of the Prey, so in order for them to coordinate their attack they must tell each other these locations. Once messages have been passed, they make a decision and move to capture the Prey. Jim and Giles use intelligent agents in their simulation; in our work we use human participants in order to generate a training set of sentences.

Five human players take the role of Predators in a cooperative game. We simplify the decision process by adding a fifth Predator called the leader, the player taking the role of the leader will make the decisions for the whole group. The leader is not present on the game board, but he has complete knowledge of the positions of both Predators and Prey. His main task is to give instructions to the rest of the Predators. Similar to [7], the players controlling a Predator are not aware of the location of their peers, additionally they do not know the position of the Prey either.

The actions executed in a game can be seen in figure 3. During a single round, the leader analyses the current state of the game and makes a decision. Through the game's chat feature, he writes orders for the other players. Once the rest of the players receive the chat message they will either execute an action or pass the turn. After all players have acted, the computer controlled Prey takes its turn and the game state is updated. If the players have not found the Prey yet then the leader is required to make another decision. The game continues until the players win the game or a maximum number of rounds have elapsed. While the game is being played the messages sent through the chat client are being recorded in order to create a training set for an AIC.

IV. TYPE-2 FUZZY SETS AND FUZZY LANGUAGES

Zadeh introduced Type-2 fuzzy sets in 1975 [23] as an extension to fuzzy sets that permit the inclusion of uncertainty about the membership functions of traditional fuzzy sets. Uncertainty arises in different ways, e.g., it could be brought about by variance in measuring instruments or even by subjective bias of the observer. At the cognitive level, uncertainty arises from the inherited vagueness and ambiguity of natural languages [24].

Normal fuzzy sets are used when determining the grade of membership of an element as one or zero is difficult. Likewise, Type-2 fuzzy sets are used if determining the grade of membership between zero and one is difficult. The membership function of Type-2 fuzzy sets is defined as follows [25]:

Definition 1. A Type-2 fuzzy set, denoted A', is characterized by a Type-2 membership function $\mu'_A(x,u)$, where $x \in X$ and $u \in J_x \subset [0;\ 1]$, such that:

$$A' = \left\{\left((x,u), \mu'_A(x,u)\right) \middle| \forall x \in X, \forall u \in J_x \subset [0,1]\right\} \tag{1}$$

Note that Type-2 fuzzy sets are defined by the primary and secondary membership function. The uncertainty in a Type-2 fuzzy set can be observed by a bounded region called the “Footprint of Uncertainty" or FOU, which is the union of all primary memberships as in Fig. 3.

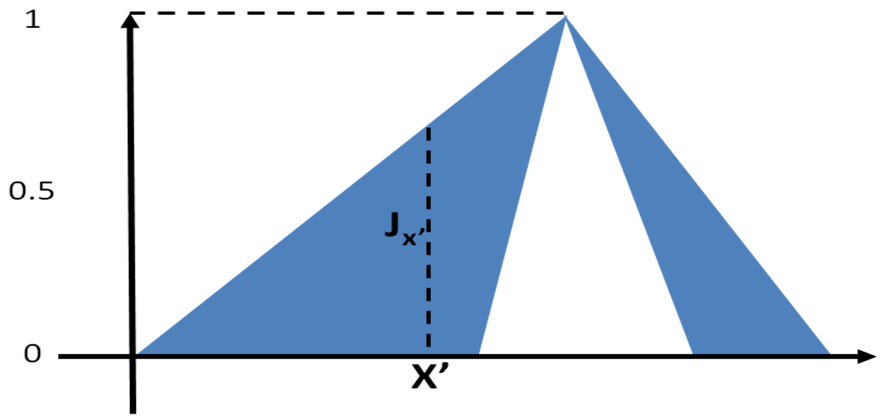

Fig. 3: Type-2 Fuzzy Sets

Fuzzy logic [26] [25] and machine learning [27] are just two examples of the applications of Type-2 fuzzy sets. In our work, we leverage the power of Type-2 fuzzy sets to capture the uncertainty of natural speech and to resolve ambiguity when it arises. We do so by extending Lee and Zadeh's fuzzy grammar [6] into a Type-2 fuzzy grammar as follows:

Definition 2. A Type-2 fuzzy grammar is a quadruple $G = (V_N;\ V_T\ ;\ P';\ S)$ in which V_T is a set of terminals, V_N is a set of non-terminals, P' is a set of fuzzy productions, and $S \in V_T$ is the set of starting variables.

The elements of P' are all productions in the form:

$$\mu'(\alpha \to \beta) = \omega', \omega' = [0,1] \tag{2}$$

where $\alpha \to \beta$ express a re-writing rule, α and β are in $(V_N \cup V_T)$ and the interval ω' is the grade of membership of β given α. In the same way, a Type-2 fuzzy production where $\omega' = [0,0]$ is assumed to not be in P'.

A string of terminals x is said to be in the fuzzy language $L(G)$ if and only if x is derivable from the starting variable S. The degree of membership of x in $L(G)$ is given by

$$\mu'_G(x) = [\underline{\mu}_G(x), \overline{\mu}_G(x)] \tag{3}$$

$$= [supmin(\underline{\mu}(S \to \alpha_1), \underline{\mu}(\alpha_1 \to \alpha_2), \ldots, \underline{\mu}(\alpha_n \to x)), supmin(\overline{\mu}(S \to \alpha_1), \overline{\mu}(\alpha_1 \to \alpha_2), \ldots, \overline{\mu}(\alpha_n \to x))]$$

ISSN: 2350-1022
International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 1, Issue 2, pp: (10-19), Month: October 2014 – March 2015, Available at: www.paperpublications.org

The uncertainty of a string of terminals x is given by $\Delta(x)$ (which is equivalent to the FOU of a Type-2 fuzzy set)

$$\Delta_G(x) = \overline{\mu}_G(x) - \underline{\mu}_G(x) \qquad (4)$$

The following is an example of a Type-2 fuzzy grammar: Let $G = (V_N; V_T; P'; S)$ be a Type-2 Fuzzy Grammar where $V_N = \{A, B, C\}$, $V_T = \{move, right\}$, $S = \{A\}$ and the productions in P' are:

$$\mu'(A \to BC) = [0.75, 0.8]$$

$$\mu'(B \to move) = [0.6, 0.7]$$

$$\mu'(C \to right) = [0.8, 0.9]$$

Thus according to equation 3, μ'_G for the sentence "move right" is:

$$\mu'_G(move\ right) = [supmin(0.75, 0.6, 0.8), supmin(0.8, 0.7, 0.9)] = [0.6, 0.7] \qquad (5)$$

And applying 4, $\Delta_G(move\ right)$ is:

$$\Delta_G(move\ right) = 0.7 - 0.6 = 0.1 \qquad (6)$$

V. EXPERIMENTS AND RESULTS

The Predator-Prey pursuit game from [7] was modified by adding an agent that knows the location of the prey and the four predators. This agent does not participate directly in the game, but it coordinates the predators by making decisions and dispatching messages to them. When the predators receive the message they parse it with their internal fuzzy grammar and execute its orders.

The experiment consists of two parts. During the first part, five humans play as predators in a multiplayer cooperative game of Predator-Prey. As explained in the previous paragraph, one of them acts as a leader and communicates with his team via in-game chat. In order to obtain a large enough training sample, messages sent over ten different games were recorded.

During the second par t of the experiment a Companion NCP is trained using sentences from the previous phase. The new Companion has no prior knowledge of the players' language, so in order to construct a starting grammar the InductiveCYK algorithm [8] is used. It is important to note that depending on the training set used by InductiveCYK the resulting grammar has the possibility of being either over-generalized or over-specialized. In the case of the former the resulting grammar derives words that aren't part of the original grammar, in case of the later the resulting grammar can only produce the words used to train it. We mitigate this issue by exploiting the benefits provided by the grades of membership in fuzzy grammar. By increasing or lowering a production rule's grade of membership, the words derived from the use of said rule will likewise be strengthen or weaken. In addition, a Companion's proficiency with the language can be modelled by introducing uncertainty to the production rules through the use of a Type-2 membership function.

Once the Companion NPC has deduced a starting grammar, it replaces the original leader, and a new set of games are played. In order to win the game, the new coordinator must construct messages that the players can understand. The following pseudocode represents the operations performed by the Companion during the second set of games:

P_i for $i = 1..4$ are the four users playing as predators

$C = \{S', L', G'\}$ is the Companion with strategy S', lexicon L' and fuzzy grammar G'

T is the training set of words used to generate a starting grammar for C

$p = \{S''\}$ is the prey agent with strategy S''

$t = 0$ is the number of turns in the game so far

B is the game board, it holds the positions of predators and prey

$C \to G' =$ inductiveCY K(T)

while p has not been caught or $t < 60$ do

$message = C{:}decide(B)$
for $i = 1..4$ **do**
  $P_i.receive(message)$
  if Pi understood message **then**
    update B with P_i new position
    C reinforces the rules in G' used to produce $message$
    C reinforces the words in L' used in $message$
  else
    C weakens the rules in G' used to produce $message$
    C weakens the words in L' used in $message$
  end if
end for
$p.decide(B)$
update B with p new position
end while

During the second part of the experiment we tested ten levels of uncertainty to measure the impact it had on the success rate, with $\Delta_{G_i} = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ respectively. For each value of Δ_{G_i} ten games were played and the average amount of turns required to capture the prey was recorded (games that reach 60 turns were stopped). Fig. 4 shows the results.

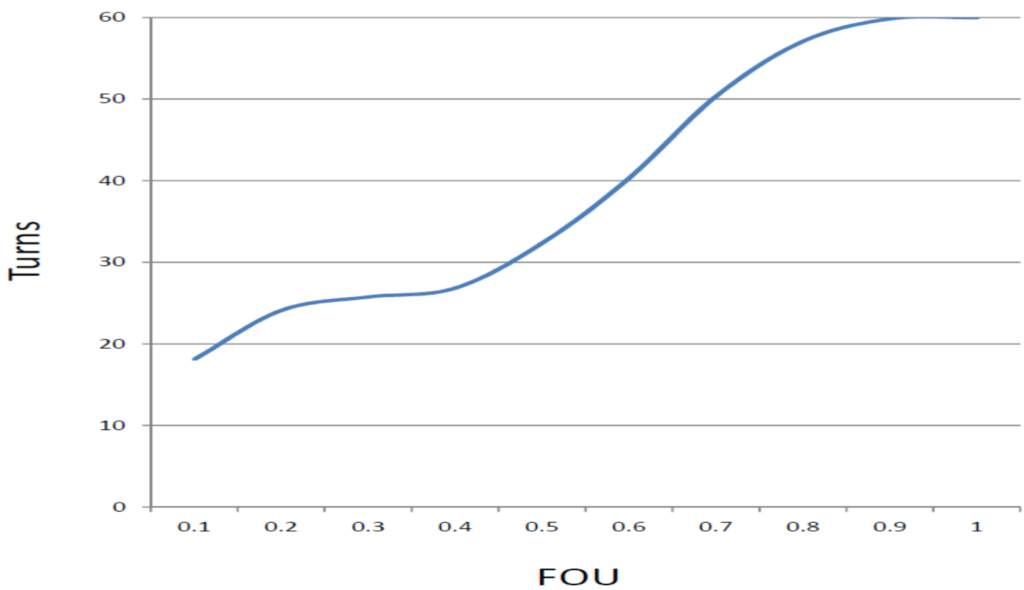

Fig. 4: Average turns to capture prey at each FOU level

An independent sample t-test was used in order to determine the effect uncertainty has on the success rate of the game. Each Δ_{G_i} sample was compared against the other samples to determine if there's any significant difference in performance. The *p-values* are shown in TABLE I, *p-values* greater than *0.05* imply an acceptance of the null hypothesis, in other words, no discernible difference between the samples is observed. Between $\Delta_{G_i} = 0.20$ and $\Delta_{G_i} = 0.30$ there is no difference, this can be appreciated by noting the low slope in Fig. 4 for that interval. It is also apparent that at very low levels of uncertainty, $\Delta_{G_i} = 0.10$ or less, the difference is significant when compared to other levels. The most notable difference can be observed between $\Delta_{G_i}= 0.50$ and $\Delta_{G_i}= 0.90$, where adjusting uncertainty can cause a noteworthy performance impact. Also, once uncertainty has reached higher levels, $\Delta_{G_i}= 0.90$ and above, communication brakes down in such a way that the game becomes unwinnable, at that point any change to the value of Δ_{G_i} will have a negligible effect on the success of the game.

VI. CONCLUSION

A review of the body of work regarding robots and intelligent agents assimilating language was reviewed. The lack of proper grammar models was observed and the prospect of contributing to the field by proposing a Type-2 Fuzzy Grammar method was identified. Type-2 Fuzzy Grammar was formally defined and its application was tested in a multiplayer scenario where an agent successfully assimilated the language used by people it interacted with. There's great interest in providing intelligent agents the means to learn a language in the same manner a child would. This approach will allow the creation of artificial companions capable of learning a language by changing the grades of membership in fuzzy grammar, giving it the flexibility to use improper or uncommon sentences initially and using proper sentences later. The flexibility of fuzzy grammar also gives the companions the possibility of mixing words from different languages, similar to how a multilingual person would. The secondary membership function in Type-2 fuzzy grammar allows us to model expertise with the language. As can be seen in the experimental results, as uncertainty with a language increases communication tends to break down. The efficiency of the Companion NPC was measured and validated and the results show that Type-2 Fuzzy Grammar provides a novel method to model language expertise and is a good tool to resolve ambiguity.

REFERENCES

[1] M. Tomasello, The Cultural Origins of Human Cognition. Harvard University Press, 1999.

[2] E. V. Clark, The Lexicon in Acquisition. Cambridge University Press, 1993.

[3] M. M. Chouinard and E. V. Clark, "Adult Reformulations of Child Errors as Negative Evidence", Journal of Child Language, vol. 30, no. 3, pp. 637-669, 2003.

[4] F. Divina and P. Vogt, "A Hybrid Model for Learning Word-Meaning Mappings," EELC, pp. 1-15, 2006.

[5] Lyon, Y. Sato, J. Saunders, and C. Nehaniva, "What is needed for a Robot to Acquire Grammar? Some Underlying Primitive Mechanisms for the Synthesis of Linguistic Ability", IEEE Transactions on Autonomous Mental Development, vol. 1, no. 3, 2009.

[6] E. T. Lee and L. A. Zadeh, "Note on fuzzy languages", Information Sciences, vol. 1, pp. 421-434, 1969.

[7] K. C. Jim and C. L. Giles, "Talking helps: Evolving communicating Agents for the Predator-Prey Pursuit Problem", Artificial Life, vol. 6, no. 3, pp. 237-254, 2000.

[8] K. Nakamura, "Incremental Learning of Context Free Grammars by Extended Inductive CYK Algorithm", in Proc. ECML Workshop Learn. Context-Free Grammars, 2003, pp. 53-64.

[9] N. Gilbert, M. den Bestn, A. Bontovics, B. G. Craenen, F. Divina, and A. Eiben, "Emerging artificial societies through learning," Journal of Artificial Societies and Social Simulation, vol. 9, 2006.

[10] Marocco and S. Nolfi, "Origins of Communication in Evolving Robots," in 9th International Conference on Simulation of Adaptive Behavior, 2006.

[11] G. Salvi, L. Montesano, A. Bernardino, and J. Santos-Victor, "Language Boot-Strapping: Learning Word Meanings from Perception - Action Association," IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, vol. 42, no. 3, 2012.

[12] Benyon, B. Gamback, P. Hansen, O. Mival, and N. Webb, "How was your day? Evaluating a Conversational Companion," Affective Computing, IEEE Transactions on, vol. 4, no. 3, pp. 299-311, July 2013.

[13] R. Safaripour, F. Khendek, R. Glitho, and F. Belqasmi, “A Restful Architecture for Enabling Rapid Development and Deployment of Companion Robot Applications", in Computing, Networking and Communications (ICNC), 2014 International Conference on, Feb 2014, pp. 971-976.

[14] C.-Y. Chou, T.-W. Chan, and C.-J. Lin, “An Approach of Implementing General Learning Companions for Problem Solving", Knowledge and Data Engineering, IEEE Transactions on, vol. 14, no. 6, pp. 1376-1386, Nov 2002.

[15] K. Rehrl, S. Bruntsch, and H.-J. Mentz, “Assisting multimodal travelers: Design and Prototypical Implementation of a Personal Travel Companion," Intelligent Transportation Systems, IEEE Transactions on, vol. 8, no. 1, pp. 31-42, March 2007.

[16] Irrational Games, “Bioshock Infinite", [DVD-ROM], 2013.

[17] Bethesda Game Studios, “Fallout 3", [DVD-ROM], 2008.

[18] Respawn Entertainment, “Titanfall," [DVD-ROM], 2014.

[19] Riot Games, “League of legends", On-Line, 2009.

[20] Valve Corporation, “Dota 2", On-Line, 2013.

[21] Childes: Child language data exchange system. [Online]. Available: http://childes.psy.cmu.edu

[22] R. Uno, D. Marocco, S. Nolfi, and T. Ikegami, “Emergence of Protosentences in Artificial Communicating Systems," IEEE Transactions on Autonomous Mental Development, vol. 3, no. 2, 2011.

[23] L. A. Zadeh, “The Concept of a Linguistic Variable and its Application to Approximate Reasoning", Information Sciences, vol. 8, pp. 199-249, 1975.

[24] G. J. Klir and J. Wierman, Uncertainty-Based Information: Elements of Generalized Information Theory. Physica-Verlag, Heidelberg, 1998.

[25] J. Mendel, Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, Upper Saddle River, NJ, 2001.

[26] O. Castillo, P. Melin, and W. Pedrycz, “Design of Interval Type-2 Fuzzy Models through Optimal Granularity Allocation", Appl. Soft Comput, vol. 11, no. 8, 2011.

[27] J. R. Castro, O. Castillo, P. Melin, and A. Rodriguez-Diaz, “A Hybrid Learning Algorithm for a Class of Interval Type-2 Fuzzy Neural Networks", Inf. Sci., vol. 179, no. 13, pp. 2175-2193, 2009.