Efficient Cross-Cultural Models for Communicative Agents

Alicia Sagae[1], *Emily Ho*[1], *Jerry R. Hobbs*[2]

[1]Alelo Inc.
Los Angeles, CA, USA
{asagae, eho}@alelo.com

[2]Information Sciences Institute
University of Southern California
Los Angeles, CA, USA
hobbs@isi.edu

ABSTRACT

This paper presents a training system that uses compositional models of culture for social simulations involving conversational agents. We compare the compositional framework to a state-of-the-art agent system, in terms of development effort, number of reused and new objects, and flexibility and accuracy of resulting conversational simulations. Resulting trends indicate that the new architecture is more efficient, especially as the number of simulations grows.

Keywords: Hybrid Modeling, Training & Simulation, Conversational Agents, Commonsense Models

1 INTRODUCTION

Cross-cultural competency is a critical need for military personnel. For example, the US Defense Regional and Cultural Capabilities Assessment Working Group has identified the ability to integrate cultural knowledge and skills into mission execution as a critical cross-cultural competency for general purpose forces

(McDonald, et al., 2008). Training of these skills, knowledge, and abilities is resource-intensive for both trainees and organizations. Simulation-based training promises anytime, anywhere access that can allow instructional material designed by a single trainer to be delivered in an effective, interactive way to thousands of trainees at lower cost and higher convenience (Fletcher, 1990). However, when instructors and domain experts encode this material, the current tools offered to them typically produce script-like, monolithic data structures that are culture-specific, non-reusable, and difficult to update or apply to new cultures and missions. As a result, creating training scenarios is costly and inefficient, especially as the number of scenarios grows large.

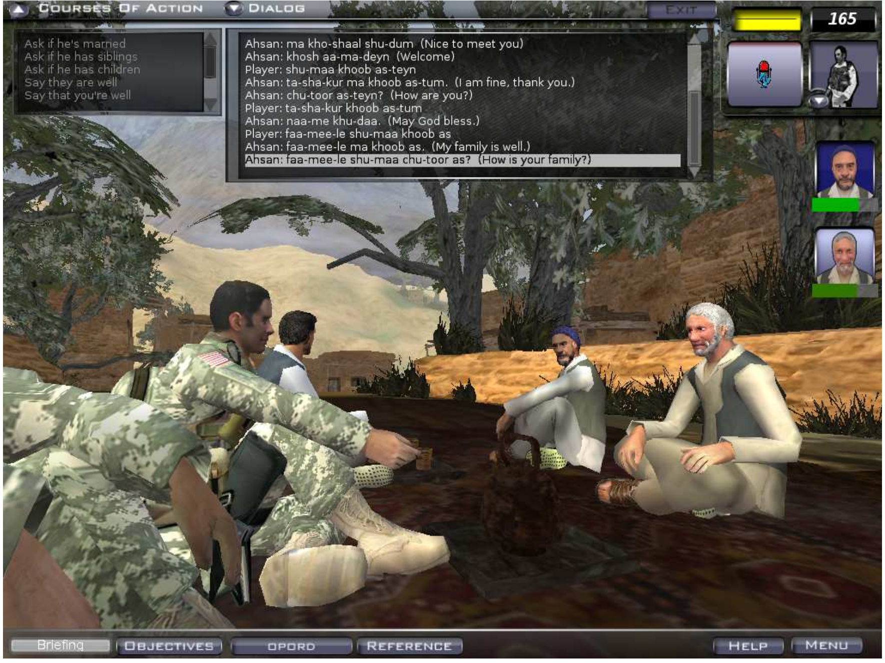
In the CultureCom project, we address these problems by developing a new system for creating training simulations in cross-cultural competency. Because these simulations encode a variety of linguistic, cultural, and task-level features, we refer to them as *social simulations*. Our system produces flexible, model-driven simulations that use both culture-general and culture-specific rules. As a result, we achieve the novel capability to swap cultural models, in the form of rule sets, in and out of a social simulation to reveal pedagogically relevant differences at the level of behavior (utterances, gestures) and intention (communicative act).

In this paper we evaluate the gains in efficiency that our new architecture provides. We encode multiple social simulations, using the CultureCom architecture and using a state-of-the-art architecture based on finite state automata. We show that the new model-driven architecture requires comparable authoring time for an initial simulation, but allows more objects to be reused, reducing authoring time and total number of objects created for each subsequent simulation.

2 CONVERSATIONAL AGENTS FOR CROSS-CULTURAL COMPETENCY TRAINING AND SIMULATION

Alelo produces language and culture training products on a range of devices, including desktop, web-based, and hand-held platforms. In these products, immersive serious games provide an integrated learning environment in which trainees must make decisions about mission goals and logistics and engage in cross-cultural and interpersonal interactions with socially intelligent virtual agents. Examples include Tactical Iraqi (Johnson & Valente, 2009) and the Virtual Cultural Awareness Trainers (VCATs), both of which were designed using Alelo's Situated Culture Methodology (Johnson & Friedland, 2010). A screen shot is given in Figure 1. In this example, the player controls an avatar (center left) in a simulated meeting with village elders. The player speaks in Dari, and the virtual elders respond to him with speech and gestures.

The architecture for simulating conversations in these systems is based on finite state automata (FSAs) that encode conversational branches at the level of communicative act. An example of such an FSA is shown in Figure 2. The sequence of communicative acts is strictly prescribed by the shape of the graphical FSA, whose objects can be manually re-authored and copied but not reused in new graphs.



Ahsan: ma kho-shaal shu-dum (Nice to meet you)
Ahsan: khosh aa-ma-deyn (Welcome)
Player: shu-maa khoob as-teyn
Ahsan: ta-sha-kur ma khoob as-tum. (I am fine, thank you.)
Ahsan: chu-toor as-teyn? (How are you?)
Player: ta-sha-kur khoob as-tum
Ahsan: naa-me khu-daa. (May God bless.)
Player: faa-mee-le shu-maa khoob as
Ahsan: faa-mee-le ma khoob as. (My family is well.)
Ahsan: faa-mee-le shu-maa chu-toor as? (How is your family?)

Figure 1. Meeting with the malek in Alelo's Operational Dari

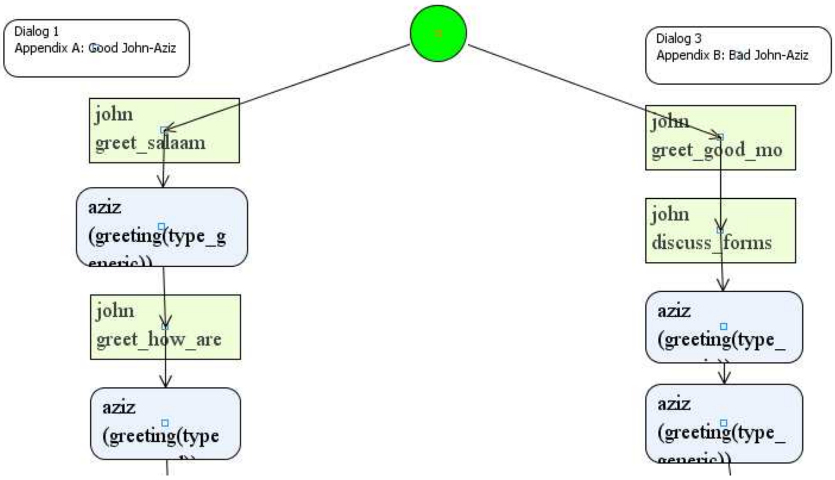


Figure 2. Sample of conversation captured as a finite state automaton

3 IMPROVED MODEL OF CONVERSATION

3.1 Modular Architecture

In contrast to the finite state system described above, CultureCom encodes agent behavior in a group of unsequenced, context-dependent rules, captured in a set of interoperable Protégé-frames ontologies (Gennari, et al. 2002) and executed using the CLIPS expert system (http://clipsrules.sourceforge.net/). This allows culture-general rules, such as "engage counterparts with respect" to be inherited and combined with culture-specific rules such as "in Afghan culture, questions about female family members is disrespectful." Crucially, the culture-general and culture-specific rule sets are stored in separate interoperable files, meaning that the agent's behavior can be adapted to a new culture by loading an American or Colombian culture model in place of the Afghan one. A sample of the inheritance hierarchy that makes this possible is shown in Figure 3.

The modular architecture also allows pieces of language that have already been authored for one simulation to be re-used in subsequent ones, rather than typed in again. This reduces the chance of misspellings and allows global management of the quality of the lexicon. As a result, linguistic behavior is consistent across all simulations that share the same language modules (for example, the "World English Language" file in Figure 3). Systems that require such knowledge to be duplicated or reauthored once per scenario run an increased risk of inconsistency. "Hello" in one simulation may become "Helo" in another. The risk increases as the number of scenarios grows; our architecture mitigates this risk.

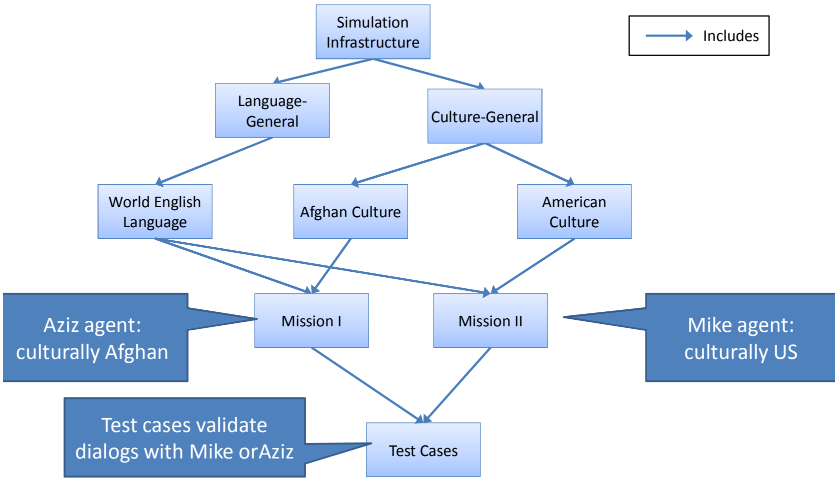

Simulation Infrastructure
Includes
Language-General
Culture-General
World English Language
Afghan Culture
American Culture
Aziz agent: culturally Afghan
Mission I
Mission II
Mike agent: culturally US
Test cases validate dialogs with Mike orAziz
Test Cases

Figure 3. Swap-in Swap-out model hierarchy

3.2 Commonsense Model of Microsociology

Figure 3 shows that files capturing culture-specific knowledge (“Afghan Culture” and “American Culture”) rely on data from culture-general files farther up the hierarchy (“Culture-General”). Together they comprise a logical commonsense model of culture that focuses on microsocial interaction (Hobbs & Sagae, 2011). The model is encoded as a set of predicates and axioms that express entities, properties, relations, events, and causal relations among events. It extends the framework described by Hobbs & Gordon (2010). The culture-general model applies to all CultureCom social simulations. An excerpt is given below:

> *A good reason for demonstrating a real friendship or establishing the pretense of a fictional one in this way is because friends are more likely to help each other out. So politeness is itself a way for people to increase the reliability of the Golden Rule.*

```
(forall (p1 p2 e)
   (if (and (polite p1 p2) (goal' e1 e p2) (believe p1 e1) (etc))
      (exist (e2) (and (help' e2 p1 p2 e) (cause e1 e2)))))            (1)
```

> *This axiom says politeness leads others to help one… many greeting conventions are motivated by exactly this rationalization -- we're friends, we care about each other's desires, and we help each other.*

This meta-rule is elaborated in more detail in the Afghan Culture model by an excerpt that explains culture-specific polite conversational openings (greetings):

…Greetings, defeasibly, are required to initiate an interaction:

```
(forall (e p1 p2)
   (if (and (interact' e p1 p2) (etc))
      (exist (e1)
         (and (greet' e1 p1 p2) (intBegins e1 e)))))                   (2)
```

> *The specifically Afghan form of the greeting has three exchanges. First there is the generic exchange "Salaam Alikum". Then each asks the other how they personally are. Then each asks the other about the well-being of their families. We can call the first turn the generic greeting, the second the personal greeting, and the third the family greeting. The generic greeting is defined simply as the specific utterance.*

```
(forall (e i u)
   (iff (genericGreet' e i u)
      (utter' e i u "Salaam Alikum")))
```

> *That is, a speaker i utters to a hearer u the string "Salaam Alikum".*

3.3 Data for Training Cross-Cultural Communication

The content of the logical model was developed in coordination with a data development and validation process (Wertheim & Agar, in press). This process was conducted by a team of Cultural and Linguistic Anthropologists, who interviewed subject matter experts from two cultures of focus: Dari-speaking urban Afghanistan, and Spanish-speaking Colombia. The interview material is annotated with ethnographic and sociolinguistic observations.

Based on this material, example *dialogs* are composed representing the performance target at which the final training system aims. A dialog is a script for the verbal communication that occurs in a social simulation. In keeping with the task-based nature of the training system as a whole, we developed dialogs with better (more successful) and worse (less successful) outcomes. A description of the developed dialogs is given in **Table 1**. Excerpts from dialogs 1a and 3a are shown in Table 2 and Table 3, respectively.

Table 1. Dialogs developed for evaluation. Language context indicates native language of interviewees. All dialogs are encoded in World English (W.E.).

ID	Player Name	Non-Player Name	Outcome	Culture Context	Language Context	Encoding	Length in Turns
1a	John	Aziz	Better	Afghanistan	Dari	W.E.	29
1b	John	Aziz	Worse	Afghanistan	Dari	W.E.	14
1c	John	Mike	Better	America	English	W.E.	10
2a	John	Aziz	Better	Afghanistan	Dari	W.E.	12
2b	John	Aziz	Worse	Afghanistan	Dari	W.E.	14
3a	John	Diego	Better	Colombia	Spanish	W.E.	31
3b	John	Diego	Worse	Colombia	Spanish	W.E.	31
4a	John	Diego	Better	Colombia	Spanish	W.E.	29
4b	John	Diego	Worse	Colombia	Spanish	W.E.	29

4 EXPERIMENTS

We conducted a series of experiments to evaluate the authoring efficiency gained by using the CultureCom system to instantiate these dialogs. In the CultureCom condition, objects are created using the Protégé ontology editor and saved into a file structure parallel to the one shown in Figure 3. The resulting files are ready to be used in the social simulation framework described by Sagae, et al. (2011). However in this evaluation we load the files into a text-based interaction loop where the author types conversational turns ("John" turns from dialogs 1-4) and views the system response, printed to the screen. These responses are produced in real time by the dialog engine, given currently-loaded models. To validate whether the models accurately capture one of the dialogs, the system can run each input turn sequentially against the current models and compare the predicted output

("Aziz" or "Diego" turns) to actual output (real-time system response).

We compare the CultureCom condition to a baseline condition where the same dialogs are authored using the FSA formalism. In this condition, dialog accuracy is tested using a tool similar to the CultureCom batch-load function. The FSA tester provides a pass/fail result, depending on whether actual output matched the predicted output exactly, or not.

4.1 Efficiency in the Number of Files and Build Process

Our first hypothesis was that instantiating a given scenario in a new culture is simpler, in terms of file changes and build process, for the CultureCom condition.

Table 2. Excerpt from dialog 1a: Better outcome in Afghanistan

Turn	Speaker	Line	Cultural Observations
1	John:	Salaam Alikum, Aziz.	Good: customary local greeting in local language.
2	Aziz:	Salaam Alikum, John.	Customary response.
3	John:	How are you today?	
4	Aziz:	I am well. And how are you?	
5	John:	I'm fine, things are going well. And how are things with your family?	Asking about family (in general, not women) before getting down to business.
		…	
9	John:	We have some forms that need to be filled out…	
		…	
24	Aziz:	I promise you that I will have the forms for you…	Fixed "promise" phrase is required to imply commitment; "ok" would not.

Table 3. Excerpt from dialog 3a: Better outcome in Colombia

Turn	Speaker	Line	Cultural Observations
1	John:	Buenas tardes, Diego.	Good: customary local greeting in local language.
2	Diego:	Buenas tardes, John. How are you today?	Customary response.
		…	
5	John:	I stopped by so we can set up a meeting …	
		…	
11	John:	Do you think it's too early for people… I suppose we could meet from 9 to 10.	Accommodates change in timing.
16	Diego:	I think it is best to give us the time.	Culturally appropriate indirectness. Doesn't come out and talk about lateness.

To test this hypothesis, we created the file hierarchy shown in Figure 3, and tested against dialogs 1b and 1c. In Dialog 1b, the Aziz character models Afghan cultural norms but John fails to observe them. John greets Aziz only once, failing to build rapport with a three-stage greeting. In dialog 1c, John has an American interlocutor and his directness results in a better, not worse, outcome. To accomplish dialog 1c given a working 1b, we create a new character, Mike, in the Mission II file. This character inherits existing data from the World English language model and new data from the American culture model. The test cases remain unchanged and the build process is unaffected. To accomplish the same behavior in the FSA case would require a new FSA, duplicates of the language objects, and duplicate communicative act objects. Since these objects are stored in a single file, total files changed is 1, but the change affects a large percentage of the file. In addition, the FSA formalism requires a build step to compile these objects into a runnable object, unlike in the CultureCom condition. As a result, such changes cannot be made on the fly to the FSA.

4.2 Efficiency in the Time Required to Author

Our second hypothesis was that, as the number of scenarios being authored grows, the CultureCom condition exhibits more efficiency than the baseline condition in terms of time required to author each scenario. To test, we encoded the same dialogs (3a, 3b, 4a, 4b) using both methods and compared authoring time for a number of steps, as well as overall. The results show that total time to author with CultureCom was greater for these four dialogs, however the trend in terms of scalability was favorable to the CultureCom condition. Time per dialog fell consistently in the CultureCom case from dialog 3a to dialog 4b, while in the baseline condition, time fell only when adapting a given dialog for a new outcome, as in 3a-3b or 4a-4b. There was no scalability in the baseline case when adapting from dialog 3 to dialog 4. An example of this trend is shown in **Table 4**, which shows authoring time for creating Communicative Act objects.

4.3 Efficiency in the Number of Authored Objects

Our third hypothesis was that, as the number of scenarios being authored grows, the CultureCom condition exhibits more efficiency than the baseline condition in terms of the number of objects that must be instantiated. To test, we used the same authoring task as for hypothesis 2, but evaluated on object counts rather than authoring time. The result shows that object reuse is greater in the CultureCom condition. In particular, language data, communicative act data, and higher-level behavior rules (the equivalent of transitions in the FSA) are all reused to greater advantage in the CultureCom condition. Object reuse in the baseline condition occurs, but as with authoring time, reuse is limited to dialogs that share the same language, culture, and topic (as in 3a-3b), but when adapting to a new topic (as in 3a-4a) there is much greater reuse in the CultureCom case. **Table 4** bears out this trend in the case of Communicative Act objects. In the CultureCom condition, the total number of Acts authored for dialog 3a is high, since we break each dialog turn into a greater number of Acts in this condition. Acts can cover a portion of a turn, and Acts which recur often (modeling “Okay”, “Bye”, or “No”) do not have to be reauthored. In the baseline case, monolithic Acts represent entire turns (“Okay, I’ll get the papers to you by Monday”). This yields fewer acts, but each of them occurs in a limited context and can rarely, if ever, be reused.

As a result, in the baseline case we see the number of Acts required for dialog 3a (31) is nearly the same as the number required for 4a (29). In the CultureCom case, we see a significant drop from 3a (77) to 4a (63)

Table 4. Time and object counts for Communicative Act authoring. Lowest time for each condition is shown in bold.

	CultureCom			Baseline		
Dialog	**Time**	**# Objects**	**Time/Obj**	**Time**	**# Objects**	**Time/Obj**
3a	00:38:04	77	00:00:30	00:12:07	31	00:00:23
3b	00:14:33	31	00:00:28	**00:08:06**	16	00:00:30
4a	00:28:53	63	00:00:28	00:14:21	29	00:00:30
4b	**00:13:52**	34	00:00:24	00:11:23	17	00:0031

5 CONCLUSIONS AND FUTURE WORK

The results presented here show that a compositional, model-based approach to social simulation development can result in greater efficiency, in terms of authoring time and reuse of linguistic and cultural resources that are expensive to develop. As the number of simulations increases, the advantage of authoring with reusable objects becomes more and more evident.

In addition to efficiency, another advantage is increased consistency. In the case of FSAs, there is no centralized data structure where cultural cues, norms, expectations, or rules can be saved. Two different authors working on FSAs for the same system must agree informally on these features, and there is no formal method for validating that a given FSA upholds the agreement. The CultureCom system identifies precisely which culture-general and culture-specific rules are in force for a given simulation, supporting consistency and formal validation.

In future work, we would like to investigate the accuracy of the CultureCom framework and the tradeoffs that exist between efficiency and word-level accuracy with respect to a given dialog. In the experiments described here, accuracy for the FSA condition was greater than for the CultureCom condition at the surface level, meaning that the FSA did a better job of replicating the dialog turns word-for-word. This effect is partly caused by the fact that CultureCom communicative acts (e.g., *greeting-response*) can be linked to multiple surface-level utterances (“I’m fine, thanks” “I’m doing well”). At evaluation time, the intent planning module of the dialog engine (Sagae et al., 2011) may select any of these utterances. The same features of the system that lead to greater object reuse and efficiency contribute to this perceived drop in accuracy, when we would like to optimize for both.

In addition, our current work focuses on dialogs, which encode verbal behavior. However the social simulation engine supports rules that capture non-verbal behavior as well. A natural extension of this work could apply the same data development, logical modeling, and social simulation architecture to model non-verbal behavior. Like verbal behavior, gestures are made and interpreted in culture-general and culture-dependent ways that make them well suited to an approach like ours.

ACKNOWLEDGMENTS

The authors would like to acknowledge the Office of Naval Research for their support of this work under contract NOOO14-09-C-0613.

REFERENCES

Fletcher, D. J. (1990). Computer-Based Instruction: Costs and Effectiveness. In A. Sage (Ed.), Concise Encyclopedia of Information Processing in Systems and Organizations. Elmsford, NY: Pergamon Press.

Gennari, John H., et al. (2002). “The Evolution of Protégé: An Environment for Knowledge-Based Systems Development.” in International Journal of Human-Computer Studies. 58:89-123.

Hobbs, Jerry R., and Gordon, A. (2010). "Goals in a Formal Theory of Commonsense Psychology", in A. Galton and R. Mizoguchi (eds.), Formal Ontology in Information Systems: Proceedings of the Sixth International Conference (FOIS 2010), IOS Press, Amsterdam, pp. 59-72.

Hobbs, J. R., & Sagae, A. (2011). “Toward a Commonsense Theory of Microsociology: Interpersonal Relationships” in *Proceedings of the 10th Symposium on Logical Formalizations of Commonsense Reasoning, AAAI Spring Symposium Series*. March 21-23, 2011. Stanford, California.

Johnson, W. L., & Friedland, L. (2010). Integrating Cross-Cultural Decision Making Skills into Military Training. In D. Schmorrow & D. Nicholson (Eds.), Advances in Cross-Cultural Decision Making. London: Taylor & Francis.

Johnson, W.L. & Valente, A. (2009), “Tactical Language and Culture Training Systems: Using AI to teach foreign languages and culture.” AI Magazine, 30 (2), 72-83.

McDonald, D. P., McGuire, G., Johnston, J., Selmeski, B., & Abbe (2008), “Developing and Managing Cross-Cultural Competence Within the Department Of Defense: Recommendations For Learning and Assessment.” US Defense Regional and Cultural Capabilities Assessment Working Group.

Sagae, A., Johnson, W. L., & Valente, A. (2011). Conversational Agents in Language and Culture Training. In D. Perez-Marin & I. Pascual-Nieto (Eds.), Conversational Agents and Natural Language Interaction: Techniques and Effective Practices (pp. 358-377). Madrid: IGI Global.

Wertheim, S. and Agar, M. (in press) “Culture that Works” in *Proceedings of the Second Conference on Cross-Cultural Decision Making*. July 23-25, 2012. San Francisco, CA.