# Interactional Adequacy as a Factor in the Perception of Synthesized Speech

*Timo Baumann*

Department of Informatics
Universität Hamburg, Germany
baumann@informatik.uni-hamburg.de

*David Schlangen*

Faculty of Linguistics and Literature
Bielefeld University, Germany
david.schlangen@uni-bielefeld.de

## Abstract

Speaking as part of a conversation is different from reading out aloud. Speech synthesis systems, however, are typically developed using assumptions (at least implicitly) that are more true of the latter than the former situation. We address one particular aspect, which is the assumption that a fully formulated sentence is available for synthesis. We have built a system that does not make this assumption but rather can synthesize speech given incrementally extended input. In an evaluation experiment, we found that in a dynamic domain where what is talked about changes quickly, subjects rated the output of this system as more 'naturally pronounced' than that of a baseline system that employed standard synthesis, despite the synthesis quality objectively being degraded. Our results highlight the importance of considering a synthesizer's ability to support interactive use-cases when determining the adequacy of synthesized speech.
**Index Terms**: speech synthesis, incremental processing, interactive behaviour, evaluation, adequacy

## 1. Introduction

Most speech synthesis software is not tailored towards interactive use, but instead operates in a way that is best described as reading out aloud. As a consequence, full sentences (or utterances in dialogue) are used as input units, and typically, input cannot be changed or extended after the synthesizer's processing has started.

This coarse input granularity and monolithical processing reduce the ability to adapt to unforeseen changes in the environment, which may be necessary (or at least advantageous) in interactive systems, such as commentary generation, or conversational dialogue systems. Thus, interactive systems may profit from speech synthesis that uses smaller, partial input units that are extended *incrementally* and just-in-time, while speech output is already ongoing, to produce an utterance in a piece-meal fashion.

Dutoit et al. [1] have previously shown that incremental, HMM-based speech synthesis is possible and only moderately degrades synthesis quality; however, their speech synthesizer is not integrated into a full text-to-speech system. We have built an interactive text-to-speech synthesizer, INPRO_iSS [2],[1] based on MaryTTS [3] and the incremental processing toolkit INPROTK [4], which is able to produce output based on incrementally expanded utterance descriptions, and which also allows to change delivery parameters of ongoing speech, such as tempo, pitch, and – added in this work – force.

We have previously shown that incremental speech synthesis, in combination with incremental natural language generation, is profitable in order to remain flexible with regards to external events from the environment [5]. In a user study, participants rated the naturalness of the formulation and the pronunciation of our system in a highly dynamic environment. Analysis of participant ratings showed that the formulations enabled by incrementally synthesizing speech were preferred (by a large margin) over baseline formulations, even if incremental formulation sometimes has to resort to using a hesitation when events unfold more slowly than anticipated [6]. In this work, we present the result that users in addition rated the incremental system's pronunciation as significantly more natural, despite that fact that objectively pronunciation quality was lower. In our opinion, this result highlights the importance of considering a synthesizer's abilities to support interactive use-cases when determining the 'quality' of the synthesized speech.

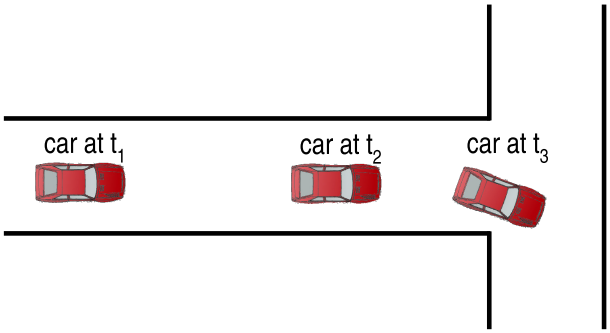
In Section 2, we detail our system's implementation for incrementally provided input as well as timely adaptation of delivery parameters. We describe the domain of our system in Section 3, the evaluation experiment in Section 4, and present the results in Section 5. We draw conclusions from the experiment in Section 6 and outline ideas for future work in Section 7.

## 2. Incrementality and timely adaptation

In our system, textual material to be synthesized is added in 'chunks', which ideally correspond to phrases, but which may also be shorter, down to individual words. Chunks are added to the system incrementally, and prosody is re-computed to reflect changes in the textual and prosodic analysis given the added material as soon as the material becomes available. This means that prosodic quality is highest when material is added early on, but our previous work has shown that having one chunk/phrase of lookahead at all times is sufficient for prosody (pitch and duration) to be almost indistinguishable to non-incrementally produced pitch and duration assignments [7]. It is also possible to revoke parts of the input (that have not been produced yet), and to construct *utterance plans* [8], which may contain multiple alternative paths for possible realization that can be selected until immediately before speech realization reaches the branching point in the plan.

Incremental extension of ongoing utterances allows the system to generate behaviour such as the one shown in Figure 1: in the figure, a car is shown driving along a street, and eventually turning. An incremental system that is to comment on these

[1] INPRO_iSS is available as part of the INPROTK distribution at http://inprotk.sf.net.




| time | event description | ongoing utterance (already realized part in **bold**, newly appended continuation in *italic*) |
|---|---|---|
| $t_1$ | car on Main Street | **The** car drives along Main Street. |
| $t_2$ | car will likely turn | . . . **drives along Main Stre**et *and then turns ‹hes›* |
| $t_3$ | car turns right | . . . **drives along Main Street and the**n turns *right.* |

Figure 1: Example of incremental utterance production as a car drives along a street and turns. The ongoing utterance is extended as information becomes available.

events is able to generate one complex, successively extended utterance, as in the figure, by adapting ongoing synthesis. As in the figure, the system may hypothesize the upcoming turn at time $t_2$ and start to output the part of the utterance that is independent of the direction of the turn. It may then speak about the direction of the car's turn immediately when it happens at $t_3$. In contrast, a non-incremental system has to wait until $t_3$ before it may start its commentary about the car turning because it requires to know the direction of the turn, despite of the fact that the car *will* likely turn was known at time $t_2$ and the beginning of the description ("and then turns") being identical for either direction. Of course, an incremental system may mis-judge the time at which the direction of the car turning (or any other anticipated event) happens. As a countermeasure, our system may be ordered to output a hesitation when it runs out of speech material, in order to gain time (as shown in the second line of the example in Figure 1). Hesitations are skipped (or immediately aborted) as soon as more speech material becomes available (as shown in the third line of the example).

The architectural overview of our system, as given in Figure 2, shows the *just-in-time* approach that is used. The overall goal is to perform processing steps as late as possible, which keeps overheads that are due to later changes of the input to a minimum. In addition, most of the processing time is moved into the *delivery time* of the speech, resulting in improved system response compared to standard processing (see also [5]). The time at which processing is required depends on the level of abstraction: vocoding need only be performed immediately before the corresponding audio is requested, and HMM optimization is performed step-wise using local phoneme contexts (as proposed by [1], but also using global variance optimization [9] within the local context) for each phoneme; higher-level processing must be performed somewhat in advance, and needs to be able to accomodate changes that may result from later addition/revocation of input.

Our processing architecture INPROTK [4] is based on *incremental units* (IUs) [10]. IUs are shown as boxes in Figure 2 and related units are connected via *same level links* for data of

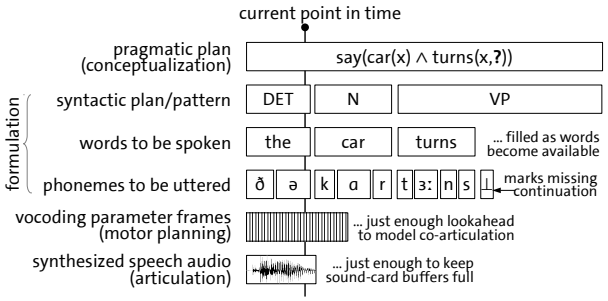



Figure 2: Hierarchical structure of incremental units describing an example utterance as it is being produced during utterance delivery.

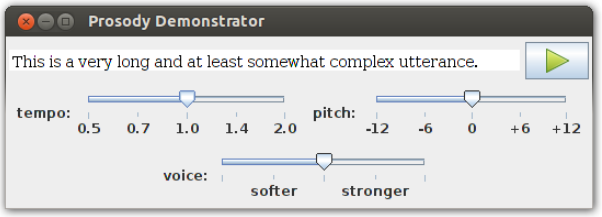



Figure 3: Example graphical interface to incrementally manipulate speech delivery parameters.

the same type (shown in the figure by horizontal alignment) and *grounding links* for hierarchical dependence over different levels (shown in the figure by placing units above/below other units. The links are used to track dependencies in the system and both links and units are revised whenever material is added, removed, or changed incrementally. Furthermore, IUs are active objects, which are set up to automatically request relevant processing steps via an update mechanism. Linguistic pre-processing and prosody assignment relies on MaryTTS [3], which is called repeatedly whenever new material is added to the ongoing utterance.

Linguistic pre-processing and (to a lesser degree) HMM optimization are computationally expensive. For this reason, we added ways to adapt speech delivery parameters outside of the HMM framework that work with almost zero delay. The system uses STRAIGHT vocoding [11], and is able to alter the different vocoding parameters (pitch, cepstrum, energy, and voicing strengths) until immediately before a frame is vocoded. Furthermore, to allow for a simple, yet effective method to change speech tempo without requiring to reperform the HMM optimization, we allow the system to skip or to repeat generated parameter frames, which leads to faster (or slower, respectively) speech – however, ignoring the HMM optimality criteria. (It should be noted that this method works well for moderate tempo changes ($\pm 30\%$) only and leads to acoustic artifacts for extreme changes.)

The capabilities of our adaptation method are exploited in a demonstrator, depicted in Figure 3: it allows to alter pitch, tempo, and voice force (a linear combination of changes in total energy, spectral tilt, as proposed in [12], and additionally voicing strength) in real time (less than 5 ms delay). However, this capability is used only to a limited degree in the experiment

reported below (pitch and duration are adapted just-in-time in the vicinity of hesitations).

## 3. System domain

To test the merit of incremental speech synthesis, we built a system for an interactive commentary domain. The domain combines aspects of sports commentary [13], which often profits from open-ended utterances, with interactive map exploration descriptions for the visually impaired [14].

In our *CarChase* domain, shown in Figure 4, a car drives around the streets on the map and a commentator (supposed to be observing the scene from above) comments on where it is driving and what turns it is taking.

The car's itinerary in our domain simulator is scripted from a configuration file which assigns target positions for the car at different points in time and from which the motion and rotation of the car is animated. The speed of the car is set so that the event density is high enough that the setting cannot be described by simply producing one utterance per event; instead, utterances need to be aborted to make room for new material (baseline behaviour), or utterances need to integrate later events while they are already ongoing (incremental behaviour).

Our system distinguishes three different types of events: street *identification*, the car taking a *turn*, and *turn preparatory* events that become active when it is obvious that the car will turn but the direction of the turn cannot yet be determined. The three event types are shown in Figure 1 at times $t_1$ (*ID*), $t_2$ (*turn-prep*), and $t_3$ (*turn*). While it is an advantage of the incremental system that it may combine multiple events into one longer, connected utterance, the main advantage for temporal adequacy of the commentary comes from *turn-prep* events, which allow to start producing some material about the event (the fact that a turn will occur) even before the direction of the turn can be specified.

The focus of our work is only on incremental speech synthesis, and hence we did not implement an automatic scene analysis/event detection nor an NLG component for the task (however, see [15, 16] for such components in a highly related domain). Instead, commentary text is scripted from the same configuration file that controls the car's motion on the board. Events that control speech synthesis lag behind motion events slightly, ensuring that visual analysis would be possible, and event/text correspondence – although hand-written – is close, matching NLG capabilities.

## 4. Experiment

We evaluated the incremental system by comparing its output to a non-incremental baseline system which is unable to extend ongoing partial utterances and hence cannot incrementally combine multiple events into one utterance. Instead, the baseline system produces one full utterance per event. To ensure timeliness of commentary even in the baseline system, some commenting events were marked as optional (in which case the corresponding utterances are skipped if the system is still outputting a previous utterance), whereas non-optional utterances abort any ongoing commentary in favour of the next utterance. All *turn* events in the domain were marked as optional, all street *ID* events as non-optional. Of course, the baseline system cannot make use of *turn-prep* events.

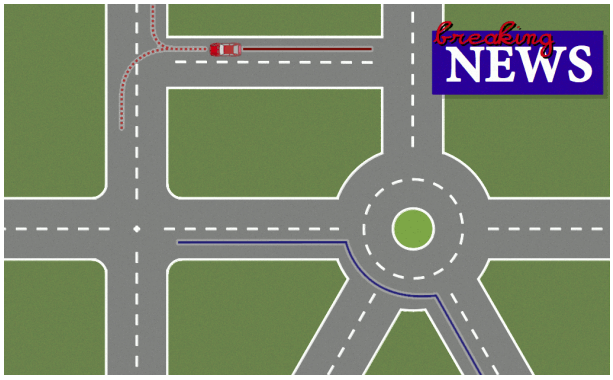



Figure 4: The map shown in the *CarChase* domain, including the car on one of its itineraries (red). At the depicted moment we can assume that the car will take a turn, but do not know whether left or right. A second itinerary is shown in blue.

We devised 4 different configurations (including the itineraries shown in Figure 4), and the timing of events was varied (by having the car go at different speeds, or by delaying some events), resulting in 9 scenarios; in 3 of these, the incremental system *over-commits* to the appearance of a *turn* event and needs to play a short hesitation ('ehm') before the direction of the turn event becomes known. These cases were meant to include errors that are specific to the incremental system's behaviour into the evaluation and thus lead to a more balanced comparison to the baseline system.

Both systems' output for the 9 scenarios was recorded with a screen recorder, resulting in 18 videos that were played in random order to 9 participants (university students not involved in the research) who were told that various versions of commentary-generating systems generated the commentary based on the running picture in the videos and were then asked to rate each video on a five-point Likert scale with regards to how natural (similar to a human) the spoken commentary was (a) formulated, and (b) pronounced. We did not further specify what exactly was meant by 'formulation' or 'pronunciation', instead relying on the participants' intuitive understanding of these terms. In total, the questionnaires resulted in 81 paired samples for each question.

The experiment was performed with an early version of the system, which still performed some prosodic mis-alignments at utterance extensions, due to various shortcomings. Furthermore, the coarsely implemented hesitations result in audible acoustic and prosodic artifacts. Overall, we hoped that the incremental system's formulation would be preferred by participants, without a significant decrease in pronunciation ratings.

## 5. Results

As expected and shown in Figure 5, participants highly preferred the incremental system's formulations over the non-incremental baseline system, with a median difference in ratings of the two conditions of 2 points (mean 1.66), which is highly significant (sign test, 68+/9=/4-; $p < .0001$). For the incremental system, we distinguished between settings where the system generated a hesitation (*hes*) and those where it did not (*no hes*). As can be seen in the figure, even utterances in which the incremental sys-

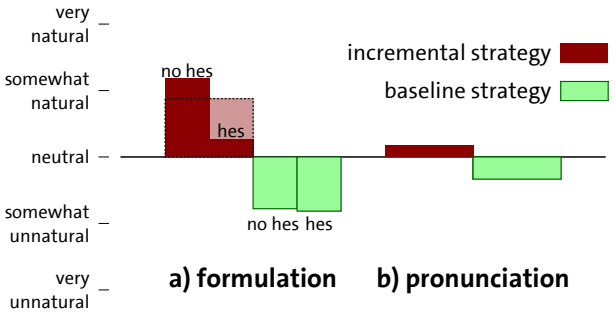



Figure 5: Mean ratings of formulation and pronunciation quality for the incremental and the baseline system. The formulation rating is shown subdivided for utterances with and without hesitations.

tem had to resort to a hesitation were rated as significantly better formulated than the baseline behaviour (see also [6]). There was no significant difference between pronunciation ratings for the *hes*/*no hes* conditions.

More relevant for the present discussion, however, are the *pronunciation* rating differences between the incremental and baseline systems, which also show a clear preference for the incremental system, with a mean difference in ratings of the two conditions of 0.51 points, which was also highly significant (sign test, 38+/30=/13-; $p < .0007$).[2]

The better pronunciation ratings are especially surprising, as objectively, the synthesis quality of the incremental system can only have been systematically lower than that of the non-incremental system, as the manipulations to synthesis required for incremental processing (and the flaws that existed in the early prototype that was used in the experiment) can only systematically result in a deterioration of the synthesis quality, but not in a systematic improvement. Thus, it appears that participants pardoned bad *synthesis quality* (which occurs in both system versions for certain words) more easily, when overall *formulation quality* is better and even compensate for hesitations that may have been realized rather unnaturally in the incremental system. More to the point: naïve participants do not clearly distinguish between pronunciation and formulation ratings (this is also evidenced by the fact that ratings for the two questions are moderately correlated; Pearson's $r = .537$), and formulation seems to outweigh pronunciation.

Of course, applied systems are most often used by naïve users. Thus, their ratings should matter much more than objective metrics or ratings given by professionals.

## 6. Conclusion

We have built an incremental speech synthesis system that accepts incrementally provided input and we tested it in a domain where this capability allows to integrate multiple, successive events into one complex utterance, and – using preparatory events – allows very timely behaviour. Our experiment shows that the incremental system's formulations are highly preferred over conventional baseline behaviour, even when they involve the introduction of (poorly synthesized) hesitations.

Furthermore, the incremental system's synthesis quality (as captured by the pronunciation rating) was rated as significantly better, despite of modifications that can only have lead to objectively lower quality. However, the speech that was synthesized incrementally was interactionally more *adequate* to the situation of continuous commentary, that is, there were other aspects than voice quality that mattered to the perception of the synthesized speech.

We conclude that synthesis quality may actually matter very little in comparison to *interaction quality*, and that speech synthesis systems should be evaluated in context, or at least taking into account the sorts of interaction behaviour that they support (such as incremental behaviour in our case). In the end, interactive adequacy as a target of speech synthesis optimization may lead to better results more easily than (isolated) perception ratings of synthesized speech samples, without their integration into the relevant context.

Similarly to spoken commentary in a dynamic domain as presented above, conversational speech requires revisions and reactions to external events, such as listener feedback (or the absence thereof) [17, 18]. Thus, we believe that our results, as well as incremental processing in general, also apply to a broad range of conversational synthesis tasks. Finally, the ability to adapt distinguishes incremental speech synthesis from canned speech, which may sound better (seen in isolation), but is completely static and unresponsive to situational demands. Thus, the current success of canned speech in dialogue systems cannot be expected to scale to more interactively advanced dialogue in conversational settings.

## 7. Future work

Our current system is a combination of incremental (vocoding, HMM optimization, top-level integration) and non-incremental strategies (linguistic pre-processing, HMM state selection), which is a compromise owing to the complexity of the full text-to-speech task. However, we plan to extend our system, which is already available as open-source software, to model more of the (phrasal) structure that is generated by linguistic preprocessing in the incremental data structure (cmp. Figure 2). This will allow to e. g. support SSML as incremental input (which is currently unsupported), to support the structured, high-level manipulation of the prosodic realization in real time (i. e. without further re-processing), and allow for a flexible blend of text-to-speech and concept-to-speech techniques in the incremental system.

Modelling higher-level structure will also include modelling *underspecified* higher-level structure, for example the fact that a question is to be synthesized (triggering the appropriate sentence intonation) despite the fact that some specific content is still unknown. In general, there is a trade-off between early specification, and the likelihood of later revision; quality of the system output might improve with explicit models of such likelihoods and corresponding processing adaptations.

**Acknowledgements** The first author would like to thank Petra Wagner and Wolfgang Menzel for fruitful discussions on the topic, and permanent encouragement.

[2] We also conducted a non-paired, two-tailed t-test for pronunciation ratings, as the different formulations of the systems might have effects on pronunciation quality; this test was also significant ($p < .0012$).

## 8. References


[1] T. Dutoit, M. Astrinaki, O. Babacan, N. d'Alessandro, and B. Picart, "pHTS for Max/MSP: A streaming architecture for statistical parametric speech synthesis," Université de Mons, Tech. Rep. 1, 3 2011. [Online]. Available: http://www.numediart.org/docs/numediart_2011_s13_p2_report.pdf

[2] T. Baumann and D. Schlangen, "INPRO_iSS: A component for just-in-time incremental speech synthesis," in *Procs. of ACL System Demonstrations*, Jeju, Korea, 2012.

[3] M. Schröder and J. Trouvain, "The German text-to-speech synthesis system MARY: A tool for research, development and teaching," *International Journal of Speech Technology*, vol. 6, no. 3, pp. 365–377, Oct. 2003.

[4] T. Baumann and D. Schlangen, "The INPROTK 2012 release," in *Proceedings of SDCTD*, Montréal, Canada, 2012.

[5] H. Buschmeier, T. Baumann, B. Dorsch, S. Kopp, and D. Schlangen, "Combining incremental language generation and incremental speech synthesis for adaptive information presentation," in *Proceedings of SigDial*, Seoul, Korea, 2012, pp. 295–303.

[6] T. Baumann and D. Schlangen, "Open-ended, extensible system utterances are preferred, even if they require filled pauses," in *Proceedings of SigDIAL*, Metz, France, Sep. 2013.

[7] T. Baumann and D. Schlangen, "Evaluating prosodic processing for incremental speech synthesis," in *Proceedings of Interspeech*. Portland, USA: ISCA, Sep. 2012.

[8] G. Skantze and A. Hjalmarsson, "Towards incremental speech generation in dialogue systems," in *Proceedings of SIGdial*, Tokyo, Japan, Sep. 2010.

[9] T. Toda and K. Tokuda, "A speech parameter generation algorithm considering global variance for hmm-based speech synthesis," *IEICE transactions on information and systems*, vol. 90, no. 5, pp. 816–824, 2007.

[10] D. Schlangen and G. Skantze, "A General, Abstract Model of Incremental Dialogue Processing," in *Proceedings of the EACL*, Athens, Greece, 2009, pp. 710–718.

[11] H. Kawahara, "Speech representation and transformation using adaptive interpolation of weighted spectrum: vocoder revisited," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., IEEE International Conference on*, vol. 2. IEEE, 1997, pp. 1303–1306.

[12] N. Ström and S. Seneff, "Intelligent barge-in in conversational systems," in *International Conference on Spoken Language Processing (ICSLP)*, Beijing, China, Oct. 2000.

[13] D. L. Chen and R. J. Mooney, "Learning to sportscast: A test of grounded language acquisition," in *Proceedings of 25th International Conference on Machine Learning (ICML-2008)*, Helsinki, Finland, Jul. 2008.

[14] K. Lohmann, M. Kerzel, and C. Habel, "Verbally assisted virtual-environment tactile maps: A prototype system," in *Proceedings of the Workshop on Spatial Knowledge Acquisition with Limited Information Displays 2012*, C. Graf, N. A. Giudice, and F. Schmid, Eds., 2012, pp. 25–30.

[15] M. Kerzel and C. Habel, "Monitoring and describing events for virtual-environment tactile-map exploration," in *Proceedings of Workshop on 'Identifying Objects, Processes and Events', 10th International Conference on Spatial Information Theory*, A. Galton, M. Worboys, and M. Duckham, Eds., 2011, pp. 13–18.

[16] K. Lohmann, O. Eichhorn, and T. Baumann, "Generating situated assisting utterances to facilitate tactile-map understanding: A prototype system," in *Proceedings of SLPAT 2012*, Montreal, Canada, 2012.

[17] H. H. Clark, *Using Language*. Cambridge University Press, 1996.

[18] H. H. Clark, "Speaking in time," *Speech Communication*, vol. 36, no. 1, pp. 5–13, 2002.