A Distributed Realtime Dialogue Architecture for Dynamically Learning Polite Human Turntaking

Gudny Ragna Jonsdottir, Kristinn R. Thórisson

Icelandic Institute for Intelligent Machines
Reykjavik University, Menntavegur 1, 101 Reykjavík, Iceland

Abstract

Giving synthetic agents human-like real-time turntaking skills is a challenging task. Attempts have been made to manually construct such skills, with systematic categorization of silences, prosody and other candidate turn-giving signals, and to use analysis of corpora to produce static decision trees for this purpose. However, for general-purpose turntaking and other skills that may be highly variable across individuals and cultures, a better solution would be a system that can learn such skills on-the-job. We are exploring ways to use incremental perception and machine learning to have an agent learn proper turntaking during interaction. We have implemented a listening/talking agent that continuously adjusts its turntaking behavior to its interlocutors based on incremental real-time analysis of the other party's prosody. The system works in a real-world setting, achieving robust learning in spite of noisy data. Results show performance to be close to a human's in natural, polite dialogue, with 20% of the turn transitions taking place in under 300 msecs and 50% under 500 msecs.

Keywords: Turntaking, Machine Learning, Real-time, Prosody, Incremental perception

1. Introduction

Fluid turntaking is a dialogue skill that most people handle with ease. To signal that they have finished speaking and are expecting a reply, for example, people use various multimodal behaviors including intonation and gaze [1]. Most of us pick up on such signals without problems, automatically producing information based on data from our sensory organs to infer what the other

participants intend. In amicable, native circumstances conversations usually go smoothly enough for people to not even realize the degree of complexity inherent in the process responsible for dynamically deciding how each person gets to speak and for how long.

Endowing synthetic agents with similar skills has not been an easy task. The challenge lies not only in the integration of perception and action in sensible planning schemes but especially in the fact that these have to be coordinated while marching to a real-world clock. Lack of temporal responsiveness is one of a few key components that sets current dialogue systems clearly apart from humans; for example, speech recognition systems that have been in development for over a decade are still far from addressing the needs of realtime dynamic dialogue [2]. In spite of moderate progress in speech synthesis and recognition, many researchers have pointed out the lack of implemented systems that can manage dynamic open-microphone dialogue (cf. [3, 4, 5]), that is, situations where a dialogue-capable system knows "instantly" when it is given the turn and where it can be interrupted at any point in time by the user, in a natural manner, and vice versa.

Although syntax, semantics and pragmatics indisputably can play a large role in the dynamics of turntaking, we have argued elsewhere that natural turntaking is partially driven by a content-free planning[1] system [6]. For this, people rely on relatively primitive signals such as multimodal coordination, prosody and facial expressions. In humans, recognition of prosodic patterns, based on the timing of speech loudness, silences and intonation, is a more light-weight process than word recognition, syntactic and semantic processing of speech [7]. This processing speed difference is even more pronounced in artificial perception, and such cues can aid in the process of recognizing turn signals in artificial dialogue systems.

In natural interaction mid-sentence pauses are a frequent occurrence. Humans have little difficulty in recognizing these from proper end-of-utterance silences, and use these to reliably determine the time at which it is appropriate to take turn – even on the phone with no visual information. Temporal analysis of conversational behaviors in human discourse shows that turn transitions in natural conversation take on average 0-250 msecs [8, 9, 1] in face-to-face conversation. Silences in telephone conversations – when visual cues

[1]We use the term "planning" in the most general sense, referring to any system that makes a priori decisions about what should happen before they are put in action.

are not available – are at least 100 msecs longer on average [10]. In a study by Wilson and Wilson [8] response time is measured in a face-to-face scenario where both parties always had something to say. They found that 30% of between-speaker silences (turn-transitions) were shorter than 200 msecs and 70% shorter than 500 msecs. Within-turn silences, that is, silences where the same person speaks before and after the silence, are on average around 200 msecs but can be as long as 1 second, which has been reported to be the average "silence tolerance" for American-English speakers [11] (longer silences are thus likely to be interpreted by a listener as a "turn-giving signal"). Tolerance for silences in dialogue varies greatly between individuals, ethnic groups and situations; participants in a political debate exhibit a considerably shorter silence tolerance than people in casual conversation – this can further be impacted by social norms (e.g. relationship of the conversants), information inferable from the interaction (type of conversation, semantics, etc.) and internal information (e.g. mood, sense of urgency, etc.). To be on par with humans in turntaking efficiency a system thus needs to be able to categorize these silences.

The motivation for the present work is to develop a conversational agent that can adapt its interaction behavior to dialogue in a short amount of time and learn to interact with (ideally) no speech overlap, yet achieving the shortest possible silence duration between speaker turns. This is addressed according to three principles. First, we want to use on-line open-mic and natural speech when communicating with the system, integrating continuous acoustic perceptions as basis for decision making. Second, we want to model turntaking with a higher level of detail than previous attempts, including incremental perception and generation. Third, we want to incorporate learning, allowing for adaptation to each person the system interacts with.

We model turntaking as a negotiation process with contexts that describe which perceptions and decisions are relevant/appropriate at any point in time, and thus they represent the disposition of the system at any point in the dialogue, e.g. whether we might expect a certain turntaking cue to be produced, whether it is relevant to generate a particular behavior (e.g. volume increase in the voice upon interruption by the other), etc. The system learns on-line to become better at taking turns in realtime dialogue, specifically improving its own ability to take turns correctly and quickly, with minimal speech overlap.

In our evaluation setup the agent conducts 10 consecutive interviews in three different conditions. 1) A closed (noise free) setup with a very consis-

tent interlocutor – another instance of itself ("Artificial"). 2) An open-mic setup (using Skype) where the system repeatedly interviews a fairly consistent interlocutor – the same human ("Single person"). 3) An open-mic setup (using Skype) with individual inconsistencies where the agent interviews 10 different human participants consecutively ("10 people").

The rest of this paper is organized as follows: First we review related work, then we describe our the theoretical underpinnings of the approach. Following this we detail the architecture and learning mechanisms. A description of the evaluation setup comes next, followed by the results, summary and future work.

2. Related Work

The problem of utterance segmenting for the purpose of proper turntaking has been addressed to some extent in prior work. Sato et. al [12] use a decision tree to classify when a silence signals to take turn. They annotated various features in a large corpus of human-human conversation to train and test the tree. The results show that semantic and syntactic categories, as well as understanding, are the most important features. These experiments have so far been limited to annotated data of a single, task-oriented domain. Applying their methods to a casual realtime conversation using today's speech recognition methods would inevitably increase the recognition time beyond any practical use because of an increased vocabulary – the content interpretation results could simply not be produced fast and reliably enough for making turntaking decisions [2]. Schlangen [13] has successfully used machine learning to categorize prosodic features from corpus, showing that acoustic features can be learnt. Traum et al. [14] have addressed the problem of utterance segmenting, showing that prosodic features such as boundary tones do play a role in turntaking. As far as we know, none of this work has been applied to real-time situations.

Raux and Eskenazi [15] presented data from a corpus analysis of an online bus scheduling/information system, showing that a number of dialogue features, including speech act type, can be used to improve the identification of speech endpoint, given a silence. The authors tested their findings in a realtime system: Using information about dialogue structure - speech act classes, a measure of semantic completeness, and probability distribution of how long utterances go (but not prosody) - the system improved turntaking latency by as much as 50% in some cases, but significantly less in others. This

work reported no benefits from prosody for this purpose, which is surprising given the many studies showing the opposite (cf. [16, 13, 17, 14, 18, 1]). We suspect one reason could be that the pitch and intensity extraction methods they used did not work very well on the data selected for analysis. The Gandalf system [17] also used prosody, adding measures of semantic, syntactic (and even pragmatic) completeness to determine turntaking behaviors, although data about its benefit from this for the purposes turntaking per se is not available. The major lessons that can be learned from Raux and Eskenazi, echoing the work on Gandalf, is that turntaking can be improved through an integrated, coordinated use of various features in context.

Prosodic information has successfully been used to determine back-channel feedback. The Rapport Agent [16] uses gaze, posture and prosodic perception, among other things, to detect backchannel opportunities. The J.Jr. system [19], is a communicative agent that could take turns in realtime casual conversation with a human. Although the system did not process the content of a user's speech the system relied on an analysis of prosodic information to make decisions about when to ask questions (i.e. take turn) and when to interject back-channel feedback, with good result. The system was based on a finite state-machine formalism, similar to the Subsumption Architecture [20]. This approach turned out to be difficult to expand into a larger, more intelligent architecture [17]. Subsequent work on Gandalf [17] incorporated mechanisms from J.Jr. into the Ymir architecture, which was built as a highly expandable, modular system of perceptors, deciders and action modules; this architecture has recently been used in building an advanced dialogue and planning system for the Honda ASIMO robot [21].

Bonaiuto and Thórisson [22] demonstrate a system of two simulated interacting dialogue participants that learn to exploit each other's multimodal behaviors (that is, modality-independent multi-dimensional behaviors) so as to achieve a "polite" interaction where minimizing speech overlaps and speech pauses is the goal, as could e.g. be considered to be the standard situation in amicable interactions between acquaintances, friends and family - an equivalent constraint as considered in the present work. The system shows that emergent properties of dialogue, pauses, hesitations, interruptions - i.e. negotiations of turn - can be learned in this system.

3. Theoretical Underpinnings

The architecture described below rests on three main theoretical pillars. The first is a distributed-systems perspective, the second relates to architectural software methodology and the third is an underlying theory of turntaking in multimodal realtime dialogue, outlined in [6], empassing negotiation as a key principle in turntaking.

Models of dialogue produced by a standard divide-and-conquer approach can only address a subset of a system's behaviors (and are even quite possibly doomed at the outset). This view has been presented in our prior work [23] and is echoed in other work on dialogue architectures (cf. [3]). Requiring a holistic approach to a complex system such as human realtime dialogue may seem to be impossibly difficult. Counter intuitively, in our experience, if we attempt to take a breath-first approach to the creation of complex architectures – where most of the significant features of the system are taken into account, the set of possible contributing underlying mechanisms will – be greatly reduced [24], quite possibly to a small finite set. A way to address the problem of building more complete models of dialogue is to take an interdisciplinary approach, bringing results from a number of sources to the table, at various levels of abstraction and detail. It is the use of levels of abstraction that is especially important for cognitive phenomena: Use of hierarchical approaches is common in other scientific fields such as physics; for example, behind models of optics lie more detailed models of electromagnetic waves [25].

When dealing with complex architectures exhibiting heterogeneous behaviors we must try to constrain the possible design space from the outset. A powerful way to do this is to build multilevel representations (cf. [24, 26, 27, 28]); this may in fact be the only way to get our models right when trying to understand complex systems such as natural human dialogue. Notice that the thrust of this argument is not that multiple levels are "valid" or even "important", as that is a commonly accepted view in science and philosophy, but rather, that to map correctly to the many ways subsystems interact in such systems they are a *critical necessity* - that, unless simulations are built at fairly high levels of fidelity, we cannot expect manipulations to the architecture at various levels of detail to produce valid results.

Following this line of reasoning modularity in the architecture is a highly desirable feature – this brings transparency and openness to the architecture, for the benefit of its developers. However, decoupling components results in

a more distributed architecture, which calls for non-centralized control. The kind of modularity and methodology one adopts is critical to the success of such decoupling. Many of the existing methodologies that have been offered in the area of distributed agent-based system construction (cf. [29, 30]) suffer from lack of actual use-case experiences, especially for artificial intelligence projects that involve construction of single-mind systems. We have built our present model using the Constructionist Design Methodology (CDM) [31] that helps us create complex multi-component systems at a fairly high fidelity level, without losing control of the development process. CDM proposes 9 iterative principles to help with the creation of such systems and has already been applied in the construction of several systems, both for robots and virtual agents (cf. [31, 32, 21, 33]). CDM assumes a relatively manual construction process whereby a large number of pieces are integrated, for example speech recognition, animation, planning, etc., some of which may be off-the-shelf while others are custom-built. As such systems have to be deconstructed and reconstructed often, CDM proposes blackboards as the backbone for such integration. This makes it relatively easy to change information flow, add or remove computational functionality, etc., even at runtime, as we have regularly done.

As far as dialogue management and turntaking is concerned, modular or distributed approaches are scarce. Among the few is the Ymir Turn-Taking Model, YTTM [6], a model of multimodal realtime turntaking. YTTM proposes that processing related to turn-taking can be separated, in a particular manner, from processing of content (i.e. topic). Echoing the CDM, its architectural approach is distributed and modular and supports full-duplex multi-layered input analysis and output generation, with natural response times (realtime). One of the background assumptions behind the approach, which has been reinforced over time by systems built using the approach [34, 21, 35], is that the realtime performance calls for incremental processing of interpretation and output generation.

The turntaking model presented here is an extended version of the YTTM. Turntaking is modeled as an agent-oriented negotiation process with eight turntaking semi-global contexts that define the perceptual and behavioral disposition of the system at any point in the dialogue, e.g. whether we might expect a certain turntaking cue to be produced. These contexts support in effect a distributed planning and control of both perception and action; the distributed learning scheme implements a negotiation-driven tuning of realtime turntaking behaviors. Further details on the assumptions behind

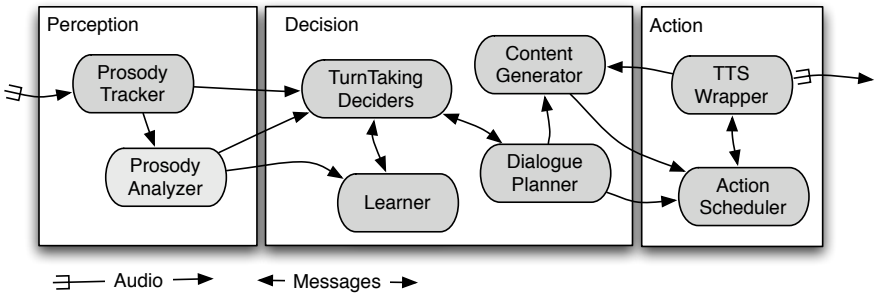

Figure 1: System components, each component consists of one or more modules.

this approach, and the research it is based on, are provided in [6].

4. System Architecture

We have built a multi-module dialogue system using the methodology described in [31, 23]. A flat view of the system's gross architecture can be seen in Figure 2 as an indication to the reader of the architecture's present scope, but here we will focus on the parts of the turntaking needed to support learning for efficient turntaking. Following the Ymir architecture [17], our system's modules are categorized based on their functionality; perception modules, decider modules and action modules, at the coarsest grain (see Figure 1). We will now describe the modules that relate to the turntaking system.

4.1. Perception

As already mentioned, although the architecture is inherently a multimodal system, the current system's input is limited to a set of prosodic features. There are two main perceptors (perception modules) in the system, the Prosody Tracker and the Prosody Analyzer. The Prosody Tracker is a low-level perceptor whose input is a raw audio signal [36]. It computes speech signal levels and compares them to a set of thresholds to determine information about speech activity, producing timestamped Speech-On and Speech-Off messages. It also analyzes the speech pitch incrementally (in steps of 16 msec) and produces pitch values, in the form of a continuous stream of pitch message updates.

Similar to [37], pitch is analyzed further by a Prosody Analyzer perceptor to compute a more compact representation of the pitch pattern in a discrete

Figure 2: Flat layout of message passing between modules.

state space, in our case to support the learning: The most recent tail of speech right before a silence, currently the last 300 msecs, are analyzed to detect minimum and maximum values of the fundamental pitch to produce a tail-slope pattern of the pitch. Slope is split into semantic categories, in the present implementation we used three categories for slope: *Up, Straight* and *Down* according to Formula 1 and three for the relative value of pitch right before silence: *Above, At, Below*, as compared to the average pitch according to Formula 2. The primary output of the Prosody Analyzer is a symbolic representation of the particular prosody pattern identified in this tail period (see Figure 3). More features could be added into the symbolic representation, with the obvious side effect of increasing the state space. Figure 3 shows a period of 9 seconds with speech periods, silences and categories. As soon as a silence is encountered (indicated by gray area) the slope of the most significant continuous pitch direction of the tail is computed 300 msec back in time.

$$m = \frac{\Delta pitch}{\Delta msecs}, \begin{cases} \text{if } m > 0.05 \rightarrow slope = Up \\ \text{if } (-0.05 \leq m \leq 0.05) \rightarrow slope = Straight \\ \text{if } m < -0.05 \rightarrow slope = Down \end{cases} \quad (1)$$

$$d = pitch_{end} - pitch_{avg} \begin{cases} \text{if } d > Pt \rightarrow end = Above \\ \text{if } (-Pt \leq d \leq Pt) \rightarrow end = At \\ \text{if } d < Pt \rightarrow end = Below \end{cases} \quad (2)$$

where Pt is the average ± 10, i.e. pitch average with a bit of tolerance for deviation.

4.2. Deciders

Our detailed turn-taking model consists of 8 dialogue states (see Figure 4). This represents the states taken when turn switches hands. The dialogue states are modeled with a distributed semi-global context system, implementing what can (approximately) be described as a distributed finite state machine that selectively applies to the activation and de-activation of (most) modules in the system. Context transition control ("state transitions") in this system is managed by a set of deciders [23]. There is no theoretical limit to how many deciders can be active for a given single system-wide context.

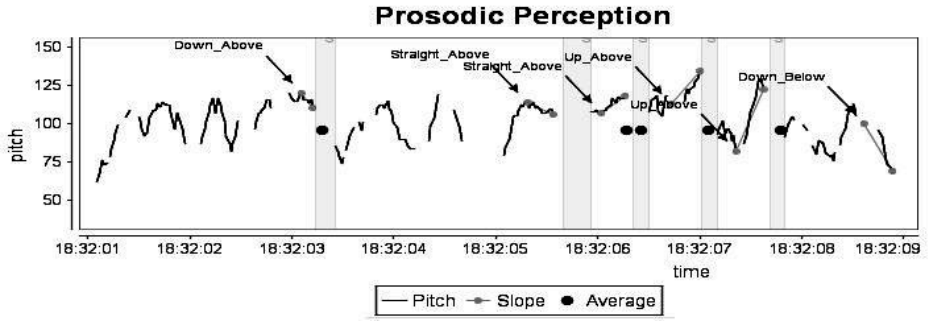

Figure 3: A window of 9 seconds of spontaneous speech, containing 6 consecutive utterances, categorized into descriptive groups for slope and end position relative to the average pitch. Only slope of the fundamental pitch during the immediate 300 msecs preceding a silence is categorized (into Up, Straight and Down). (Abscissa: Voice F0 in Hz, as produced in near-realtime by Prosodica; mantissa: Time - Hours/minutes/seconds.)

Likewise, there is no limit to how many deciders can manage identical or non-identical transitions. Reactive deciders (IGTD,OWTD,...) are the simplest, with one decider per transition. Each contains at least one rule about when to transition, based on both temporal and other information. Transitions are made in a pull manner; the Other-Accepts-Turn-Decider e.g. transits to context Other-Accepts-Turn (see Figure 4).

The Dialogue Planner (DP) and Learning modules (see further description below) can influence the dialogue state directly by sending context transition messages I-Want-Turn, I-Accept-Turn and I-Give-Turn; however, all these decisions are under the supervisory control of the DP: If the Content Generator (CG) has some content ready to be communicated, the agent might want to signal that it wants turn and it may want to signal I-Give-Turn when content queue is empty (i.e. have nothing to say). Decisions made by these modules override decisions made by other turntaking modules. The DP also manages the content delivery, that is, when to start speaking, withdraw or raise one's voice. The CG is responsible for creating utterances incrementally, in "thought chunks", typically of shorter duration than 1 second. We are developing a dynamic content generation system at present based on these principles the CG simulates its activity by selecting thought units to speak from a predefined list. It signals when content is available to be communicated and when content has been delivered.

In the present system the module Other-Gives-Turn-Decider-2 (OGTD-2) uses the data produced by the Learner module to change the behavior of the

Reactive deciders	
I-Have-Turn-Decider (IHTD)	Other-Has-Turn-Decider (OHTD)
I-Accept-Turn-Decider (IATD)	Other-Accepts-Turn-Decider (OATD)
I-Give-Turn-Decider (IGTD)	Other-Gives-Turn-Decider (OGTD)
I-Want-Turn-Decider (IWTD)	Other-Wants-Turn-Decider (OWTD)

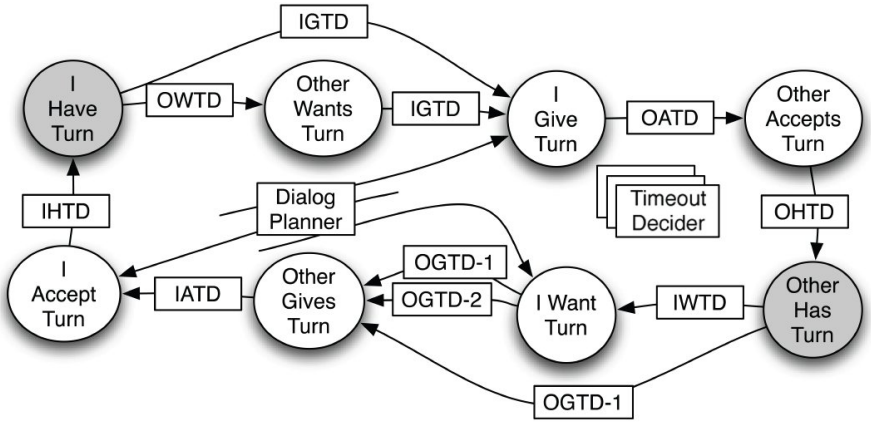

Figure 4: The heart of turntaking control in the system consists of a set of 8 semi-global context-states and 11 deciders. Each context has at least one associated decider for determining transition to it but each decider is only active in a limited set of contexts. In context-state I-Have-Turn, both I-Give-Turn-Decider (IGTD) and Other-Wants-Turn-Decider (OWTD) are active. Unlike other modules, the Dialog Planner (DP) can transition independently of the system's current context-state and override the decisions from the reactive deciders. A Timeout-Decider handles transitions if one of the negotiating contexts is being held unacceptably long (but it's transitions are not included in this diagram; also not shown are which modules are active during which contexts).

system. At the point where the speaker stops speaking the challenge for the listening agent is to decide how long to wait before starting to speak (OGTD-1 has a static behavior of transitioning to Other-Gives-Turn after a 2 second silence). If the agent waits too long, and the speaker does not continue, there will be an unwanted silence; if he starts too soon and the speaker continues speaking, overlapping speech will result. We solve this by having OGTD-2 use information about past prosody (right before latest silence) to select an optimal silence tolerance window, as will now be described in detail.

5. The Learner

The learning mechanism is implemented as a relatively independent component (Learner module) in the modular architecture described above. It is based on the Actor-Critic distribution of functionality [38], where one or more actors make decisions about which actions to perform and a critic evaluates the effect of these actions on the environment; the separation between decision and action is important because in our system a decision can be made to act in the future. In the highly general and distributed learning mechanism we have implemented any module in the system can take the role of an actor by sending out decisions and receiving, in return, an updated decision policy from an associated Learner module. A decision consists of a state-action pair: the action being selected and the evidence used in making that action represents the state. Each actor follows its own action selection policy, which controls how it explores its actions; various methods such as ϵ-greedy exploration, guided exploration, or confidence value thresholds, can be used [38].

In our system the Learner module takes the role of a critic. It consists of the learning method, reward functions, and the decision policy being learnt. A Learner monitors decisions being made in the system and calculates rewards based on a reward function, a list of decision/event pairs, and signals from the environment - in our case overlapping speech and too long silences - and publishes updated decision policy (the environment consists of the relevant modules in the system), which subsequently any actor module can use to base its decision on.

We use a delayed one-step Q-Learning method according to the formula:

$$Q(s,a) = Q(s,a) + \alpha[reward - Q(s,a)] \tag{3}$$

Where *Q(s,a)* is the learnt estimated return for picking action a in state s, and *alpha* is the learning rate. The reward functions - what events following what actions lead to what reward - need to be pre-determined in the Learner's configuration in the form of rules: A *reward* of x if *event* y succeeds at *action* z. Each decision has a lifetime in which system events can determine a reward, but reward can also be calculated in case of the absence of an event after its given lifetime has passed (e.g. no overlapping speech). Each time an action gets reward the return value is recalculated according to the formula above and the Learner broadcasts the new value.

In the current setup, Other-Gives-Turn-Decider-2 (OGTD-2) is an actor in Sutton's [38] sense that decides essentially what its name implies. This decider is only active in state I-Want-Turn. It learns an "optimal" silence tolerance window (STW) so as not to speak on top of the other, while minimizing the lag in starting to speak, given a silence. Each time a Speech-Off signal is detected OGTD-2 receives analysis of the pitch in the last part of the utterance preceding the silence, from the Prosody Analyzer. The prosody information is in turn used to represent the state for the decision; a predicted safe silence tolerance window is selected as the *action* and the Decision is posted. The end of the STW determines when, in the future, the participant who currently doesn't have the turn will start speaking (take the turn). In the case where the interlocutor starts speaking again before this STW closes, the decider doesn't signal Other-Giving-Turn, essentially canceling the plan to start speaking. This leads to a better reward, since no overlapping speech occurred. If he starts talking just after the STW closes, after the decider signals Other-Gives-Turn, overlapping speech will likely occur (keep in mind that, due to processing time, once a decision has been made it can take time before it is actually executed), leading to negative reinforcement for this size of STW given the particular prosodic information observed.

This learning strategy is based on the assumption that both agents want to take turns politely and efficiently. We have already begun expanding the system to be able to interrupt dynamically and deliberately - i.e. be "rude" - and the ability to switch back to being polite at any time, without destroying the learned data. This work will be reported at a later date.

6. Evaluation

We will look at system performance across three dependent measures:

- The system's ability to select an appropriate STW (Silence Tolerance Window). Given a silence in the user's speech, the selection of an STW is based on the type of prosody pattern perceived right before the silence. If turn-giving indicators are perceivable to the system we should find clear variations in STW lengths based on pattern perceived. If no evidence of turn-giving is detected by the system we should find an even distribution of STW size between patterns.

- How quickly the agent takes turn. We evaluate this by measuring the length of the silence before each successful turn-transition (from other to the agent) and compare the results to human data.

- Frequency of overlapping speech. Because the agent should be learning to be polite – i.e. not speak on top of the other – the number of overlaps should get reduced over time. (Note: In our Speaking-with-Self condition we use a closed sound loop (no open mic), but an open mic setup when the system speaks with humans).

6.1. Hypotheses and Statistics

To evaluate the learning mechanism we used linear regression on the single-case data sessions (Artificial - talking to itself for 10 consecutive sessions with 30 questions each; Single person - talking to one person for 10 consecutive sessions with 30 questions each). For the 10 person condition (asking 10 different people 30 questions each) we used a within-subject t-tests between the first 5 sessions and the second 5 sessions. In all cases the dependent variables are (a) Taking Turn in less than 500 msecs, (b) Taking Turn in less than 300 msecs and (c) Number of Overlaps.

The hypotheses are:

- H1: Frequency of taking turn within less than 500 msecs should increase as a function of number of turns.

- H2: Frequency of taking turn within less than 300 msecs should increase as a function of number of turns.

- H3: Frequency of overlapping speech should be higher in the first 5 interviews than in the second 5 interviews.

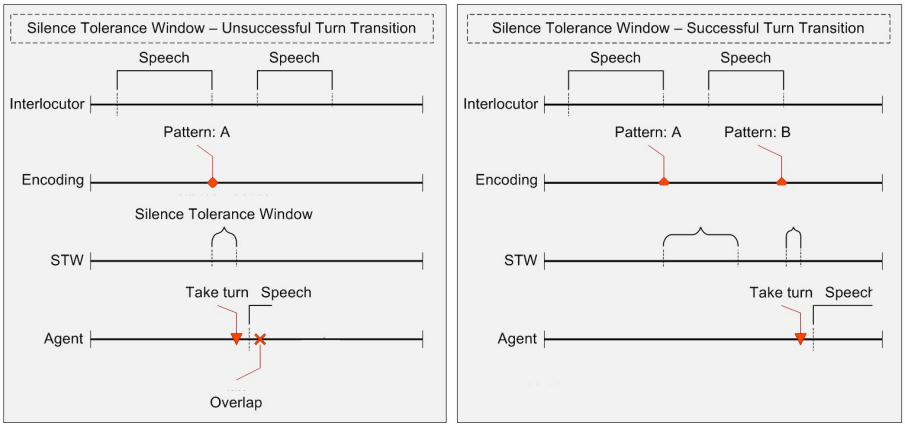

Figure 5: The interlocutor's speech is analyzed in realtime; as soon as a silence is detected the prosody preceding the silence is decoded. The system makes a prediction by selecting a Silence Tolerance Window (STW), based on the prosody pattern perceived in the interlocutor. This window is a prediction of the shortest safe duration to wait before taking turn: A window that is too short will probably result in overlapping speech while a window that is too large may cause unnecessary/unwanted silences.

6.2. Interview Setup

The agent is configured to ask 30 predefined questions, using among other things STW to control its turntaking behavior during the interlocutors' turn (see Figure 5). We have run three different evaluation conditions with the system. In all conditions the system is learning to take turn in a polite manner but still with the shortest silence between turns. Each evaluation consists of 10 consecutive interviews. Our system, named Askur, begins the first interview with no knowledge, and gradually adapts to its interlocutors throughout the 10 interview sessions.

The goal of the learning system is to learn to take turns with (ideally) no speech overlap, yet achieving the shortest possible silence duration between speaker turns. To eliminate variations in STW (Silence Tolerance Window) due to lack of something to say we have chosen an interview scenario where the learning agent is the interviewer, in which case it always has something to say (until it runs out of questions and the interview is over).

We are aiming at an agent that can adapt its turntaking behavior to dialogue in a short amount of time using incremental perception. In the evaluations we focus exclusively on detecting turn-giving indicators in deliberately-generated prosody, leaving out the topic of turn-opportunity detection (i.e.

turn transition without prior indication from the speaker that she's giving the turn), which would for example be necessary for producing (human-like) interruptions.

1. **The system interviewing itself ("Artificial").** Having an single artificial interlocutor interacting with a non-learning instance of itself gives us a very consistent behavior in a setup with no background noise.
2. **The system interviewing a single person ("Single person").** A single person should be fairly consistent in behavior, but some external noise is inevitable since the communication is through Skype.
3. **The system interviewing 10 people ("10 people").** This is the most complex condition, as there is both individual variation between participants as well as background noise.

A convenience sample of 11 Icelandic volunteers took part in the experiment, none of whom had interacted with the system before. All subjects spoke English to the agent, with varying amounts of Icelandic prosody patterns, which differ from native English-speaking subjects. The study was done in a partially controlled setup; all subjects interacted with the system through Skype using the same hardware (computer, microphone, etc.) but the location was only semi-private and some background noise was present in all cases.

6.3. Parameter Settings

The main goal of the learning task is to differentiate silences in realtime based on partial information of an interlocutor's behavior (prosody only) and predict the best reciprocal behavior. For best performance the system needs to find the right trade off between shorter silences and the risk of overlapping speech. To formulate this as a Reinforcement Learning problem we need to define states and actions for our scenario.

Using single-step Q-Learning the feature combination in the prosody preceding the current silence becomes the *state* and the length of the Silence Tolerance Window (STW) becomes the action to be learned. For efficiency we have split the continuous action space into discrete logarithmic values (see Table 1), starting with 10 msecs and doubling the value up to 1.28 seconds (the maximum STW where the system takes the turn by default). The action selection policy for OGTD-2 is ϵ-greedy with 10% exploration and always selecting the shorter STW if two or more actions share the top spot.

Table 1: Discrete actions representing STW size in msecs.

Action	Rewards	
(STW)	Successful transition	Unsuccessful transition
10	-10	-2000
20	-20	-2000
40	-40	-2000
80	-80	-2000
160	-160	-2000
320	-320	-2000
640	-640	-2000
1280	-1280	-2000

The reward given for decisions that do not lead to overlapping speech (successful transitions) is the milliseconds in the selected STW; a 100 msec STW will receive a reward of -100 if successful and STW of 10 msec -10 points. If, however, overlapping speech results from the decision (unsuccessful), a fixed reward of -2000 (same as waiting the maximum amount of time) is given. This is to simulate that when two STWs are without overlap the smaller is better. All rewards in the system are negative, resulting in unexplored actions being the best option at each time since return starts at 0.0 and once a reward has been given the return goes down (exploration is guided towards getting faster time than the currently best action, so a STW larger than optimal is not explored). In the beginning the agent is only aware of actions 1280 and 640 and only explores shorter STW's for patterns where the lowest available STW is considered the best.

7. Results

To reiterate, there are three main conditions, Artificial, Single person, and 10 people. First we will answer the question of whether the system is learning; then we will look at the above dependent measures in more detail.

7.1. Is the System Learning?

The system showed significant learning effects for Artificial condition, both for reaction time (simple regression F=12,83; p<0.0005) and overlaps (simple regression F=10,41; p<0,0047). The system also showed significant

learning effects for 10 person condition, for reaction time (see Table 2) and overlaps (see Table 3). Although 89 msec gain in STW does not seem a lot, the system starts each interview with previous learning and thus optimal STW based on another person's prosody patterns instead of beginning with a "safe" 1-2 second STW. To shorten this previous optimal STW at the same time as overlaps drop from 24% to 10% suggest that the agent is learning new skills on the fly, becoming increasingly more polite by improving its reaction time and speech overlap performance between – as well as within – interviews.

Table 2: Paired one-tail t-Test: Interviewing 10 consecutive people.

Turn	Observations (N)	Mean	St.Dev
Turn 1 - 15	10	655 msecs	137,25
Turn 16 - 30	10	566 msecs	73,83
T-value = 2,46, P-value = 0,018, DF = 9			

Table 3: Paired one-tail t-Test: Overlaps when interviewing 10 consecutive people.

Turn	Observations (N)	Mean	St.Dev
Turn 1 - 15	10	0,24	0,11
Turn 16 - 30	10	0,10	0,09
T-value = 4,16, P-value = 0,0012, DF = 10			

Although the results for Single person condition are indicative of the same trends as observed in the other conditions, they are not significant. It is possible that the Single person condition was somehow more noisy than the other sessions; after all we are only talking about 30 turns overall (a significantly lower number of trials than most reinforcement-learning paradigms require and thousands of trials faster than standard learning approaches using artificial neural nets), and the interviews were conducted over Skype, which is known to have highly variable noise levels depending on a number of factors. It is also possible that the person chosen had peculiarities in its prosody patterns, although this may be difficult to verify. We will investigate these factors further in future work.

7.2. STW by pattern

We look for turn-giving intonation patterns in the last 300 msecs of speech before each silence. Tail pattern of the pitch is currently categorized into 9

semantic categories based on slope (Up, At, Down) and final pitch compared to average (Above, At, Below).

To begin with we will analyze the distribution of these patterns before silences that mark end of turn, and before silences that are within turn. In both our artificial interviewee evaluation and single person evaluation the pattern *Down_Below* (representing a final fall in pitch) is most widely used at end of turn (see Table 4). This rhymes well with previous research [18] which has associated a final fall in pitch with a turn-giving signal. Furthermore, the person and the artificial interviewee have very similar distribution of patterns at the end of turn. The same cannot be said about prosody patterns perceived before silences that do not lead to turn transition. Prosody before silences within turn are much more evenly distributed between categories in the person's speech then in the artificial interviewee's speech. The artificial interviewee is as stated before very consistent, he decides what to say beforehand and sticks to that. After listening to the recordings of the person speaking there is a lot more variation going on, decisions being made and changed at the spur of the moment leading to more inconsistencies in prosody. An example of that is a person giving a short answer with prosody that can be perceived as giving turn and then adding to the answer and again ending with a give-turn prosody (e.g. "My favorite actor is Will Smith. and Ben Affleck.").

Table 4: Distribution between prosody patterns.

	Artificial interlocutor		Human interlocutor	
Pattern	At end	Within	At end	Within
Down_Below	58,6%	0,4%	42,0%	12,6%
Straight_Below	10,3%	0,1%	14,1%	17,2%
Up_Below	8,6%	2,4%	7,3%	3,6%
Down_At	8,4%	20,6%	10,4%	14,6%
Up_Above	5,6%	38,1%	5,0%	14,4%
Straight_At	2,7%	10,6%	6,5%	15,7%
Straight_Above	2,2%	13,7%	7,3%	9,7%
Down_Above	2,1%	10,2%	3,4%	4,6%
Up_At	1,5%	4,1%	3,9%	7,6%

When the agent interviews 10 consecutive people we analyzed which patterns were most widely used at end of turn. We found that 4 patterns out

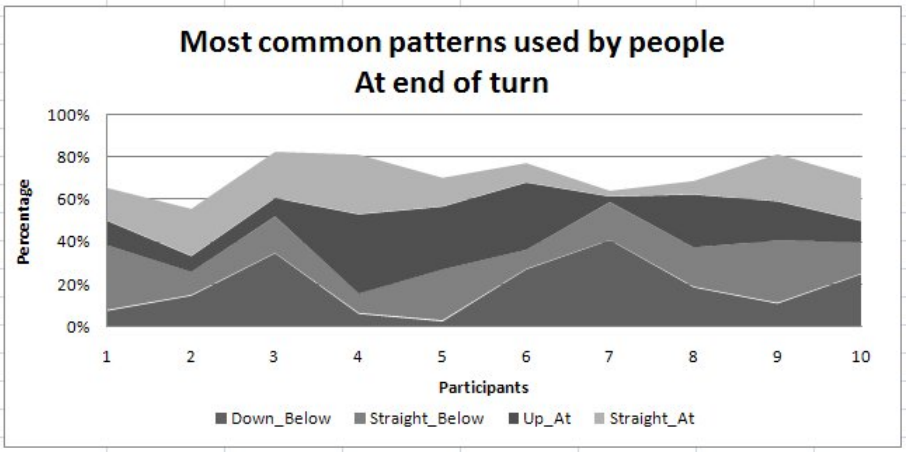

Figure 6: Four prosody categories out of nine are seen in up to 80% of turn-transitions before the agent takes turn.

of 9 are seen in up to 80% of turn-transitions (see Figure 6). None of these patterns have an end pitch ending above session average. This might be due to the fact that people are not asking the agent any questions – questions tend to end on a higher-than-average pitch.

We further analyzed the use of the final fall pattern *Down_Below* both as turn-transition and within turn. The use of final fall both at end of turn and within turn varies considerably between participants. The person that uses final fall the most at end of turn uses it in 41% at end of turn while the person that uses it the least only uses it in 2,7% of cases (see Table 5). This is surprising as the pool of participants are all from the same culture pool and we would thus speculate a more similar behavior.

7.3. Silence length

In a study by Wilson and Wilson [8] into human behavior, silences in face-to-face conversation where participants always had something to say was measured. In this study they found response time to be shorter than 500 msecs in 70% of turn-transitions and shorter than 200 msecs in 30% of turn-transitions. Our study is done over telephone (Skype) and not face-to-face and thus allows only for voice cues to communicate envelope feedback regarding turns. The studies are compatible in the sense that our agent always has something to say (while people might have to think a bit before they answer). Silences in telephone conversation tend to have average silences

Table 5: Usage of Down_Below in the 10 person study.

Participant	At end	Within
1	7,7%	14,9%
2	14,8%	7,3%
3	34,8%	6,7%
4	6,3%	9,1%
5	2,7%	7,1%
6	27,3%	15,4%
7	41,0%	8,7%
8	18,8%	5,0%
9	11,1%	2,5%
10	25,0%	19,2%

about 100 msecs longer than in face-to-face conversation [10] so we have measured silences shorter than 300 msecs and shorter than 500 msecs.

Our agent takes turn in less than 500 mescs in 53,1% to 43,7% of turns for our three conditions (see Table 6). This is the average over the last 9 interviews, skipping the first interview due to STW being preset to 1280 and 640 msecs in the beginning, which influences the data.

Table 6: Average silences for each condition.

Condition	Shorter than 500 msecs	Shorter than 300 msecs
Artificial	53,1%	32,2%
Single person	44,0%	16,3%
10 people	43,7%	8,4%

When looking at how silence length evolves during the series of interviews it is obvious that Askur adapts relatively quickly in the beginning in all cases. The first session where the agent interviews itself it is obviously interviewing the most consistent interviewee; the agent gets constantly better with only minor lapses until it reaches about 70% of silences shorter than 500 msecs and around 40% of silences shorter than 300 msecs. When interviewing a single person for 10 consecutive interviews the system cannot learn as well since there is more variation in behavior.

When interviewing 10 people Askur has reached about 50% of before-turn silences shorter than 500 msecs (see Figure 7), compared to 70% in the

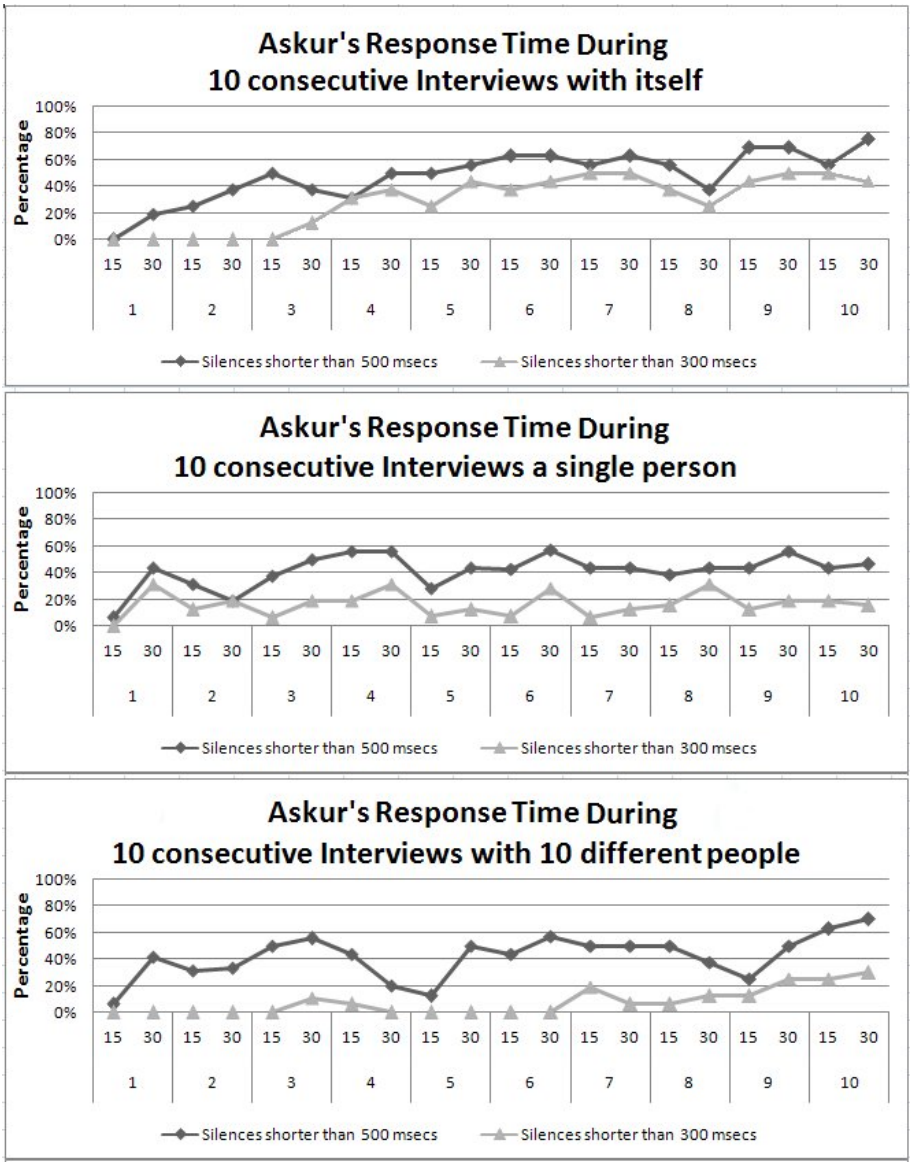

Figure 7: Proportion of silences with human speed characteristics. The graphs show 10 consecutive interviews in 3 different conditions. Each session is 10 consecutive interviews, each interview is 30 turns.

human-human comparison data. There are two distincts dips in performance in interviews 4 and 8. These can be explained with differences in prosody patterns used by participants (see Figure 6). In the case of interviewee number 4 the agent needs to learn that *Up_At* is a turn giving signal (used in 37,5% of 4's turns) but in the case of participants 8 it is not as obvious. While examining overlaps it can be seen that a lot of overlaps occurr in interview 8 and beginning of interview 9 indicating that the agent is making mistakes (see Figure 8).

7.4. Overlapped turns

The final evaluation of success is to view the overlapped turns in each condition. In the first condition when interviewing self (Artificial), the overlaps are mostly in the first half of the evaluation. After that overlaps drop considerably and stay low through the remainder of the sessions. This is due to the consistency of the interlocutor, the system learns to handle the interlocutor and makes very few mistakes towards the end of the evaluation. In the second condition (Single person) when interviewing a single person for 10 sessions, overlaps are below or around 10% for all interviews except beginning of 3rd and 5th interview. In the last scenario where the system interviews 10 different people overlaps occur more randomly due to differences in participants.

It is not surprising that most overlaps are perceived in the last condition, when the system interviews 10 different people (17%). It is however surprising that fewer overlaps are perceived when interviewing a single person over an open microphone than when interviewing an artificial interlocutor in a closed (sound card to sound card) setting (see Table 7). The artificial interlocutor always selects 1 to 3 sentence fragments and inserts artificial "think pauses" of length 0 to 1000 msecs between them, people tend to answer in shorter sentences, not allowing for as many opportunities for mistakes.

Table 7: Average silences for each condition.

	Overlapped turns
Artificial	15,3%
Single person	10,3%
10 people	17,0%

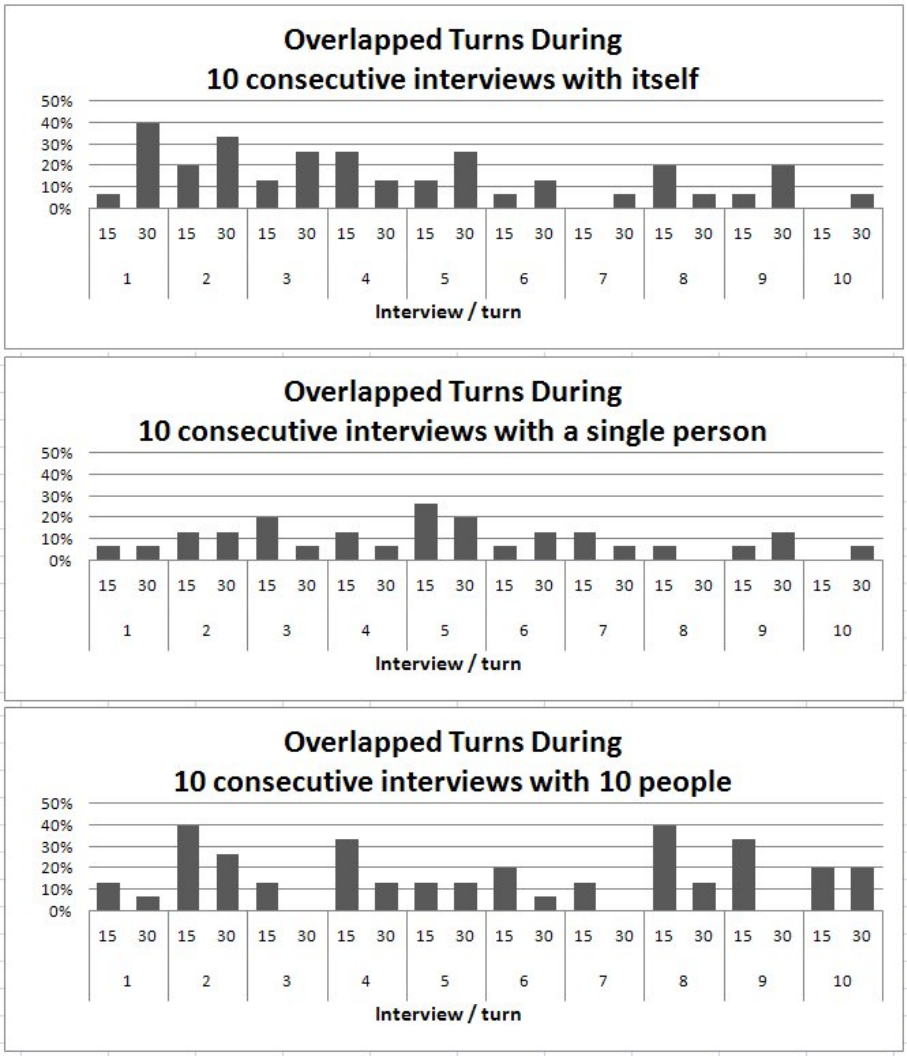

Figure 8: Overlapped turns in our three evaluations.

8. Conclusions & Future Work

We have built a system that uses prosody to learn the optimal silence tolerance window, minimizing speech overlaps and awkward silences. The system learns this on the fly, in a full-duplex "open-mic" (dynamic interaction) setup, and can take turns very efficiently in dialogues with itself and with people, in human-like ways. The system uses prosodic information for finding features of pitch that can serve as a predictor of turn-giving behavior of interlocutors and incremental perception to work in as close to real-time as possible. As the system learns on-line it is able to adjust to the particulars of individual speaking styles. At present, the system strongly targets the temporal characteristics of human-human dialogue, something that is mostly considered irrelevant by prior and related work on dialogue systems, as the above discussion shows. Nevertheless, there is room for significantly more work to be done in this direction.

At present the system is limited in two main ways: it assumes a small set of turntaking circumstances where content does not play a role and it assumes "polite" conversation where both parties want to minimize overlaps in speech. Furthermore, silences caused by outside interruptions - e.g. barge-in techniques and deliberate interruption techniques - are topics for future study. The system is highly expandable, however, as it was built as part of a much larger system architecture that addresses multiple topic- and task-oriented dialogue, as well as multiple modes. In the near future we expect to expand the system to more advanced interaction types and situations. The learning mechanism described here will be expanded to learn not just the shortest durations but also the most efficient turntaking techniques in multimodal interaction under many different conditions. Because of the distributed nature of the architecture the turntaking system is architected in such a way as to allow a mixed-control relationship with outside processes. This means that we can expand it to handle situations where the goals of the dialogue may be very different from being "friendly", even adversarial, as for example in on-air open-mic political debates. How easy this is remains to be seen; the main question revolves around the learning systems - how to manage learning in multiple circumstances without negatively affecting prior training.

Acknowledgments.

This work was supported in part by a research grant from RANNIS, Iceland, and by a Marie Curie European Reintegration Grant within the 6th European Community Framework Programme. The authors wish to thank Yngvi Bjornsson for his contributions to the development of the reinforcement mechanisms.

References

[1] C. Goodwin, Conversational organization: Interaction between speakers and hearers, New York: Academic Press.

[2] G. R. Jonsdottir, J. Gratch, E. Fast, K. R. Thórisson, Fluid semantic back-channel feedback in dialogue: Challenges and progress, in: IVA, Springer, 2007, pp. 154–160.

[3] R. Moore, Presence: A human-inspired architecture for speech-based human-machine interaction, IEEE Trans. Comput. 56 (9) (2007) 1176–1188. doi:http://dx.doi.org/10.1109/TC.2007.1080.

[4] J. F. Allen, G. Ferguson, A. Stent, An architecture for more realistic conversational systems, in: Intelligent User Interfaces, 2001, pp. 1–8. URL `citeseer.ist.psu.edu/allen01architecture.html`

[5] A. Raux, M. Eskenazi, A multi-layer architecture for semi-synchronous event-driven dialogue management, in: ASRU, Kyoto, Japan, 2007, pp. 514–519.

[6] K. R. Thórisson, Natural turn-taking needs no manual: Computational theory and model, from perception to action, in: I. K. B. Granström, D. House (Ed.), Multimodality in Language and Speech Systems, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002, pp. 173–207.

[7] S. K. Card, T. P. Moran, A. Newell, The model human processor: An engineering model of human performance, in: Handbook of Human Perception, Vol. II, John Wiley and Sons, New York, New York, 1986.

[8] M. Wilson, T. P. Wilson, An oscillator model of the timing of turn-taking, Psychonomic Bulletin Review 38(12) (2005) 957–968.

[9] C. Ford, S. A. Thompson, Interactional units in conversation: Syntactic, intonational, and pragmatic resources for the management of turns, in: E. Ochs, E. Schegloff, S. A. Thompson (Eds.), Interaction and Grammar, Cambridge University Press, Cambridge, 1996, pp. 134–184.

[10] L. ten Bosch, N. Oostdijk, L. Boves, On temporal aspects of turn taking in conversational dialogues, Speech Communication 47 (1-2) (2005) 80–86.

[11] G. Jefferson, Preliminary notes on a possible metric which provides for a standard maximum silence of approximately one second in conversation, in: Conversation: an Interdisciplinary Perspective, Multilingual Matters, 1989, pp. 166–196.

[12] R. Sato, R. Higashinaka, M. Tamoto, M. Nakano, K. Aikawa, Learning decision trees to determine turn-taking by spoken dialogue systems, in: ICSLP '02, 2002, pp. 861–864.

[13] D. Schlangen, From reaction to prediction: Experiments with computational models of turn-taking, in: Proceedings of Interspeech 2006, Panel on Prosody of Dialogue Acts and Turn-Taking, Pittsburgh, USA, 2006.

[14] D. R. Traum, P. A. Heeman, Utterance units and grounding in spoken dialogue, in: Proc. ICSLP '96, Vol. 3, Philadelphia, PA, 1996, pp. 1884–1887.

[15] A. Raux, M. Eskenazi, Optimizing endpointing thresholds using dialogue features in a spoken dialogue system, in: Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue, Association for Computational Linguistics, Columbus, Ohio, 2008, pp. 1–10.
URL http://www.aclweb.org/anthology/W/W08/W08-0101

[16] J. Gratch, A. Okhmatovskaia, F. Lamothe, S. Marsella, M. Morales, R. J. van der Werf, L.-P. Morency, Virtual rapport, in: IVA, 2006, pp. 14–27.

[17] K. R. Thórisson, Communicative humanoids: A computational model of psycho-social dialogue skills, Ph.D. thesis, Massachusetts Institute of Technology (1996).

[18] J. Pierrehumbert, J. Hirschberg, The meaning of intonational contours in the interpretation of discourse, in: P. R. Cohen, J. Morgan, M. Pollack (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 271–311.

[19] K. R. Thórisson, Dialogue control in social interface agents, in: InterCHI Adjunct Proceedings, ACM Press, 1993, pp. 139–140.

[20] R. A. Brooks, A robust layered control system for a mobile robot, IEEE Journal of Robotics and Automation 2 (1) (1986) 14–23.
URL http://www.ai.mit.edu/people/brooks/papers/AIM-864.ps.Z

[21] V. Ng-Thow-Hing, T. List, K. R. Thórisson, J. Lim, J. Wormer, Design and evaluation of communication middleware in a distributed humanoid robot architecture, in: E. Prassler, K. Nilsson, A. Shakhimardanov (Eds.), IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'07) Workshop on Measures and Procedures for the Evaluation of Robot Architectures and Middleware, 2007.
URL http://xenia.media.mit.edu/ kris/ftp/IROS07.pdf

[22] J. Bonaiuto, K. R. Thórisson, Towards a neurocognitive model of realtime turntaking in face-to-face dialogue, in: G. K. I. Wachsmuth, M. Lenzen (Ed.), Embodied Communication in Humans And Machines, U.K.: Oxford University Press., 2008, pp. 451–484.

[23] K. R. Thórisson, Modeling multimodal communication as a complex system, in: I. Wachsmuth, G. Knoblich (Eds.), Modeling Communication with Robots and Virtual Humans, Vol. 4930 of Lecture Notes in Computer Science, Springer, 2008, pp. 143–168.

[24] M. Schwabacher, A. Gelsey, Multi-level simulation and numerical optimization of complex engineering designs, in: In 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium, 1996, pp. 96–4021.

[25] K. F. Schaffner, Reduction: the cheshire cat problem and a return to roots, in: Synthese, Vol. 151(3), Springer Netherlands, 2006, pp. 377–402.

[26] N. Gaud, F. Gechter, S. Galland, A. Koukam, Holonic multiagent multilevel simulation: Application to real-time pedestrian simulation in urban environment, in: IJCAI, 2007, pp. 1275–1280.

[27] P. Dayan, Levels of analysis in neural modeling, Encyclopedia of cognitive science.

[28] M. Arbib, Levels of modeling of visually guided behavior, Behavioral and Brain Sciences 10 (1987) 407–465.

[29] M. F. Wood, S. A. Deloach, An overview of the multiagent systems engineering methodology, in: The First International Workshop on Agent-Oriented software Engineering (AOSE-2000, 2000, pp. 207–221.

[30] M. Wooldridge, N. R. Jennings, D. Kinny, The gaia methodology for agent-oriented analysis and design, Autonomous Agents and Multi-Agent Systems 3 (3) (2000) 285–312.
URL http://citeseer.ist.psu.edu/wooldridge00gaia.html

[31] K. R. Thorisson, H. Benko, A. Arnold, D. Abramov, S. Maskey, A. Vaseekaran, Constructionist design methodology for interactive intelligences, A.I. Magazine 25 (4) (2004) 77–90, menlo Park, CA: American Association for Artificial Intelligence.

[32] K. R. Thórisson, T. List, J. DiPirro, C. Pennock, Openair: A publish-subscribe message & routing specification 1.0., Tech. rep. (2004).

[33] K. R. Thórisson, G. R. Jonsdottir, A granular architecture for dynamic realtime dialogue, in: Proceedings of the 8th international conference on Intelligent Virtual Agents, IVA '08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 131–138.

[34] K. R. Thórisson, G. R. Jonsdottir, E. Nivel, Methods for complex single-mind architecture designs, in: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems - Volume 3, AAMAS '08, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2008, pp. 1273–1276.
URL http://dl.acm.org/citation.cfm?id=1402821.1402849

[35] G. R. Jonsdottir, A distributed dialogue architecture with learning, Master's thesis, Reykjavik University (2008).

[36] E. Nivel, K. R. Thórisson, Prosodica realtime prosody tracker, Tech. rep., Reykjavik University Department of Computer Science, technical Report RUTR-CS08001 (2008).

[37] K. R. Thórisson, Machine perception of multimodal natural dialogue, in: P. McKevitt, S. Ó. Nulláin, C. Mulvihill (Eds.), Language, Vision & Music, John Benjamins, 2002, 2002, pp. 97–115.

[38] R. S. Sutton, A. G. Barto, Reinforcement Learning: An Introduction, The MIT Press, Cambridge, MA, 1998.