

Generating referring expressions with a cognitive model

Markus Guhe (m.guhe@ed.ac.uk)

Department of Informatics, Informatics Forum, 10 Crichton Street
Edinburgh EH8 9AB, Scotland

Abstract

The low computational cost of the incremental algorithm for generating referring expressions makes it an interesting starting point for a cognitive model. But the algorithm has a very limited cognitive adequacy; first of all because of the fact that the preference list (the order in which features are considered for inclusion in the referring expression) is fixed. This means the algorithm does not adapt to feedback of whether a referring expression was used successfully – the algorithm does not learn from past experience. This paper presents a cognitive model in the ACT-R cognitive architecture that simulates the generation of referring expressions in the iMAP task – a task-oriented dialogue. The model takes the feedback it receives from the simulated dialogue partner and adjusts its estimation of the utility of the features it can use in the referring expressions. A property of the task environment in iMAP is that the colour feature is unreliable for identifying referents while other features are reliable. The model correctly simulates the observed human behaviour of decreasing use of colour terms over the course of the dialogues.

Keywords: generation of referring expressions; ACT-R; cognitive modelling; task-oriented dialogue; adaptation; probability matching; rational analysis

Introduction

The incremental algorithm for generating referring expressions by Dale and Reiter (1995) has properties that make it an interesting starting point for a cognitively adequate model. Its two main features are that it has a linear run-time, i.e. a low computational complexity, and that it can produce non-minimal (overspecified) referring expressions. These properties are the main reasons for the large amount of research that this algorithm has inspired. In this paper I will describe an implementation of an algorithm for generating referring expressions that takes the basic ideas from the incremental algorithm but fixes one of its major shortcomings, namely that it chooses features in a fixed sequence, which means that it does not adapt to the demands of the current task or to properties of the task environment.

The incremental algorithm

Dale and Reiter's incremental algorithm is a feature-selection algorithm. This means, to describe the referent (the target object) it selects features that describe the referent and that at the same time do not describe the distractors (the other possible referents). In this way, the algorithm generates a *uniquely distinguishing referring expression*, i.e. a referring expression that refers to only one potential referent in the given situation. The features are chosen according to

the *preference list*, which is an ordered list of features, e.g. *colour, number, size, orientation*.

The algorithm takes the first feature of the preference list, determines the value of this feature for the referent and then checks whether adding the feature value to the referring expression rules out elements in the distractor set. Consider, for example, that the target object is the topmost group of bugs in Figure 1. This means, the other two groups are the distractor set. The algorithm starts with determining *red* as the value of the referent's colour feature. Using *red* in the referring expression rules out the middle group, because it is purple, and the distractor is removed from the distractor set. The algorithm then takes the next feature (number) and determines its value (four). Since the remaining distractor has the value *two*, the algorithm selects the number feature as well. This leaves no distractors, i.e. the algorithm has produced the uniquely distinguishing referring expression: *four, red bugs*. (The kind feature – bugs – is added by default, because referring expressions are usually noun phrases, which syntactically require a noun – which in most cases encodes the kind of feature.)

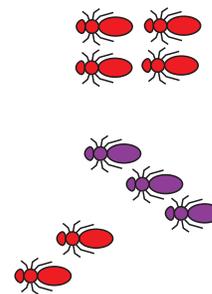


Figure 1: Example of a display for the incremental algorithm

The example also demonstrates how the algorithm generates non-minimal referring expressions, because *four bugs* would be the minimal uniquely distinguishing referring expression. This is an effect of the fact that *colour* was checked before *number*, because it comes first in the preference list. Once the algorithm has selected a feature, it does not reverse its decision. This incremental mode of operation is what gives the algorithm its name and is the main reason for its linear complexity.

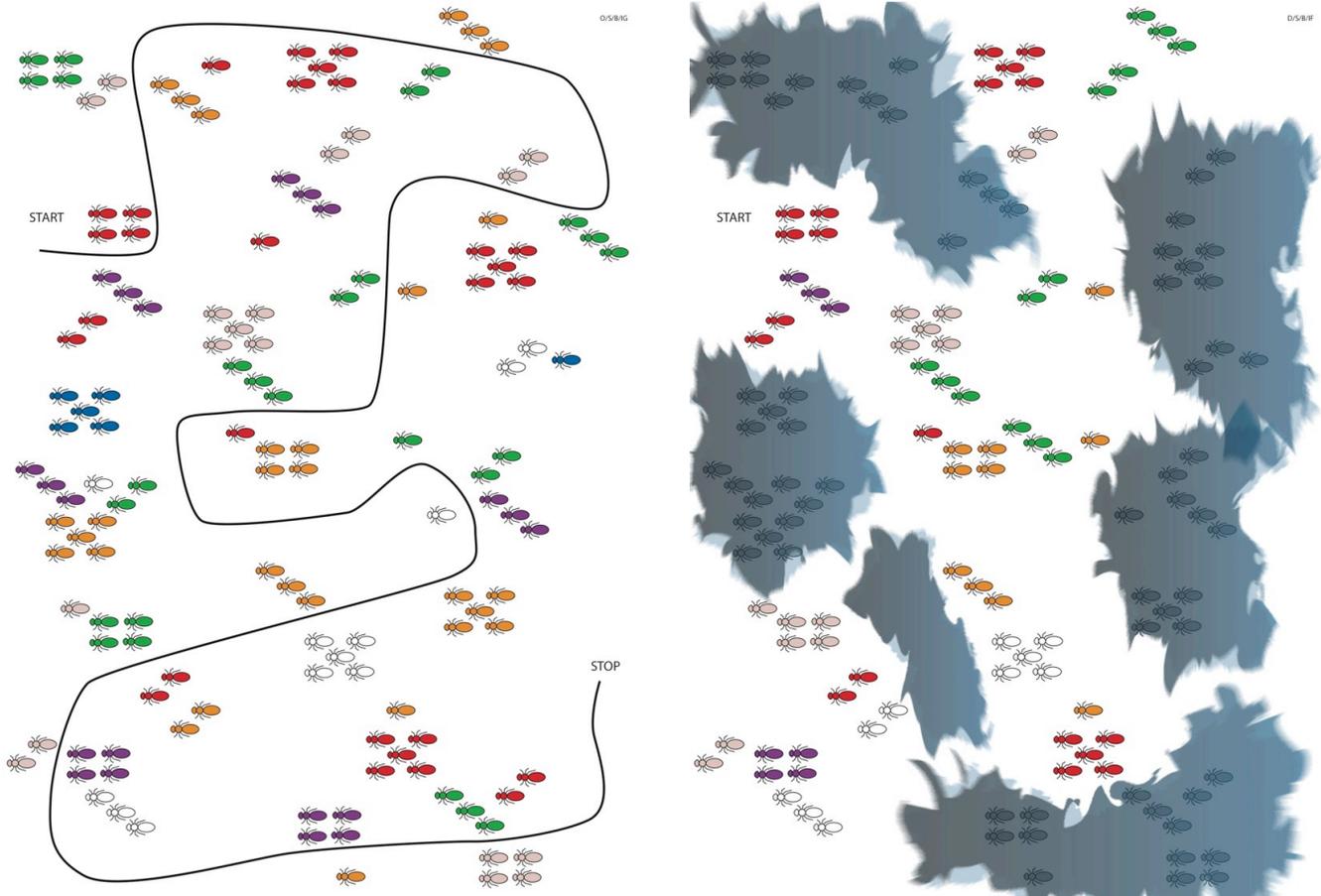


Figure 2: An example map from the iMAP Map Task corpus. The left map is the Instruction Giver's map, the right map the Instruction Follower's. This map was also used for the simulations.

Fixed preference lists

The rationale for the order of features in the preference list is that perceptually salient features like colour are preferred over less salient ones and that discrete features like number are preferred over relative features (e.g., size, orientation). The motivation for preferring salient features is that they are usually useful, because they are easy to identify for both speaker and hearer. (It is, however, open to debate how the different degrees of salience of features are determined.) The motivation for preferring discrete features is that they are easier to process because they do not require to make comparisons. Thus, they require less computational power and are, therefore, more efficient to use.

A major shortcoming of the incremental algorithm is that the preference list is fixed, i.e. it does not change after the system using the algorithm has been started. Thus, the algorithm cannot adapt to the requirements of a task or task environment by learning from past successes or failures of correctly referring to target objects, where *correct* means that the addressee identifies the intended referent on the basis of the referring expression.

The issue of fixed preference lists has previously been addressed by Jordan and Walker (2005). They use machine learning techniques on linguistic corpora to extract the way features are selected for modified versions of the incremental algorithm. Although these algorithms already incorporate some psychological findings like conceptual pacts (Brennan and Clark 1996; Clark 1996), they only provide global adaptations to properties of linguistic corpora and do not account for changes over time and for adaptations to the properties of the task environment.

The iMAP Map Task

The iMAP task, which is a task-oriented dialogue, highlights the limitations of assuming a fixed preference list. This kind of task is a much more natural setting than standard experiments in which participants produce a referring expression for a fixed display, which, after the expression has been produced, is replaced by the next display. The main problem of such settings is that the participant usually receives no feedback whether the expression was used successfully – and if there is feedback it does not specify what caused the

expression to be unsuccessful. Thus, the participants cannot learn from past mistakes or successes.

The iMAP experiment is a modified Map Task (Anderson et al. 1991). The Map Task is an unscripted, task-oriented dialogue in which an Instruction Giver and an Instruction Follower each have a map of the same fictional location. The task is to collaborate to reproduce a route that only exists on the Instruction Giver's map on the Instruction Follower's map, see Figure 2. The dialogue partners are not restricted in their communication.

Materials

Some landmarks, which are the main means of identifying and describing the route, differ between the two maps. In the iMAP task they can differ by:

- Being absent on one of the maps or present on both;
- Mismatching in a feature between the two maps (most notably colour);
- Being affected or not by 'ink damage' that obscures the colour of about half the landmarks on the Instruction Follower's map.

There are four landmark features, each with two levels to distinguish:

- Number (bugs, trees),
- Pattern (fish, cars),
- Kind (birds, buildings),
- Shape: (aliens, traffic signs).

There are three experimental variables:

- Homogeneity: whether the landmarks on a map are of just one kind or whether the landmarks are of different kinds. The two maps in Figure 2 are homogeneous.
- Orderliness: whether the ink blot on the Instruction Follower's map obscures a contiguous stretch of the route (orderly) or a non-contiguous stretch (disorderly). The Instruction Follower's map in Figure 2 is a disorderly map.
- Animacy: whether the landmarks on a map are animate or inanimate (thus, on the non-homogeneous maps there are only landmarks from the 4 inanimate or the 4 animate kinds of landmarks).

Procedure

The participants are told that the maps are 'of the same location but drawn by different explorers'; but they are not told how or where the maps differ. They are instructed to recreate the route on the Instruction Follower's map as accurately as possible. There is no time limit.

Each dyad had to complete two simple training maps and then eight trial maps, one of each kind of landmarks. The maps were counterbalanced with respect to the experimental conditions. After the fourth map, the roles of Instruction Giver and Instruction Follower were swapped.

To reduce the variability of referring expressions each participant had to name a few landmarks that would occur

on the following map and was prompted textually how to name them. Landmarks were not labelled.

Setup and data collection

Participants sat in front of individual computers, facing each other, but separated by a visual barrier.

The communication was recorded using 5 camcorders. Eye gaze was recorded for the Information Giver only using a remote eye tracker. Speech was recorded using a Marantz PMD670 recorder whereby Instruction Giver and Instruction Follower were recorded on two separate (left and right) channels using two AKG C420 headset microphones. The speech was transcribed manually from the recordings. The routes drawn by the Instruction Followers were recorded by his/her computer.

As they were in the same room, participants could hear each other's speech, and they could see each other in the left half of their monitor, which showed the dialogue partner's upper torso video stream. The right half of the monitor showed the map.

Participants and data coding

Sixty-four undergraduates of the University of Memphis participated for course credits.

For the current analysis, the recorded dialogues were transcribed verbatim and all referring expressions were coded for use of colour terms and for terms describing the landmark features (number, pattern, kind, shape). We did not consider the data for the other modalities.

Colour is not useful

In Guhe and Bard (2008a,b) we demonstrated that a feature's utility (usefulness) in a given task environment influences the likelihood that it will be used in a referring expression. More precisely, in the iMAP task, colour had only a limited usefulness, because the colour of landmarks on the Instruction Follower's maps was obscured in half the cases by the ink blots, and colour did not always match between the maps, e.g. a green landmark could be orange on the other map. Thus, colour, which is usually considered to have a high degree of salience and thereby a high utility (and, thus, is one of the first features in the preference list), is an unreliable distinguisher in this particular task environment, because it can only be used successfully in 40% of the cases.

The data from the iMAP experiment show that the utility the dialogue partners attribute to the colour feature changes over time, so that it approaches the utility in the task environment. In short: its usage decreases from ca. 0.6 to ca. 0.2 over the course of the 8 dialogues and also decreases significantly within each of the 8 dialogues. We also showed that this limitation of utility of a feature is restricted to colour. The reliable features (number, pattern, kind, shape) that are the main distinguisher for a particular map do not show

the pattern observed for colour. In fact, the usage of these features increased over the course of a dialogue.

This kind of change in behaviour is a case of *probability matching* (Anderson and Schooler 1991; also called *rational analysis*). Probability matching means that the probability that an agent decides on a particular action matches the probability that the action is successful in the given context. Thus, if action A has a success probability of 0.7 and action B one of 0.3, the agent will choose A in 70% of the cases and B in 30%. Note that the probability only matches, i.e. the agent will not necessarily choose A in those 70% of cases in which action A is actually successful. Note also that in the experiment the usage of the colour feature goes down to 0.2 instead of 0.4 because of the other features that can be used in the referring expressions. See Anderson (2007) or Guhe and Bard (2008b) on how the selection probabilities are computed based on the features' utilities.

Determining distractor sets

In addition to the observed changes in the use of the colour feature, the iMAP data also show that the dialogue partners use a localised way of navigating the maps. More precisely, in the example maps the Instruction Givers followed the routes from START to STOP, using the pattern shown in Figure 3. These areas (corresponding to the 'magic circles' of Guhe 2007b), constitute the distractor sets the Instruction Giver uses. They are established by taking the next landmark or landmarks that the route passes by or through together with the neighbouring landmarks that are (1) closer to the landmarks than to the other landmarks on the map and that are (2) not 'hidden' by another landmark as seen from the current location on the map. The exact cognitive mechanism for this still needs to be modelled but should be along the lines of the proposal by Thórisson (1994).

Cognitive model

To sum up, the iMAP data show that

- there is a change in the use of colour terms over time that adheres to the probability matching principle;
- the distractor sets are groups of landmarks close to the target referent. (This corresponds to the focussed elements and local contexts of Guhe 2007a.)

The cognitive model for generating referring expressions that takes these findings into account is realised in ACT-R (Anderson, 2007). It follows the basic structure of the incremental algorithm, but (1) it does not have a fixed preference list, and (2) the distractor sets are not given to the model but are determined by the model itself.

ACT-R is a hybrid cognitive architecture, i.e. it is a production system with a subsymbolic layer. For the purposes of this paper it suffices to know that each production has a utility value that is a measure of how often the production was fired (used) successfully in the past. To decide which production to fire next, ACT-R creates the conflict set (con-

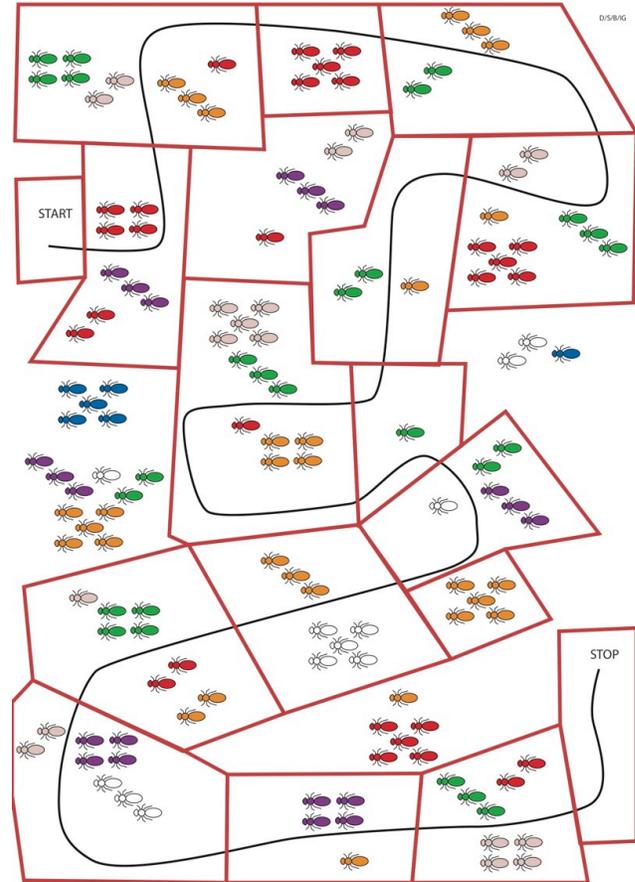


Figure 3: The example map from the iMAP Map Task corpus used for the simulations and the assumed magic circles (distractor sets)

sisting of all currently applicable productions), determines their selection probability (how likely will a production be successful in comparison to the other productions in the conflict set) and chooses a production according to this probability.

The model is a model of the Instruction Giver and only produces initial referring expressions, i.e. expressions that mention a landmark for the first time, to exclude effects of repeated mentions like conceptual pacts and reduced expressions (Guhe and Bard 2008b).

Distractor sets

Determining distractor sets is not yet implemented, because it requires an appropriate model of visual search, i.e. a model of what landmark to focus on next. However, this is not important for the current model. What is relevant is that the distractor sets are not available to the model as wholes, i.e. there is no chunk (entity, concept) in memory that encodes the distractor set as such. Instead (as a preliminary solution), the model knows the 'next' landmark for any given landmark, viz. the landmark that the visual system would focus on next, given a current landmark.

Adaptations

As in Guhe and Bard (2008b), the choice of whether to use colour is based on the utility that the model ascribes to it. More precisely, there are productions that propose using a particular feature (colour, number, shape, etc.). The utilities of these productions drive the change in usage of the colour features. (These productions correspond to the ‘productions’ modelled in Guhe and Bard, 2008b.)

Algorithm

The model follows the following steps:

1. get next target object (if available)
2. propose feature
3. get feature value for target
4. get next distractor
5. if feature value rules out distractor → 4
if feature value does not rule out distractor → 2
6. if next distractor is the target object (referring expression is complete) → propagate reward;
otherwise → execute step 4
7. execute step 1

The only productions that are competing in this model are the ‘propose’ productions of step 2. Their utilities change according to the reward that is propagated in step 6. (Actually, the reward affects all productions that fired since the last reward was propagated, but as no other productions compete with each other the propose productions are the only ones showing an effect.) If colour is used in the referring expression, the model receives a positive reward in 40% of the cases, a negative one otherwise, reflecting the probability that the Instruction Follower knows the colour of the landmark (the referent).

Results

The comparison between the iMAP data and the simulations carried out with the model are shown in Figure 4. The green curve shows how often the model proposed to use colour as an average of 500 simulations for the ‘bug’ map, i.e. the map containing only bugs as landmarks. The data only show introductory mentions of landmarks by the Instruction Giver for the first map (cf. Guhe and Bard 2008b). The correlation between the experimental data and the model is not significant ($\beta_1 = 0.2897$, $R^2 = 0.0839$, $F(1, 27) = 2.47$, $p > 0.05$).

The reason for this is quite obvious: the model curve has many outliers. A comparison with the corresponding landmarks on the test map shows that colour is never chosen for the target landmarks with an empty distractor set, e.g. the five red bugs in the top middle, and that colour is always chosen in the one case where it is actually required to make a uniquely distinguishing referring expression, namely for the two red bugs to the top left of two orange bugs in the lower left of the map.

Two factors contribute to this. Firstly, only considering magic circles to be distractor sets is too simple, because in

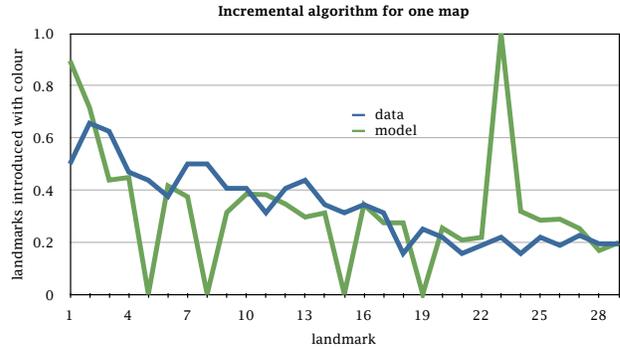


Figure 4: Results of the model simulations

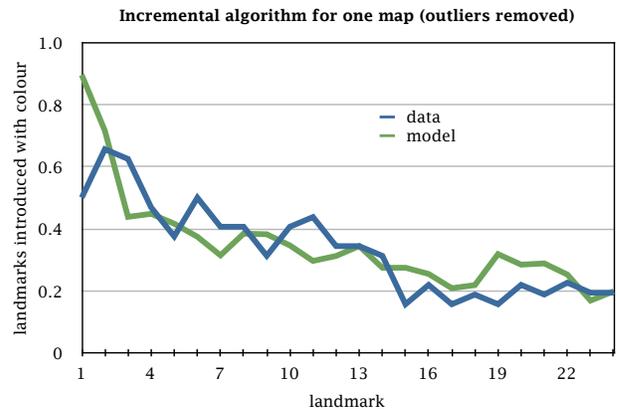


Figure 5: Model simulations without outliers

the cases where there is only the target object in the magic circle, the model just generates *the bugs*, which does not correspond to the expressions in the iMAP corpus. (And with good reason, too, because an instruction like *Go above the bugs* is clearly not precise enough.) Secondly, the human data show the landmarks in the order in which they were introduced, which is not necessarily the (currently fixed) order in which the model encounters them.

Once these outliers are removed, the model performs at the same level as the original one in Guhe and Bard (2008b), cf. Figure 5 ($\beta_1 = 0.7240$, $R^2 = 0.5242$, $F(1, 22) = 24.24$, $p < 0.001$). Using only the data for the ‘bug’ type map shows (literally) identical correlations, which shows that this map is a good test case.

A first extension of the base model

The previous analysis shows that the model’s main limitation is that it does not adequately produce overspecified or underspecified referring expressions. Humans tend to produce overspecified expressions in the case where the distractor set is empty and underspecified expressions in the case where many features (well, 2 in this case) would be

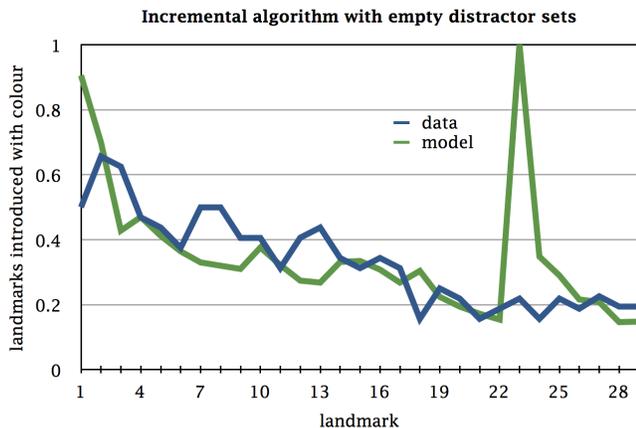


Figure 6: Model simulations with special treatment of empty distractor sets

needed to produce a uniquely distinguishing referring expression.

A straightforward extension of the model, therefore, consists in providing special treatments for these cases, which are not addressed by the incremental algorithm. The simpler of the two cases is to provide overspecified expressions for the case of empty distractor sets. Figure 6 shows the performance of a model that is extended by additional productions that select one feature in this case. Thus, the minimal expressions that are produced by this model are of the form *the five bugs* or *the red bugs*. Despite a substantial improvement in the correlation between model and data the remaining outlier means that the correlation is less than in the original model ($\beta_1 = 0.437$, $R^2 = 0.191$, $F(1, 27) = 6.38$, $p < 0.001$).

Conclusions and future work

Despite obvious shortcomings, the model provides a number of important conclusions. Firstly, the main ideas of the incremental algorithm (feature-selection, incremental production with linear complexity) are suitable assumptions for a cognitive model of generating referring expressions. Secondly, by using ACT-R, the fixed preference list of the incremental algorithm can be replaced by productions that propose features. The probabilities (preferences) with which these productions fire change as a function of the interactions with the task environment. Thirdly, the model accounts for the empirical data in a straightforward manner.

Using cognitive modelling allows to integrate results from computational linguistics (incremental algorithm) and psycholinguistics (iMAP task) in a straightforward way. What is more, the explanatory mechanisms used are not specific to this phenomenon but are general mechanisms of cognition.

Finally, the modelling process itself highlights necessary improvements. A first extension of the basic model shows how the human tendency of using overspecified expressions can be easily integrated into the model. Another necessary

extension is to account for underspecified expressions as well. Finally, to make the most of the model it should include the visual search processes that establish the distractor sets – an extension for which ACT-R again is a good choice because of its visual module that has been used in numerous models for very similar tasks.

Acknowledgements

This research was supported by the NSF (grant NSF-IIS-0416128) and by the EPSRC (grant EP/F035594/1).

References

- Anderson, A. H., Bader, M., Bard, E. G., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., Sotillo, C., Thompson, H. S. & Weinert, R. (1991). The HCRC Map Task corpus. *Language and Speech*, 34(4), 351–366.
- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.
- Anderson, J. R. & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science*, 2(6), 396–408.
- Brennan, S. E. & Clark, H. H. (1996). Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(6), 1482–1493.
- Clark, H. H. (1996). *Using Language*. Cambridge, MA: Cambridge University Press.
- Dale, R. & Reiter, E. (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2), 233–263.
- Guhe, M. (2007a). *Incremental conceptualization for language production*. Mahwah, NJ: Lawrence Erlbaum.
- Guhe, M. (2007b). Marking theme and rheme in preverbal messages. In: *Proceedings of the Seventh International Workshop on Computational Semantics IWCS-7, January 10–12, 2007* (pp. 330–333). Tilburg: Tilburg University.
- Guhe, Markus & Bard, E. G. (2008a). Adapting referring expressions to the task environment. In: *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (pp. 2404–2409). Austin, TX: Cognitive Science Society.
- Guhe, M. & Bard, E. G. (2008b). Adapting the use of attributes to the task environment in joint action: Results and a model. In: *Proceedings of Londial – The 11th Workshop on the Semantics and Pragmatics of Dialogue* (pp. 91–98).
- Jordan, P. W. & Walker, M. A. (2005). Learning content selection rules for generating object descriptions in dialogue. *Journal of Artificial Intelligence Research*, 24, 157–194.
- Thórisson, K. R. (1994). Situated perceptual groupings: An application to human–computer interaction. In: *Proceedings of the 16th Annual Conference of the Cognitive Science Society* (pp. 876–881).