A Granular Architecture for Dynamic Realtime Dialogue

Kristinn R. Thórisson and Gudny R. Jonsdottir

Center for Analysis & Design of Intelligent Agents
and School of Computer Science
Kringlunni 1, 103 Reykjavik, Iceland
{thorisson, gudny04}@ru.is

Abstract. We present a dialogue architecture that addresses perception, planning and execution of multimodal dialogue behavior. Motivated by realtime human performance and modular architectural principles, the architecture is full-duplex ("open-mic"); prosody is continuously analyzed and used for mixed-control turntaking behaviors (reactive and deliberative) and incremental utterance production. The architecture is fine-grain and highly expandable; we are currently applying it in more complex multimodal interaction and dynamic task environments. We describe here the theoretical underpinnings behind the architecture, compare it to prior efforts, discuss the methodology and give a brief overview of its current runtime characteristics.

Keywords: Architecture, turntaking, dialogue, realtime, incremental, planning, interaction, full-duplex, granular.

1 Introduction

Many researchers have pointed out the lack of implemented systems that can manage full-duplex ("open-microphone") dialogue (cf. [1,2,3]), that is, systems that can interrupt – and be interrupted – at any point in time, in a natural manner. As pointed out by Allen et al. [3], Moore [1] and others, much of the work in the field of dialogue over the last 2-3 decades has enforced strict turntaking between the system and the user, resulting in fairly unnatural, stilted dialogue. The challenge in building such systems lies, among other things, in the complexity of integration that needs to be done: Several complex systems, each composed of several complex subsystems – and those possibly going another level down – need to be combined in such a way as to produce coordinated action in light of complex multimodal input.

In this paper we describe work on building an architecture that can address many of the rich features of realtime multimodal dialogue. As many have pointed out (cf. [4]), a proper theory of turntaking should cover varied situations ranging from debates, to lectures, negotiations, task-oriented interactions, media interviews, dramatic performances, casual chats, formal meetings, task-oriented communication on a noisy factory floor, communication between the deaf, successful communication on the telephone with no multimodal information but plenty of paraverbal information, etc. While we do not claim to be close to any such comprehensive theory

or model, our approach nods in the direction of methods, techniques and architectural features that promise to take us closer to such a comprehensive state.

Among the criticisms fielded by Moore [1] towards the present state of the art in dialogue systems are the brittleness of current approaches to both recognition and synthesis and lack of holistic integrative approaches. We address these with a two-prong approach: A more granular architecture that is easier to extend and manage than alternative approaches, and a more thorough methodology for building the architecture, growing out of prior work of one of the authors on similar topics [5]. While clearly not addressing all of the topics relevant to dialogue, the architecture has already shown itself to be highly expandable, in particular supporting complex modeling of turntaking [6,7]. The system has been successfully outfitted with learning capabilities for adjusting realtime behavior to match prosodical speech patterns of interlocutors [8]. In this paper we focus on the gross architecture and describe – from a 10k-foot viewpoint – the fundamental theoretical pillars of our approach.

2 Related Work

The present work builds directly on the Ymir architecture and framework [6], and the Ymir Turn-Taking Model (YTTM) [7]. Thórisson [6] presents the goals behind the Ymir architecture in 12 main points, chief among them being full-duplex, natural multimodal interaction, with natural response times (realtime). This includes also incremental interpretation and output generation, and the requirement that generation can progress in parallel with interpretation. Of the architectures built for spoken discourse, most have left one or more of the above constraints unaddressed, often focusing on isolated integration problems in dialogue such as concept-to-speech generation with proper intonation [9] and integration of dialogue modeling and speech generation [10].

Notable prior work has aimed to identify so-called "turn-constructional units" from a proposed set of candidates that has included the sentence [4,12], syllables [5], multimodal cues [4], phoneme timing [11] and semantics [4]. We take the view that turntaking is an *emergent* property of a large set of interacting systems [13]. Therefore, we consider attempts at modeling turntaking based on such "units of construction" a futile exercise. A less obvious results is that any attempt that shies away from addressing a substantial amount of the gross features and richness of dialogue up front may be doomed, as reductionist approaches are in general bad for studying highly emergent phenomena [13,14]. Thus, any attempt at building such integrative architectures in stages must directly address the challenges of expandability and incremental construction up front.

Allen et al.'s [3] system integrates a large number of functionalities in a comprehensive architecture that, while using a push-button interface for turn exchange, performs interpretation and output generation planning incrementally, in parallel. This supports user barge-in, with supported backtracking and other sophisticated dialogue handling. The architecture goes beyond the Gandalf system [6] in content interpretation; although Gandalf could interpret gestures, body language and prosody incrementally the content of speech was done "batch-processing style"

(caused by limitations in the speech recognition technology used). Similar to Thórisson [6,7], the authors argue for a separation of dialogue skills and topic skills.

Raux [2] presents a broad dialogue-capable architecture, extending ideas from Ymir [7] with more extensive dialogue management techniques. The architecture is blackboard-based like Ymir with two rather large-grain modules: An Interaction Manager, bridging between high-level and lower-level system components, controlling directly reactive behaviors (e.g. back-channel feedback), and a Dialogue Manager that plans the contributions to the conversation. The former corresponds roughly to the turntaking mechanisms in Ymir, the latter to Knowledge Bases in the Content Layer (like Ymir, the DM makes no prescriptions for the particular technology used for utterance planning). The work demonstrates the flexibility of the blackboard approach for building mixed-granularity architectures [15]. Although the resulting system is comparable to Ymir at the high level, it proposes different solutions for integrating reactive behaviors, turntaking and dialogue state, while promising to handle a wider range of dialogue styles and phenomena. However, as no comparative evaluation is provided it is difficult to assess the benefits of the alternative approach. As it is based on more coarse-grain components, we would expect it to be less expandable than the finer-grain approach we have chosen.

3 Theoretical Underpinnings

In addition to an underlying theory of turntaking in multimodal realtime dialogue, outlined in [7], our approach rests on three main theoretical pillars. The first is motivated by arguments from – or rather a critique of – the standard scientific reductionist method, the second by architectural methodology and the third by data and models from psychological studies.

We view embodied dialogue as a heterogeneous, large, densely-coupled system (HeLD), identifying dialogue in a class of *complex systems,* in the sense of Simon [14] and his concept of *near decomposability*. The main arguments for this view have been presented in [13] and [16]. HeLD systems embody and express emergent properties that have been difficult to understand without resorting to large, detailed computational models built to relatively high levels of fidelity. Without the ability to experiment with changes and modifications to the architecture at various levels of detail we cannot differentiate between a large set of models that, on paper, look like they might all work. The conclusion can only be that 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 is echoed in some recent work on dialogue architectures (cf. [1]). This, however, may seem to present an impossible difficulty; creating even *approximately* complete models of dialogue – ones that address multiple modes, prosody, realtime content generation, etc. – may seem to entail an insurmountable effort. Counterintuitively, for most complex systems, however, if we attempt to take *all* of the most significant behaviors of the system into account, the set of possible contributing underlying mechanisms will be greatly reduced [17] – quite possibly to a small finite set (while initial formulation of plausible mechanisms may be harder, because the constraints of the

work are greater up front, the search is now over a manageable set of possibilities). One way to build satisfactory models of dialogue is to bring results from a number of disciplines to the table, at various levels of abstraction. 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 [18].

Following this view we have built our present model using a methodology that helps us create complex multi-component systems at a fairly high fidelity without losing control of the development process, the Constructionist Design Methodology (CDM) [5]. Many of the extant methodologies that have been offered in the area of agent-based simulation and modeling suffer from lack of actual use case experiences, especially for artificial intelligence projects that involve construction of single-mind systems. CDM's 9 iterative principles (semi-independent steps) have already been applied in the construction of several systems, both for robots and virtual agents (cf. [5,15,19]). Our approach has followed it fairly closely.

When dealing with HeLDs we must try to constrain the possible design space. A powerful way to do this is to build multilevel representations (cf. [17,20,21]); this may in fact be the only way to get our models right when trying to understand natural HeLDs. Notice that the thrust of the 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 HeLDs the multiple levels are a *critical necessity*: Without simulations at fairly high levels of fidelity we cannot expect manipulations to the architecture (at various levels of detail) to produce valid results. Without this ability we cannot select from a large set of candidate approaches that, on paper, look like they might all work. We have tried to do this using data from the psychological literature to constrain the design achieved, most significantly temporal characteristics. As we are in the beginning stages of building such multi-level-of-detail models, work remains in this respect. However, the effort has already helped with the construction; Bonaiuto and Thórisson's work [22] on systems that mutually learn to perceive and produce multimodal turntaking cues is based on the turntaking mechanisms described here, built using neurally plausible models of planning which are modeled after brain research on Macaques.

4 The Architecture

The architecture is composed of two main functional clusters. First, a turntaking (TT) system that is able to support realtime turntaking [8], built such that it can interface with external systems in a disciplined way. The second cluster manages (a) continuous assessment of dialogue state, (b) the internal drive for delivering content, (c) and planning for future actions, including utterances and multimodal behavior.

Following the CDM [5] we started at the low level with several perception modules for extracting prosodical information from a person's speech (pitch, silences, speaking volume and compounds of these). Then we expanded the system with a set of control/decision modules that, based on the perceptual processing output of the perception modules, decide which turntaking context was most likely (I-Have-Turn,

Other-Has-Turn, etc.). The design process is described further in [23]. Last but not least, we make extensive use of *contexts* – semi-global states that determine which modules are active when [15], allowing us to control large groups of modules in the architecture as one unit, improving management; they also allow us to better view the runtime operation of the full system and more carefully control CPU load.

We will now give a short overview of some of the key architectural components, some of which are depicted in Figure 1. *Prosody Tracker* (low-level module): Analyses the pitch, pitch derivative, speech on/off, silences and hums in a continuous manner. *Prosody Analyzer* (mid-level): Analyzes data from the PT to identify speech overlap and silences (also quantizes continuous pitch for efficiency). *Speech Recognizer* (high-level): Produces semi-continuous text from continuous audio signal [8]. *Interpreters* (high-level): A group of perception/interpretation modules that take in output from a single SR; each having its specified identification task, e.g. finding nouns, dates, fillers, etc. *Interpretation Director* (high-level): Receives input from Content Planner (see below) on what to look for at any point in time and analyzes the output of all Interpreters based on that information [8]. *Turntaking System* (TT) (low-level): The TT consists of unimodal/multimodal preceptors and deciders; its role is to maintain a coupling between the interlocutors of turn "state", or *disposition – a grouping of goals and expectations, in the form of perceptor and decider activations*. Based on it, all content-unaware decisions and actions related to the delivery and interpretation of speech acts can be coordinated. Global states (contexts) prescribe which perceptions and decisions are appropriate at any point in time, e.g. whether to expect a certain turntaking cue, whether it is relevant to generate a particular behavior on a particular cue, etc. *Dialogue Planner* (high-level): Is responsible for delivering the next "thought unit", embodied as a short segment of speech spanning roughly 1-2 second of delivery time, on average, when it is available from the CP, producing fillers and gracefully giving turn when content is not generated within a set time frame. The DP contains a motivation-to-speak, which currently is linked directly to have-something-to-say. *Content Planner* (high-level): Decides *what* to say based on inner goals and information from the ID. *Speech Synthesizer* (mid-level): Takes commands from CP and TT to start/stop speech, raise or lower speech volume. *Learner* (mid-level): Computes a decision strategy incrementally while learning on-line, and publishes it to the Other-Gives-Turn-Decider-2, which determines how long to wait until starting to speak as a silence is detected, using as an indication the prosody of the last 300 msec of the other's utterance right before the silence [8].

The Dialogue Planner and Learning module share control of turntaking, even though they might be considered to be outside of the TT system proper; typically affecting the contexts I-Want-Turn, I-Accept-Turn and I-Give-Turn. If the Content Planner has some content ready to be communicated – irrespective of what the perceptions inform the TT system that the coupled dialogue context is – the agent can signal that it wants turn via this "deliberate" route; it can also signal I-Give-Turn when content queue is empty – the TT system will not be forced to handle this as our theory suggests it should be kept content-unaware. These decisions made by the DP typically override decisions made in the reactive level.

Currently the system is implemented on three workstations, running within the Psyclone framework [15]. The distribution of modules across the three machines has

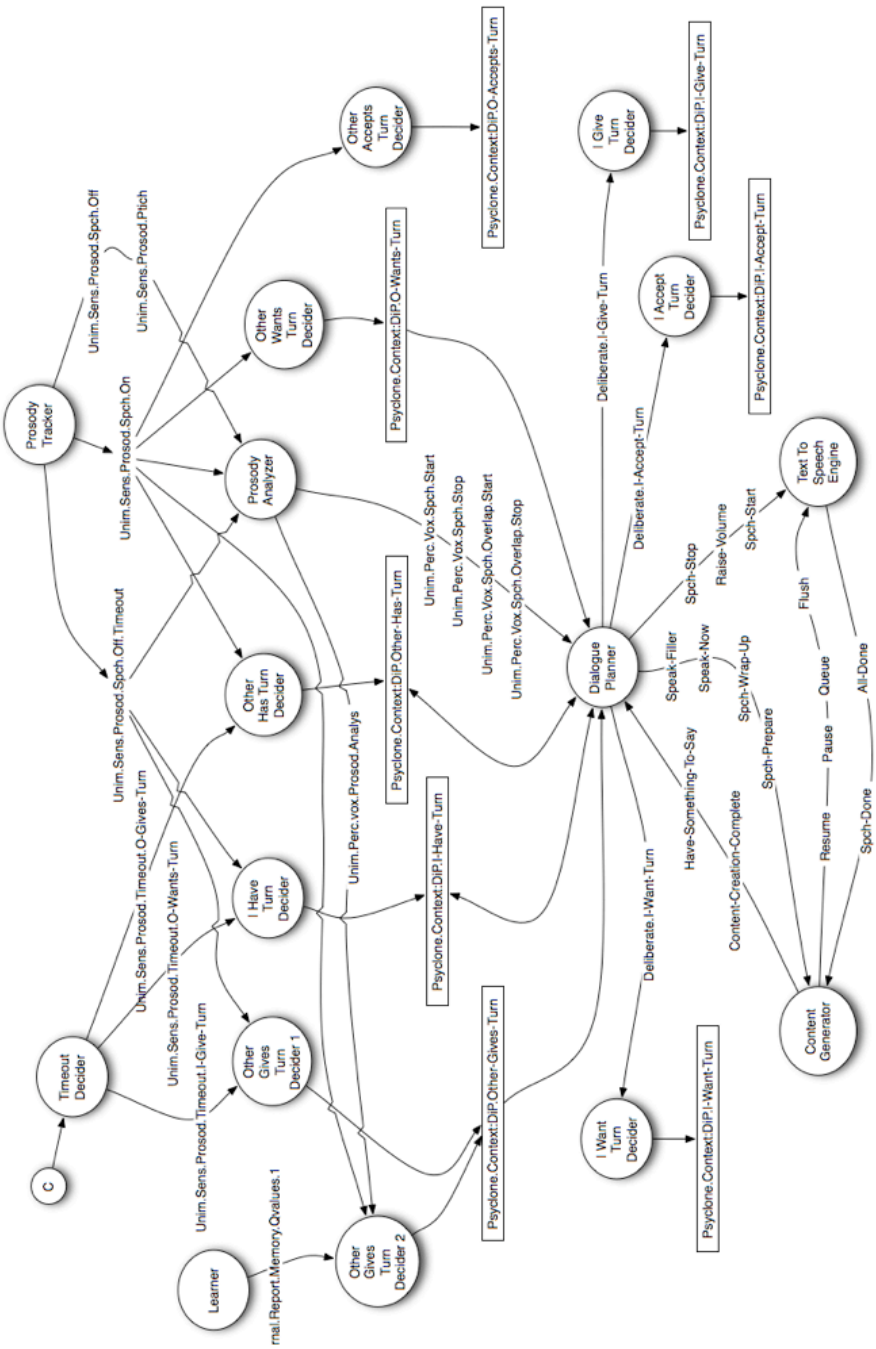

Figure 1. Partial view of architectural components, routhing and message types. Messages are given names in the order from the most significant to least significant descriptor of the message's contents [15].

Table 1. Temporal characteristics of selected message passing at runtime. Time shows how long, in msec, each type of message takes to be transmitted between sender and receiver. (Resolution = 16 msec.)

Message Type	Receiver	Ave	Min	Max
Deliberate.I-Accept-Turn	I-accept-turn	**107**	97	207
Psyclone.Context:DiP.I-Accept-Turn	I-have-turn	**39**	15	63
Unim.Sens.Prosod.Spch.Off.Timeout	I-have-turn	**4**	0	16
Psyclone.Context:DiP.I-Give-Turn	Other-accepts-turn	**59**	4	176
Unim.Sens.Prosod.Spch.On	Other-accepts-turn	**9**	0	47
Unim.Sens.Prosod.Spch.On	Other-gives-turn-2	**20**	0	297
Internal.Report.Memory.Qvalues	Other-gives-turn-2	**18**	15	47
Unim.Perc.vox.Prosod.Analys	Other-gives-turn-2	**82**	62	469
Psyclone.Context:DiP.I-Want-Turn	Other-gives-turn-1	**53**	3	207
Unim.Sens.Prosod.Spch.Off.Timeout	Other-gives-turn-1	**7**	0	31
Unim.Sens.Prosod.Timeout.I-Give-Turn	Other-gives-turn-1	**31**	0	47
Psyclone.Context:DiP.Other-Accepts-Turn	Other-gives-turn-1	**87**	16	329
Psyclone.Context:DiP.Other-Accepts-Turn	Other-has-turn	**39**	0	203
Unim.Sens.Prosod.Spch.On	Other-has-turn	**18**	0	110
Unim.Sens.Prosod.Spch.On	Other-wants-turn	**5**	0	16
Unim.Sens.Prosod.Spch.Off	ProsodyAnalyzer	**73**	62	110
Unim.Sens.Prosod.Spch.Pitch	ProsodyAnalyzer	**6**	0	31
Internal.Instruct.Spch.Prepare	ContentGenerator	**4**	0	15
Internal.Instruct.Spch.Start	ContentGenerator	**2**	0	16
Output.Plan.Task.Spch.Done	ContentGenerator	**4**	0	16
Output.Plan.Task.Speak.Now	ContentGenerator	**2**	0	15
Output.Plan.Task.Spch.All-Done	ContentGenerator	**0**	0	0
Internal.Content-Creation-Complete	DialoguePlanner	**19**	0	32
Have-Something-To-Say	DialoguePlanner	**25**	0	47

been carefully tuned by hand to achieve sufficient processing and transmission speeds to support the realtime running of the system. The system has been tested both in simulated interactions with itself and in interactions with people (part of this data is published in [8]). Table 1 summarizes key operating characteristics in terms of message passing times.

Acknowledgments. This work was supported in part by a research grant from RANNÍS, Iceland, and by a Marie Curie European Reintegration Grant within the 6th European Community Framework Programme. The authors would like to thank Eric Nivel for his work on the *Prosodica* prosody tracker and Yngvi Björnsson for his contributions to the learning methods.

References

1. Moore, R. K.: PRESENCE: A Human-Inspired Architecture for Speech-Based Human-Machine Interaction. IEEE Transactions on Computers, 56(9), 1176-1188 (2007)
2. Raux, A., Eskenazi, M.: A Multi-Layer Architecture for Semi-Synchronous Event-Driven Dialogue Management, ASRU, Japan, 514-519 (2007)
3. Allen, J., Ferguson, G., Stent, A.: An Architecture for More Realistic Conversational Systems. In: IUI, ACM Press, Santa Fe, 14-17 (2001)

4. O'Connell, D.C., Kowal, S., Kaltenbacher, E.: Turn-Taking: A Critical Analysis of the Research Tradition. Journal of Psycholinguistic Research 19(6), 345-373 (1990)
5. Thórisson, K. R., H. Benko, A. Arnold, D. Abramov, S. Maskey, A. Vaseekaran: Constructionist Design Methodology for Interactive Intelligences. A.I. Magazine, 25(4): 77-90, American Association for Artificial Intelligence, Menlo Park, CA (2004)
6. Thórisson, K. R.: A Mind Model for Multimodal Communicative Creatures and Humanoids. International J. Appl. Artif. Intell., 13(4-5):449-486 (1999)
7. Thórisson, K. R.: Natural Turn-Taking Needs No Manual: Computational Theory and Model, from Perception to Action. In: B. Granström, D. House, I. Karlsson (Eds.), Multimodality in Language and Speech Systems, Kluwer Academic Publishers, Dordrecht, the Netherlands, 173-207 (2002)
8. Jonsdottir, G. R., Thórisson, K. R., Nivel, E.: Learning Smooth, Human-Like Turntaking in Realtime Dialogue. In: IVA Japan, this volume (2008)
9. Pan, S., McKeown, K. R.: Integrating Language Generation with Speech Synthesis in a Concept to Speech System. Proceedings of the ACL Workshop on Concept to Speech Generation Systems. ACL/EACL, (1997)
10. Grote, B., Hagen, E., Teich, E.: Matchmaking: Dialogue Modeling and Speech Generation Meet. Proceedings of the 1996 International Workshop on Natural Language Generation, Herstmonceux, England, 171-180 (1996)
11. Wilson, M., Wilson, T.P.: An oscillator model of the timing of turn-taking. Psychonomic Bulletin and Review 12(6), 957-968 (2005)
12. Sacks, H., Schegloff, E.A. Jefferson, G.A.: A Simplest Systematics for the Organization of Turn-Taking in Conversation. Language 50, 696-735 (1974)
13. Thórisson, K. R.: Modeling Multimodal Communication as a Complex System. In: I. Wachsmuth, M. Lenzen, G. Knoblich (eds.), Modeling Communication with Robots and Virtual Humans, Springer, New York, 143-168 (2008)
14. Simon, H.A.: Can there be a science of complex systems? In: Y. Bar-Yam (Ed.), Unifying themes in complex systems: Proceedings from the International Conference on Complex Systems, 4-14. Perseus Press, Cambridge (1999)
15. Thórisson, K. R., T. List, C. Pennock, J. DiPirro: Whiteboards: Scheduling Blackboards for Semantic Routing of Messages & Streams. Proceedings of AAAI-05, AAAI Technical Report WS-05-08, 8-15 (2005)
16. Thórisson, K. R.: Integrated A.I. Systems. Minds & Machines, 17:11-25 (2007)
17. Scwabacher, M., Gelsey, A.: Multi-Level Simulation and Numerical Optimization of Complex Engineering Designs. 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optmization Symposium, Bellevue, WA, AIAA-96-4021 (1996)
18. Schaffner, K.F.: Reduction: the Cheshire cat problem and a return to roots. Synthese 151(3), 377-402 (2006)
19. Ng-Thow-Hing, V., List, T., Thórisson, K.R., Lim, J., Wormer, J.: Design and Evaluation of Communication Middleware in a Distributed Humanoid Robot Architecture. IROS '07 Workshop Measures and Procedures for the Evaluation of Robot Architectures and Middleware, 29 Oct. - 2 Nov. San Diego, California (2007)
20. Gaud, N., Gechter, F., Galland, S., Koukam, A.: Holonic Multiagent Multilevel Simulation Application to Real-time Pedestrians Simulation in Urban Environment. Proceedings of IJCAI-07, 1275-1280 (2007)
21. Arbib, M.A.: Levels of Modeling of Visually Guided Behavior (with peer commentary and author's response), Behavioral and Brain Sciences 10, 407-465 (1987)
22. Bonaiuto, J. and Thórisson, K. R.: Towards a Neurocognitive Model of Realtime Turntaking in Face-to-Face Dialogue. In: I. Wachsmuth, M. Lenzen, G. Knoblich (eds.), Embodied Communication in Humans and Machines. Oxford University Press, U.K. (2008)
23. Thórisson, K. R., G. R. Jonsdottir and E. Nivel: Methods for Complex Single-Mind Architecture Designs. In: Proc. AAMAS, Portugal, June (2008)