Smaller, Faster Agent Dialogues via Conversational Probing

Tim Menzies[‡], Bojan Cukic[¶], Enrico Coiera[†]
[‡]NASA/WVU Software Research Lab, 100 University Drive, Fairmont, USA
[¶]Department of Computing Science and Electrical Engineering, West Virginia University, Morgantown, WV
[†]Department of Computer Science and Engineering, University of New South Wales, Australia 2052
<tim@menzies.com, cukic@csee.wvu.edu, ewc@pobox.com>

Abstract

Studies of human conversation suggest that agents whose world models are in consensus can work well together using only very narrow bandwidths. The total bandwidth required between agents could hence be minimised if we could recognise when model consensus breaks down. At the breakdown point, the communication policy could switch from some usual-case low value to a temporary high value while the model conflict is resolved. To effectively recognise the breakdown point, we need tools that recognise model conflicts without requiring extensive bandwidth. A mathematical model of probing and-or graphs suggests that, for a large range of interesting models, the number of probes required to detect consensus breakdown is quite low.

Keywords: Agent communication, Negotiation protocols, Negotiating to maintain coalitions, Facilitating the negotiation process

Agents running on distributed computers may interact under a variety of resource restrictions. One restriction is bandwidth. Every transmission also takes time since it must be processed by both the sender and the receiver. Studies of human communication suggests that failure to optimise the use of available resources can result in inefficient communication within organisations and consequent task errors (Coiera 1996). How can we minimize the use of those resources for agents, and maximise the likelihood that communication tasks are successfully completed?

Consider two humans communicating via a channel with a certain bandwidth. A robust finding from human conversations is that restricting bandwidth has little impact on the effectiveness of the outcomes of many collaborative problem solving tasks (McCarthy & Monk 1994). This occurs when the communicating individuals have a high degree of shared common ground. A particularly striking finding is that humans assume they share common world views in conversation until they detect an error, and then attempt to probe the cause of the error. Interestingly, human conversation seems to consist of both an explicit small set of exchanges devoted to testing or confirming understanding, as well as the primary exchange (J Clarke 1991).

Can we use these observations to minimize the bandwidth required between computational agents? Suppose agents move through alternating stages of assumed model consensus, detection of model conflict, and then model repair. While agent models are in conflict, we broaden the bandwidth between them. When models are in consensus, we restrict the bandwidth by assuming shared knowledge between agents. As a result, bandwidth is conserved during conversation until it is explicitly demanded to reconcile the world models of conversing agents. Time is also conserved by minimising unnecessary sharing of knowledge between agents (Coiera 1999).

This "assume-consensus" assumption has another advantage. Consider agents with a learning component. Such agents may update their world model. If they wish to coordinate with other agents, then they would need to reflect on the beliefs of their fellows. In the worst case, this means that each agent must maintain one belief set for every other agent in its community. This belief set should include what the other agent thinks about all the other agents. If that second-level belief set includes the original agents beliefs, an infinite regress may occur (e.g. "I think he thinks I think that he thinks that I think that.."). This "assume-consensus" model addresses the infinite regress problem. Agents in assumed consensus only need to store their own beliefs and one extra axiom; i.e. "if I believe that she believes what I do, then my beliefs equals her beliefs".

To apply this approach, agents have to continually test that other agents hold their beliefs. One method for doing this would be to ask agents to dump their belief sets to each other. We consider this approach impractical for three reasons:

- It incurs the penalty of the infinite regress, discussed above.
- Such belief-dumps could exhaust the available bandwidth.
- Agents may not wish to give other full access to their internal beliefs. For example, security issues may block an agent from one vendor accessing the beliefs of an agent from another vendor.

Without access to internal structures, how can one agent assess the contents of another? Software engineering has one answer to this question: black-box specification-based probing (Hamlet & Taylor 1990; Lowrey, Boyd, & Kulkarni 1998). Agent-A could log its own behaviour to generate a library of input-output pairs. In doing so, Agent-A is using itself as a specification of the expected behaviour of Agent-B, assuming our two agents are in consensus. If Agent-A

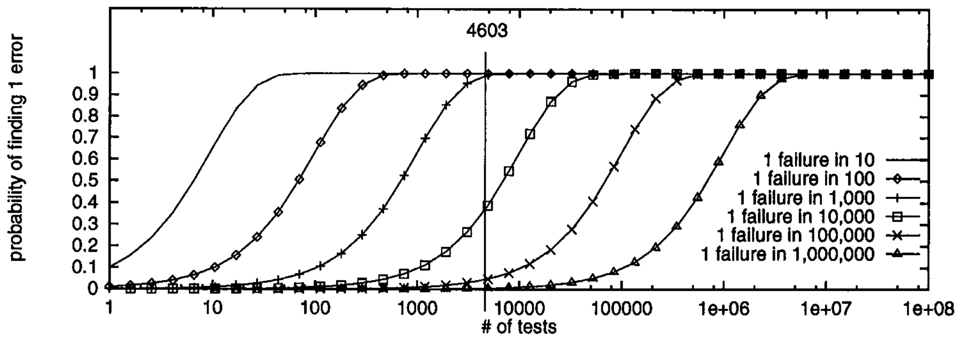

Figure 1: $C = 1 - (1 - \alpha)^N$. Theoretically, 4603 probes are required to achieve a 99% chance of detecting moderately infrequent items; i.e. ones that only occupy which $\frac{1}{1000}$-th of the model. (Hamlet & Taylor 1990).

sees those outputs when Agent-B is presented with those inputs, then Agent A could infer that Agent B has the same beliefs as itself.

Unfortunately, black-box probes can be very bandwidth expensive. Equation 1 shows a widely-used statistical model connecting the number of probes N to the confidence C that we will find an error with frequency of occurrence α.

$$C = 1 - (1 - \alpha)^N \tag{1}$$

Equation 1 says that for probes to be 99% certain of detecting anomalies, then the number of black-box probes required is 4.6 times divided by the frequency of that anomaly (see Figure 1). For example, to be 99% certain that Agent-B has less than 1% difference in its beliefs to Agent-A, then 460 probes would be required. This statistical model is hence very pessimistic on the possibility of Agent-A accurately assessing its consensus with Agent-B, without using large bandwidths for the probing.

This paper will argue that Figure 1 is a gross over-estimate of the number of probes. Figure 1 assumes no knowledge of the internal structure of our agents. If we commit to some view of structure, then estimates can be generated for the odds of finding differences between Agent-A and Agent-B. Hence, assuming that agents are usually in consensus, then we can reduce inter-agent bandwidth as follows:

- Assume consensus and restrict bandwidth.
- As part of the normal inter-agent dialogue, Agent-A occasionally drops a probe question to Agent-B. If the number of probes required is very small, then these process will add little to the overall bandwidth.
- If conflicts are detected, increase the bandwidth between Agent-A and Agent-B to allow for discussions.

The rest of this paper is structured as follows. First, we present a simple mathematical model of the odds of reaching some random node in an and-or graph from some inputs. Secondly, we present simulation results suggesting that the average odds of finding a random node has only one of two behaviours:

- The odds quickly asymptotes to a high plateau
- The odds quickly fall to nearly zero.

The simulation model also offers clear guidelines as to which of these two behaviours will occur. That is, when we design agents, the simulation model can indicate what system features allow agents to quickly assess each other.

Two caveats before continuing. This paper presents a probabilistic model of how agents of a particular form (an and-or graph) can assess each other. Hence:

- It offers average case results which will be inaccurate in certain circumstances. For mission-critical systems, our average-case analysis is hence inappropriate.
- The psychological likelihood of this model is questionable and we should not use these results to make statements about how humans should co-ordinate[1].

A Model of Probing

Roughly speaking, probing is a process of finding a needle in a haystack. What are the odds of finding some random needle? To answer this question, we have commit to some model of a program. The following model applies to any program which can be reduced to a directed and-or graph between concepts. For example, a propositional rule base is such an and-or graph where the primitive concepts are (e.g.) *a=true* or *a=false*.

Graphs

We represent a program a directed-cyclic graph G containing vertices $\{V_i, V_j, ...\}$ and edges E. G has roots $roots(G)$ and leaves $leaves(G)$. A vertex is one of two types:

- *Ors* can be believed if any their parents are believed or it has been labeled an In vertex (see below). The average number of parents for the ors is

$$parents_{or} = \frac{|parents(ors(G))|}{|ors(G)|} \tag{2}$$

Each or-node contradicts 0 or more other or-nodes, denoted $no(V_i) = \{V_j, V_k, ...\}$. The average size of the *no* sets for all V_i is denoted $constraints(G)$.

[1]Exception: in the special case where humans are collaborating to test software, these results could be used to optimise their testing efforts.

- *Ands* can be believed if *all* their parents are believed. The average number of parents for the ands is

$$parents_{and} = \frac{|parents(ands(G))|}{|ands(G)|} \tag{3}$$

Proofs

The aim of our probing is to generate proofs across G to some output goal Out_i. Out_i comes from a test suite. A test suite is the database $\{< In_1, Out_1 >, ... < In_n, Out_n >\}$. We also assume that each In_i, Out_i contains only or-vertices. Each output Out_i can generated 0 or more proofs $\{P_x, P_y, ...\}$. We make no other comment on the nature of Out_i: it may be some undesirable state or some desired goal.

In and Out are sets of vertices from G. A proof $P \subseteq G$ is a tree containing the vertices $uses(P_x) = \{V_i, V_j, ...\}$.The proof tree has:

- Exactly one leaf which is an output; i.e.

$$|leaves(uses(P_x))| = 1 \wedge$$
$$V_i \in leaves(uses(P_x)) \wedge V_i \in Out$$

- 1 or more roots $roots(uses(P_x)) \subseteq In$.
- Height $height(P_x)$ being the largest pathway from the leaf to the roots.

When growing a proof, a new vertex is added to the set of vertices already on that proof. The new vertex must not contradict the vertices already in the proof; that is, the new vertex V_{new} must not satisfy:

$$V_{old} \in uses(P_x) \wedge V_{new} \in no(V_{old}) \tag{4}$$

Odds of Finding Out_i

Based on the above, we can compute $OddsR_H$, the odds of reaching some arbitrary output $V_i \in Out$ use a proof tree of height H. The section assumes that the probability that a new vertex does not conflict with the *no* sets already in a proof with odds $OddsOK_H$ (this probability is computed later).

An output in a tree of height $H = 1$ can only be believed if it is also an input. Only or-vertices can be inputs, so:

$$OddsR_1^{or} = \frac{|In|}{|V|} * OddsOK_1^{or} \tag{5}$$

$$OddsR_1^{and} = 0 \tag{6}$$

Otherwise, for $H > 1$:

- If V_i is an and-vertex, then we believe it with the probability of believing all its parents; i.e.

$$OddsR_H^{and} = (OddsR_{H-1})^{parents_{and}} * OddsOK_H^{and} \tag{7}$$

- If V_i is an or-vertex, then we believe it with the probability of believing any of its parents; i.e.

$$OddsR_H^{or} = (parents_{or}) * OddsR_{H-1} * OddsOK_H^{or} \tag{8}$$

From Equations 7 and 8, we make the following predictions:

Prediction 1 *If and-nodes dominate G, then OddsR will decrease very rapidly. If or-nodes dominate G, then OddsR will increase very rapidly*[2].

Also, from Equation 5 and 8 it follows that:

Prediction 2 $OddsR_H \propto \frac{|Ins|}{|V|}$

The probability of believing some arbitrary V_i is some combination of

- The probability that it is an and-vertex and it is believed.
- The probability that it is an or-vertex and it is believed.

For the moment, we will assume that the combining function is the maximum of the two probabilities (and experiment with this later). That is:

$$OddsR_{H>1} = max\left(\begin{array}{c}\frac{|ands(G)|*OddsR_H^{and}}{|V|},\\ \frac{|ors(G)|*OddsR_H^{or}}{|V|}\end{array}\right) \tag{9}$$

The maximum value for $OddsR_H$ is when $OddsR_H^{and}$ or $OddsR_H^{and}$ is 1, i.e.

$$OddsR_{H>1} = max\left(\begin{array}{c}\frac{|ands(G)|}{|V|},\\ \frac{|ors(G)|}{|V|}\end{array}\right)$$

Hence:

Prediction 3 *We expect $OddsR_H$ to be asymptotic to some percentage of the ratio of ands/ors in G.*

Odds of $OddsOK_H$

Let $OddsOK_H$ be the probability of the one world assumption; i.e. the odds that a new vertex at height H can be added into the current proof of height $H-1$ without contradicting anything else in that proof.

At $H = 1$, we are adding a vertex into an empty proof; hence, the chances of a new vertex contradicting existing proof vertices is zero and:

$$OddsOK_1 = 1 \tag{10}$$

And-vertices have no *no* set. Hence for all H:

$$OddsOK_H^{and} = 1 \tag{11}$$

For $H > 1$, the odds of an or-vertex contradicting another or-vertex on the proof is the compliment of the sum of the number of or-vertices on that proof times the frequency of constraint violations:

$$OddsOK_H^{or} = 1 - \left(\left(\sum_{1..H-1}^{i} OddsR_i^{or}\right) * \frac{constraints(G)}{|V|}\right) \tag{12}$$

Prediction 4 *Unless $constraints(G)$ is a very large fraction of $|V|$, then OddsOK will be very large.*

[2]Note that this prediction, and the ones to follow, are not certain inferences since feedback factors may influence our results.

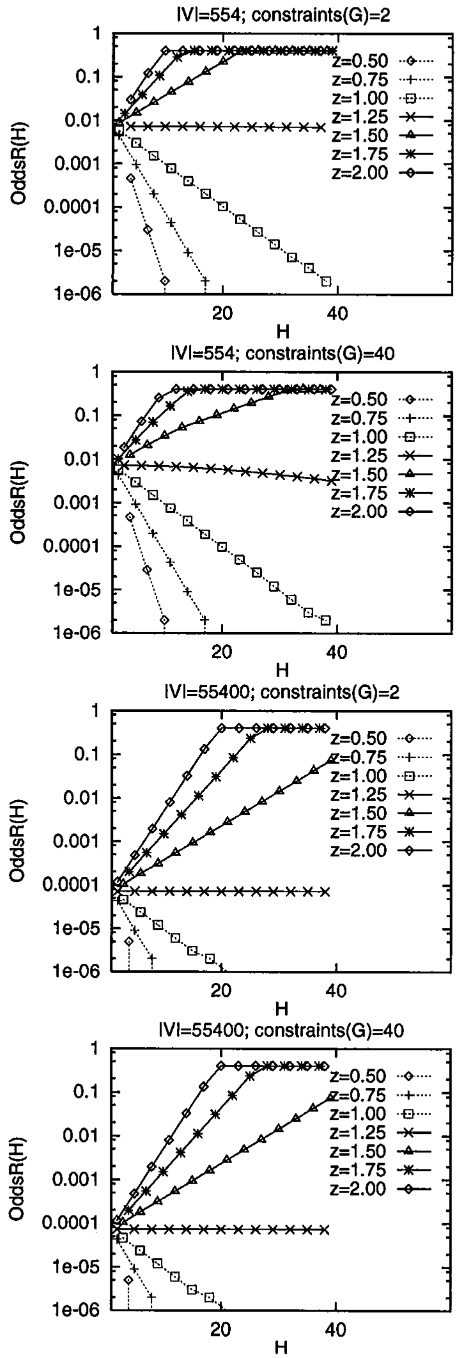

Figure 2: Simulation results at $y = 0, repeats = 1$.

$\lvert V\rvert$	=	554 .. 55400
$\lvert In\rvert$	=	4
$constraints(G)$	=	2 .. 40
$\lvert ands(G)\rvert$	=	$0.6 * \lvert V\rvert$
$\lvert ors(G)\rvert$	=	$\lvert V\rvert - \lvert ands(G)\rvert = 0.4$
$parents_{and}$	=	$normal\_dist(\mu_{pands}, \sigma_{pands})$
$parents_{or}$	=	$normal\_dist(\mu_{pors}, \sigma_{pors})$
μ_{pands}	=	2
μ_{pors}	=	$z * \mu_{ands}$
σ_{pands}	=	$y * \mu_{pands}$
σ_{pors}	=	$y * \mu_{pors}$
y	=	0 .. 0.5
z	=	0.5 .. 2.0
$repeats$	=	1..20

Table 1: Simulation parameters

Simulations

Table 1 shows parameters for an and-or graph from the real-world model of neuroendocrinology (Smythe 1989) explored previously by Menzies & Compton (Menzies & Compton 1997). Initial experiments with our formulae assumed $y = 0, repeats = 1$; i.e. the number of parents of and-nodes and or-nodes was constant. The results of the $y = 0, repeats = 1$ run are shown in Figure 2. Note that for a range of graph sizes and number of constraints, the same pattern emerges:

- Confirming predictions 1 and 2, when or-nodes dominate (above $z = 1.25$), the odds of reaching some random nodes asymptotes to the percentage of or-nodes in the graph (in these simulations, $\frac{ors(G)}{|V|} = 0.4$). When and-nodes dominate (below $z = 1.25$), the odds of reaching a node drops dramatically.
- Confirming prediction 3, $OddsR_1$ of the top two plots of Figure 2 are 100 times larger than $OddsR_1$ of the bottom two plots. This follows since the ratio of $\frac{|In|}{|V|}$ decreased by a factor of 100 when $|V|$ moved from 554 to 55400 and $|IN|$ remained constant.
- Confirming prediction 4, an order of magnitude change in $constraints(G)$ had little effect on $OddsR_H$ since $(constraints(G) = 40) \ll (|V| = 554)$.
- In the case of the asymptotic effect, after a certain proof depth, the odds of reaching a node are not improved by further searching. In these simulations, that point was $H = 10$ (for small theories with $|V| = 554$) and $H = 20$ (for large theories with $|V| = 55400$).

In summary nodes are either very reachable or barely reachable, depending on the average number of and-node parents and the average-number of or-node parents. Hence, if we design our agents with more or-parents than and-parents, these results suggest that the agents will be able to check for consensus very easily. Figure 3 is an expansion of the far left-hand-side of Figure 1 and shows that (e.g.) 4 probes gives a 87% confidence of finding a node when $OddsR_H = 0.4$. 4 probes is a very small overhead to an

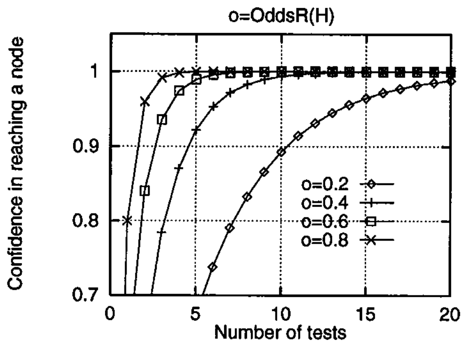

Figure 3: $C = 1-(1-OddsR_H)^N; 0.2 \leq OddsR_H \leq 0.8$.

on-going conversation. On the other hand, if we make the internal logic of our agents very complex (more and-nodes than or-nodes) then it will be very hard to monitor consensus without requiring many probes and a very large bandwidth.

External Validity

This section explores the generalizability of the above results.

There are at least three challenges to the above results. Firstly, they assume a simple model of the graph being processed; i.e. uniform distributions of and-parents and or-parents. How sensitive are the above findings to changes in the distributions? To explore this issue, the runs of Figure 2 were repeated several times with increasing variance in the distributions on node parents. In terms of Table 1, this meant runs with $repeats = 20, y = 0..0.5$. The results of these runs is shown as:

- The standard error on the mean of $OddsR$...
- Expressed as a percentage of the mean (e.g. a 50% standard error means that our expectation of X varies from $0.5X$ to $1.5 * X$).

Large deviations were only noted for $H > 50$ (see the curves marked $H < 150, H < 125, H < 100, H < 75$ in Figure 4). Returning to Figure 2, note that by $H > 50$, the odds have either risen to their plateau or fallen away to be vanishingly small. That is, increasing the variances (by changing y) only effects our results in uninteresting regions.

Secondly, the above runs used Equation 9, which made the optimistic assumption that

$$OddsR_H = maximum(OddsR_H^{and}, OddsR_H^{or})$$

What happens if we reason pessimistically, i.e.

$$OddsR_H = minimum(OddsR_H^{and}, OddsR_H^{or})$$

To study this, a parameter λ was introduced:

- $\lambda = 0$ implies Equation 9 uses minimums;
- $\lambda = 10$ implies Equation 9 uses maximums;
- $1 \leq \lambda \leq 9$ moves Equation 9 on a linear sliding scale between minimum and maximum.

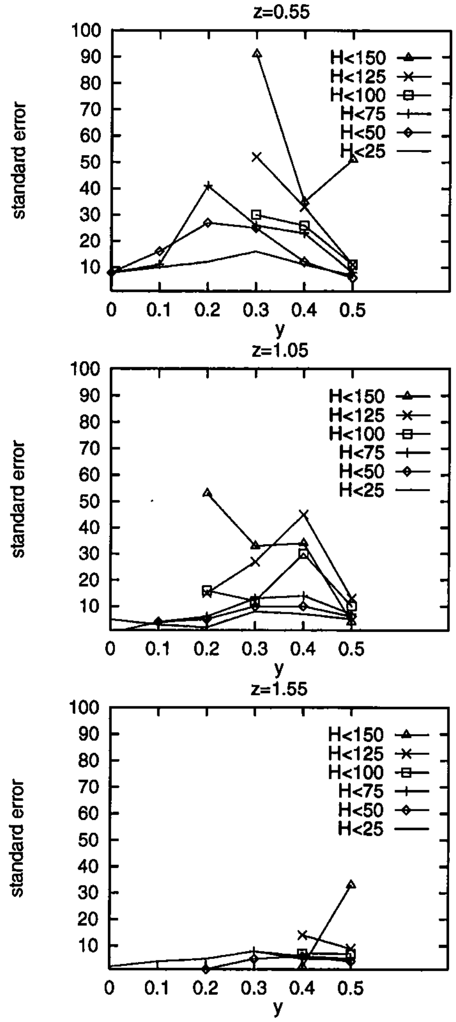

Figure 4: Simulation results at $y = 0..0.5, repeats = 20$.

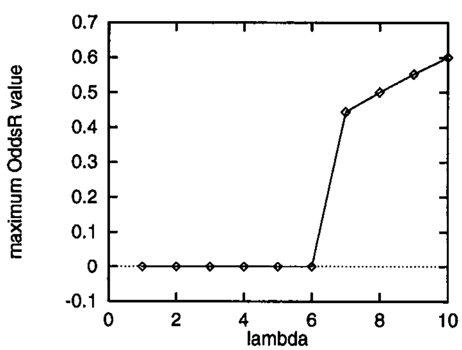

Figure 5: Combining via minimization (left) to combining via maximization (right).

The run of Figure 2 was repeated for $|V| = 554$ and $constraints(G) = 2$. The results (see Figure 5) show that minimization, or even averaging ($\lambda = 5$), is inappropriate. Only when we nearly maximize ($\lambda > 7$) do we get any behaviour at all (i.e. maximum $OddsR_H$ rises above zero).

Thirdly, we could refute Figure 2 by showing that it does not accurately reflect the behaviour of known search engines. Two predictions can be generated from Figure 2:

- As z is altered, the odds of reaching solutions switches suddenly from very low to very high (this was observed at the $z = 1.25$ point).
- Recalling the discussion around Figure 3, Figure 2 is saying that a small number of probes should reach as many nodes as a large number of probes.

Both these behaviours has been noted in the literature:

- *Sudden phase transitions in solvability*: In many NP-hard problems, it has been noted that there exist very narrow regions around which problems switch from being easily solvable to easily unsolvable (Cheeseman, Kanefsky, & Taylor 1991).
- *A small number of probes suffices*: HT4 is a multi-world reasoner that searches for worlds that contain the most known behaviour of a system. HT4 explores all assumptions that could potentially lead to desired goals. HT0 is a cut-down version of HT4 which, a small number of times, finds one random world. In millions of runs over tens of thousands of theories, HT0 reached 98% of the desired goals as HT4 (Menzies 1999). Dozens of analogous results showing that a small number of probes suffice have been reported in the software engineering and knowledge engineering literature (Menzies & Cukic 1999).

Discussion

If we make some assumptions about the internal structure of an agent, we can generate estimates of how many probes are required to test if that structure conforms to an external specification. In the case of and-or graphs, the total number of probes is either very large or very small depending on certain design choices (average number of and-node parents and or-node parents).

A partial list follows of language features that generate theory dependency graphs with many or-nodes:

- Disjunctions.
- Any indeterminacy, such as a conditional using a randomly generated number and a threshold comparison.
- Polymorphism: one "or" would be generated for each type in the system that is accessed by this polymorphic operator.
- Calls to methods implemented and over-ridden many times in a hierarchy: one "or" would be generated for each possible received of the message.

A clear research direction from this work is the further definition of language features that simplify the time required to check consensus amongst agents.

Acknowledgements

This work was partially supported by NASA through co-operative agreement #NCC 2-979. Discussions with Goran Trajkovski prompted the λ study.

References

Cheeseman, P.; Kanefsky, B.; and Taylor, W. 1991. Where the really hard problems are. In *Proceedings of IJCAI-91*, 331–337.

Coiera, E. 1996. Communication in organisations. Technical report, Hewlett-Packard Laboratories. Technical Report HPL-96-65, May, 1996.

Coiera, E. 1999. Communication under scarcity of resources. Technical report, Hewlett-Packard Laboratories. Technical Report, 1999, to appear.

Hamlet, D., and Taylor, R. 1990. Partition testing does not inspire confidence. *IEEE Transactions on Software Engineering* 16(12):1402–1411.

J Clarke, S. B. 1991. Grounding in communication. In *Perspectives on Socially Shared Cognition*. American Psychological Association.

Lowrey, M.; Boyd, M.; and Kulkarni, D. 1998. Towards a theory for integration of mathematical verification and empirical testing. In *Proceedings, ASE'98: Automated Software Engineering*, 322–331.

McCarthy, J. C., and Monk, A. F. 1994. Channels, conversation, co-operation and relevance: all you wanted to know about communication but were afraid to ask. *Collaborative Computing* 1:35–60.

Menzies, T., and Compton, P. 1997. Applications of abduction: Hypothesis testing of neuroendocrinological qualitative compartmental models. *Artificial Intelligence in Medicine* 10:145–175. Available from `http://www.cse.unsw.edu.au/~timm/pub/docs/96aim.ps.gz`.

Menzies, T., and Cukic, B. 1999. Intelligent testing can be very lazy. NASA/WVU IV&V tech report. Submitted to ISE'99.

Menzies, T. 1999. Simpler, faster abductive validation. *AAAI-99 (submitted)*.

Smythe, G. 1989. Brain-hypothalmus, Pituitary and the Endocrine Pancreas. *The Endocrine Pancreas*.