Spectrum Curricula for Measuring Teachability

Jacob Beal
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
jakebeal@bbn.com

Alice Leung
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
aleung@bbn.com

Robert Laddaga
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
rladdaga@bbn.com

ABSTRACT

Machines learning from human teachers can advance both the capabilities of engineered systems and our understanding of human intelligence. A key challenge for this field, however, is how to effectively measure and compare the teachability of machine learners, particularly given the diversity of potential learners and the inherent adaptivity and variability of human instructors. We are addressing this challenge with *spectrum curricula*, where each spectrum curriculum is a suite of lessons, all with the same instructional goal, but varied incrementally with respect to some property of interest. We have designed a set of seven spectrum curricula, investigating three instructional properties in the RoboCup domain, and are implementing these within the Bootstrapped Learning Framework produced by the DARPA Bootstrapped Learning program[2]. The materials we are producing are being made publicly available on the Open Bootstrapped Learning Project website, such that any researcher can test against the curricula available or can contribute their own curricula to improve the quality of this community resource.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning

General Terms

Measurement

Keywords

Teachability, Bootstrapped Learning, Robocup, Spectrum Curriculum

1. INTRODUCTION

Humans are much better at learning from other humans than our current machine learning systems are at learning from humans. Right now, a human who wants to instruct a machine needs to be two types of expert at once—an expert in the subject that is to be instructed, and also an expert in some method of configuring the machine, such as programming or knowledge representation.

Cite as: Spectrum Curricula for Measuring Teachability, Jacob Beal, Alice Leung and Robert Laddaga, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. XXX-XXX.

In contrast, even the most naive instructors can easily teach other humans, as when little kids teach one another games. Outside of formal classroom settings, humans teach one another informally, imprecisely, and highly effectively (and even a classroom is extremely informal and imprecise compared to the current requirements for machine instruction). Cultural knowledge, like how to change a flat tire, what to do when you're sick, or fun games to play together, passes easily from person to person in a population, even when the teacher has no training as an instructor.

Building systems capable of such human-like learning is important for both engineering and scientific reasons. From an engineering perspective, if humans and machines are to work together in challenging environments, then machines must be able to learn from humans in a manner similar to how humans learn from one another. The environments that humans act in are highly diverse and constantly changing—whether in the office, at home, or on a battlefield. Even a machine with an exhaustive background knowledge must be customized for the particular environment in which it acts, just as a secretary must become accustomed to the peculiarities of a particular office's policies and customs or a soldier must learn the particular patterns and cautions of a deployment area.

For humans and machines to work together routinely, machines must have this human-like capacity to learn from naive instructors. Neither the homemaker nor the soldier have the time and inclination to bother with a machine partner otherwise. Although machines are still far from human-like teachability, much might be accomplished even by meeting the humans partway, as with Graffiti character recognition[5]. Even a partial win could be revolutionary, allowing machines to adapt "culturally" to the environments in which they are deployed and also allowing the vast reservoirs of human curricula to be used in configuring machines for application domains.

From a scientific perspective, building a machine that learns more like a human does may shed light on the nature of human learning. We may hope to investigate questions such as:

- How important are shared assumptions and biases for human-like learning?
- Are humans powerful but "rational" learners, or do they leap to unwarranted (but often correct) conclusions?
- How does instruction enable representational change?
- What are the relative contributions of shared architec-

ture, shared inherent biases, shared culture, and shared experiences?

- Is our arsenal of machine learning techniques missing any fundamental tools that humans are employing?

A key challenge for this field, however, is how to effectively measure and compare the effectiveness of machine learners, particularly given the diversity of learners and the inherent adaptivity and variability of human instructors. *Spectrum curricula* address this challenge by measuring a student's ability to adapt to a set of fixed teachers, which all teach the same lesson but vary incrementally along some property of interest in how the lesson is taught.

We have previously proposed the notion of spectrum curricula in [1]; in this paper, we develop the idea further and give specific and detailed examples of how such curricula can be designed. In particular, we have designed a set of seven spectrum curricula, investigating three properties of instruction—strength of assumptions about mutual knowledge, distance of transfer between lesson and use in context, and level of detail—in the domain of 3-on-2 keepaway in RoboCup simulated soccer[3]. We are implementing these curricula within the Bootstrapped Learning Framework produced by the DARPA Bootstrapped Learning program[2], and making the materials we are producing publicly available on the Open Bootstrapped Learning Project website, `http://dsl.bbn.com/BL/`, such that any researcher can test against the curricula available or can contribute their own curricula to improve the quality of this community resource.

2. SPECTRUM CURRICULA

When a skilled human teacher instructs a single student, the teacher typically adapts their curriculum to the background and needs of the student. Consider, for example, teaching a person how to play Kriegspiel, a chess variant in which each player sees only their own pieces and a referee adjudicates. If the student is already deeply familiar with chess, the game may be explained just so simply, with many of the details in making such a variant work left implicit, whereas all the rules may need to be spelled out explicitly for a student unfamiliar with chess. Likewise, some students may pick things up rapidly, whereas others may need much reinforcement before they are comfortable with the game. In a tutoring environment, a teacher will adapt to convey the lesson to the student, expanding where prompted by student difficulties and spending less time where the student picks up readily.

While this sort of adaptability is good for making learning systems work, since a human teacher can adjust to match the feedback being given by the system, it presents a problem for measuring the teachability of a system. An adaptive teacher can teach many systems with different capabilities the same lesson, masking their differences. Moreover, no human teacher will instruct exactly the same way twice and different teachers will instruct the same student in different ways.

A *spectrum curriculum* addresses this problem by turning it around, and measuring the adaptability of the student to a set of teachers, each executing a relatively rigid curriculum. Each spectrum curriculum is built around a single variable property of instruction, such as the level of detail in a lesson, and contains a set of lessons that incrementally vary this property from "too hard" to "too easy." For example, a level of detail spectrum for moving pawns might range from a single *en passant* capture on the "too hard" end of the spectrum to hundreds of examples with different rest-of-board configurations on the "too easy" end of the spectrum. The series of lessons is taught in order from difficult to easy, and the student is tested before the first lesson and after each lesson. The product of training against such a curriculum is thus a curve showing how the student's cumulative understanding improves as the aspect varies.

Figure 1: A game of 3-on-2 keepaway being played in the Robocup simulator.

Testing an instruction-based learning system against several curricula for the same aspect should produce a good measure of the system's adaptability with respect to that aspect, with more area under the curve generally being better. No system should be expected to perform well on the most challenging lessons, but the performance of well-designed system should degrade gracefully as the challenge increases. Moreover, a spectrum that ranges from "too hard" to "too easy" should also be able to provide a fine-grained measure of performance on which incremental progress towards human-like teachability can be easily tracked.

Note, however, that the systems with the most human-like teachability may not show the most breadth on a given spectrum! For example, an assumption about human-like teacher's intent may cause a system to fail on an "easy" task where too much detail is provided, since the student cannot believe something so painstakingly instructed can be so simple as the concept it has deduced.

To make a spectrum curriculum, one must first specify the spectrum property of interest. The might be any topic of interest to a researcher: the length of the lesson in seconds, the number of examples provided, the number of instructional techniques employed, the level of math knowledge that the instruction depends on, and many others. Spectrum curricula therefore can vary enormously, and the question then is which properties are the most interesting, the most useful and the most predictive of the behavior of human or machine learners. We are not able to provide an extensive characterization in this paper, but the topic itself is of scientific and philosophical interest, and we have barely scratched the surface in terms of the range of spectrum curricula we have begun to study. One of the things we would like this paper to accomplish is the beginning of a discussion of spectrum curricula as a topic of scientific interest and study in the area of educational technology.

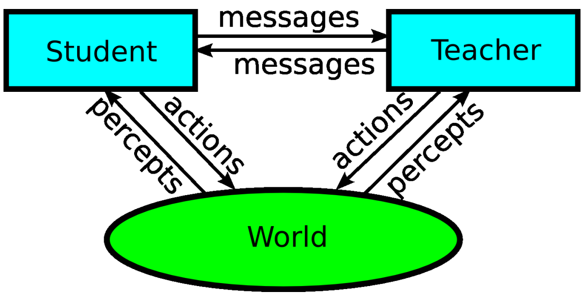


Figure 2: In an instruction scenario, three agents—teacher, student, and world—interact by sending symbolic messages. The world sends percept messages to both teacher and student, and either of them can manipulate it by sending it action messages.

3. BBN ROBOCUP KEEPAWAY CURRICULA

We have begun applying the notion of spectrum curricula by designing a set of spectrum curricula, using the domain of Robocup soccer simulations[3]. We have designed seven curricula, each of which teaches a different skill for playing a 3-on-2 keepaway game in a restricted portion of the soccer field.

Our curricula use an agent-based instructional framework. An instruction scenario is set up in terms of a set of interacting agents: a teacher, a student, and a world (Figure 2). These all evolve independently, communicating by passing symbolic messages to one another. The teacher and student jointly observe the world, so any time the world's state changes, it sends updated percepts to both. The teacher and the student can both affect the world's state by commanding it to take actions. Finally, the teacher and the student send one another a variety of messages for different stages of the teaching process. This general framework should be able to support "classroom" worlds with multiple students and/or multiple teachers, as well as dialogue between teacher and student. At present, however, the curricula are all designed for one-on-one instruction with low interactivity.

Each of the seven spectrum curricula we have designed is a collection of 6 to 10 lessons, incrementally varying from "hard" to "easy" along one of three properties of interest:

- Strength of assumptions about mutual knowledge (2)
- Distance of transfer between lesson and use in context (3)
- Detail of instruction (2)

Three lessons teach knowledge for the "keeper" team, three teach knowledge for the "taker" team, and one is used by both teams. In every case, the piece of knowledge being learned is a function that either returns a boolean result or chooses one of two options.

These seven curricula also exercise three of the "natural instruction method"[4] teaching modalities that have been identified in the DARPA Bootstrapped Learning project:

- learning from examples (6)
- learning from "telling" (2)
- learning from feedback (from the instructor or environment) (2)

To clearly illustrate the spectrum curricula concept and show how it can be implemented, we present all seven curricula, organizing them by property to show how different curricula can investigate the same property of interest.

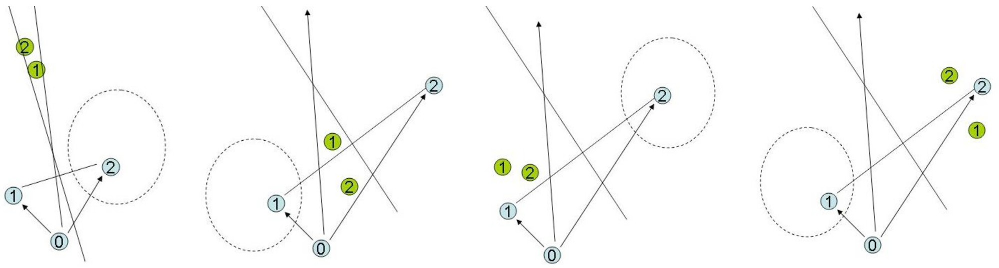
3.1 Assumptions about mutual knowledge

Out of Bounds.
Both keeper and taker players try to avoid going out of bounds. This spectrum teaches a function that is used to tell when a nearby location is illegal.
Spectrum Property: Strength of assumptions about mutual knowledge.
Natural Instruction Methods: Learning by Example
Test: 10 random locations
Lessons (Easy to Difficult):

- 50 labelled examples scattered randomly around the field
- 20 labelled examples scattered around the boundary (assumes values not near the boundary are indicated by the examples given)
- 8 labelled examples, one in and one out on each line (assumes a rectilinear area)
- 2 labelled examples, in or out, at opposite corners (assumes the area is aligned with cardinal directions)
- hint at the line, two labelled examples, one in and one out (assumes the line is the border)
- hint at the line, one labelled example in or out (assumes the other side of the line is the opposite value)
- only hint at the line (assumes interior of boundary is likely to be "good")

Taker: Triangle Bounds.
In order to be effective at intercepting, a taker always tries to stay "near enough" to the center of the group. This spectrum teaches that "near enough" means within the triangle formed by the three keepers. Assumes prior knowledge of a "Distance" function that takes two positions and returns a scalar and an "Angle" function that takes a vertex and two points on rays extending from it and reports the angle between the rays.
Spectrum Property: Strength of assumptions about mutual knowledge.
Natural Instruction Methods: Learning from Examples
Test: 10 random configurations of keepers and the taker
Lessons (Easy to Difficult):

- 50 labelled examples scattered randomly around the field, with the three keepers in canonical positions. In every case, the positions of the three keepers are hinted at, along with the "distance" and "angle" functions.
- 20 labelled examples scattered around the boundary (assumes values not near the boundary are indicated by the examples given).
- Three pairs of examples, one in and on out on each edge (assumes boundaries are straight lines).



(a) Case 1 (b) Case 2 (c) Case 3 (d) Case 4

Figure 3: Configuration cases for teaching a keeper player where to pass. Keepers are blue, takers are green, and the player is keeper 0.

- One pair of examples, in and out, next to a boundary (assumes symmetry in decision boundary).
- One pair of examples, in and out, not near a boundary (assumes the function is set by the in/out spatial relation).
- One example, in or out, not near a boundary (assumes the other side is the opposite value).

3.2 Transfer between lesson and use

Keeper: Where to Pass.

When a keeper has decided to pass, it needs to choose one of its teammates to pass to. This spectrum teaches the keeper to pass to the the team-mate with the greatest minimum angle to a taker. Assumes prior knowledge of “Distance” and “Angle” functions.

Spectrum Property: Distance of transfer between lesson and use in context.

Natural Instruction Methods: Learning from Examples

Test: 10 random configurations of all five players

These lessons involve four types of configurations, as shown in Figure 3. We will refer to the three keepers are K0, K1, K2, and the two takes as T1 and T2. The player is always K0 and has the ball, and is choosing whether to pass to K1 or to K2. The configurations are divided into types based on what types of mistakes a human-like learner might make, given examples of that type.

Figure 3(a) shows an example of a Case 1 configuration. With examples of this type of configuration, a learner might mistakenly come to believe that the answer is to the pick the furthest teammate or the teammate with the least average distance from the two takers. In Case 1, the correct answer, K2, is the teammate furthest from K0. T1 and T2 are closer in distance to K2 but closer in angle to K1.

Figure 3(b) shows an example of a Case 2 configuration. With examples of this type of configuration, a learner might mistakenly come to believe that the answer is to the pick the closest teammate or the teammate with the least average distance from the two takers. In Case 2, the correct answer, K1, is the teammate closest from K0. T1 and T2 are closer in distance to K1 but closer in angle to K2.

Figure 3(c) shows an example of a Case 3 configuration. With examples of this type of configuration, a learner might mistakenly come to believe that the answer is to the pick the furthest teammate or the teammate with the greatest average distance from the two takers. In Case 3, the correct answer, K2, is the teammate furthest from K0. Also, T1 and T2 are closer in both distance and angle to K1 than they are to K2.

Figure 3(d) shows an example of a Case 4 configuration. With examples of this type of configuration, a learner might mistakenly come to believe that the answer is to the pick the closest teammate or the teammate with the greatest average distance from the two takers. In Case 4, the correct answer, K1, is the teammate closest to K0. Also, T1 and T2 are closer in both distance and angle to K2 than they are to K1.

For each case, it is also possible to generate a Case nA version. In the Case nA version, T1 and T2 are in the same place, simplifying learning.

Lessons (Easy to Difficult):

- 36 labelled examples with 3 keepers in different positions and the 2 takers at the same location: 12 examples each of Case 1A, Case 2A, and Case 3A.
- 36 labelled examples of all 5 players in different positions: 9 examples each of Case 1, Case 2, Case 3, and Case 4.
- 12 labelled examples with 3 keepers in different positions and the 2 takers at the same location: 2 examples each of Case 1A, Case 2A, and Case 3A and 6 examples of Case 4A.
- 2 labelled examples of all 5 players in different positions. 1 example each of Case 2 and Case 3.
- 1 labelled examples with 3 keepers in different positions and the 2 takers at the same location. The example is Case 1A.
- 1 labelled example of all 5 players in different positions. The example is Case 1.

Taker: Who to Guard?.

Assuming that a taker player has decided to guard one of the keepers that does not have the ball, it needs to pick one of the two to be the one that it guards. This spectrum teaches that a taker should guard the keeper that it is closest to. Assumes prior knowledge of the “Distance” function.

Some potential incorrect conclusions that the learner might make are it should select the keeper who does not have the ball and is (1) closest to the ball (2) furthest from the ball (3) nearest to the other taker (4) farthest from the other taker.

Spectrum Property: Distance of transfer between lesson

and use in context.
Natural Instruction Methods: Learning from Examples
Test: 10 random configurations of you, the other taker, a keeper with the ball, and two keepers who don't have the ball.

In all the following lesson examples, the taker player is T1 and the keeper with the ball is K0.

Lessons (Easy to Difficult):

- 20 examples with only T1, K1, and K2. No ball. Make sure T1 is not equidistant from K1 and K2.
- 20 examples with all 5 players and T2 at the same location as K0. Make sure T1 is not equidistant from K1 and K2.
- 4 examples with K0 in the center of the field. K1 and K2 are equidistant from K0. T2 is at the same location as K0. T1 is located approximately on the line that runs through K1 and K2. In one example, T1 is on the side of K1 that is away from K2. In one example, T1 is between K1 and K2, but closer to K1. In one example, T1 is between K1 and K2, but closer to K2. In one example, T1 is on the side of K2 that is away from K1.
- 4 examples with K1 in the center of the field. K0 and K2 are equidistant from K1. T2 is at the same location as K0. T1 is located approximately on the line between K1 and K2. In one example, T1 is approximately 20% of the distance between K1 and K2. In the other examples, T2 is approximately 40%, 60%, and 80% of the distance between K1 and K2.
- One example with T1, K1, and K2 randomly placed on the field. No ball, no K0, no T2. Make sure that T1 is not equidistant from K1 and K2.
- One example with K0 in the center of the field. K1 and K2 are equidistant from K0. T2 is in the same places as K0. T1 is near K1. The answer is K1.

Keeper: Where to Move.

When a keeper does not have the ball, it tries to get open so that its team-mate can pass to it. Thus spectrum teaches the keeper that good "open" locations are in a band 50-80% of the field away from the keeper with the ball, and not within 10 degrees of either taker or the other keeper without the ball. Assumes prior knowledge of "Distance" and "Angle" functions.
Spectrum Property: Distance of transfer between lesson and use in context.
Natural Instruction Methods: Learning from Feedback and Learning by Examples.
Test: 10 random configurations with the following properties: two are acceptable locations, and two each exercise the distance prohibition and the three player proximity prohibitions.

In all the following examples, the keeper player is K0.

Lessons (Easy to Difficult):

- 100 labelled examples of all 5 players in random plausible positions. The ball is with a different keeper. Student is then allowed to propose up to 10 position/label pairs and told whether each is right or wrong.
- 40 labelled examples with three keepers and one taker in random plausible positions. The ball position and feedback stage are as before.
- 20 labelled examples with two keepers in fixed positions, K0 at a random distance and one taker in a random position between them. The ball position and feedback stage are as before.
- 10 labelled examples with the ball keeper in a fixed position, K0 at a random distance, and one taker in a random position between them. The ball position and feedback stage are as before.
- Four examples, one per rule. One has just two keepers, and K0 starts too close and moves away until its distance is acceptable and the ball is kicked to it. Another is the same except that the keeper starts too far away and moves closer. The other two have a taker at too near an angle and a keeper at too near an angle. Feedback is as before.
- Two examples: two keepers and a taker slightly off the line between the keepers. In the first example, K0 is at an acceptable distance, and moves until its angle is acceptable, at which point the ball is kicked to it. In the second example, the angle is acceptable, but the distance is too close. Feedback is as before.
- One example: two keepers and a taker slightly off the line between the keepers. K0 starts too close in distance and angle, then moves away from the line, simultaneously hitting an acceptable distance and angle, at which point the other keeper kicks the ball to it. Feedback is as before.
- Same as before, but the ball starts with K0.

3.3 Detail of instruction

Taker: Guard or Take?.

When a keeper is holding the ball, each taker player periodically checks to see if it should be the one trying to take the ball from that keeper. If not, it will guard one of the other keepers. This spectrum teaches that the taker should try to take if it is the one closest to the ball. Assumes prior knowledge of a "Distance" function.
Spectrum Property: Detail of instruction
Natural Instruction Methods: Learning from Examples, becoming Learning by Telling as lessons become easier.
Test: 10 random configurations of takers and ball-holding keeper

Lessons (Easy to Difficult):

- Full calculation told as hints: `GoTake = Distance(me.pos,ball.pos)` `<Distance(other.pos,ball.pos)` Given an example of two takers and a ball on the field, with the function returning opposite answers when applied to the two takers.
- Given both distance calls as hints, but not comparison.
- Hints just point to distance from teammate to ball.
- Hints just point to distance function and ball.
- Hint just points to distance function.

- Hint just points to position of ball.
- No hints at all, just the example.

Keeper: When to Pass.

Whenever a keeper player has the ball, it periodically checks to see whether it should continue holding the ball or to pass. This spectrum teaches that it should pass whenever a taker comes "near," and how to determine what is "near." Assumes prior knowledge of a "Distance" function.
Spectrum Property: Detail of instruction
Natural Instruction Methods: Learning by Feedback, becoming Learning by Telling as lessons become easier.
Test: 10 random configurations of takers
Lessons (Easy to Difficult):

- Full calculation told as hints:
  `PassNow = Or(Near(taker0),Near(taker1))`
  `Near(taker) = Distance(me.pos,taker.pos)<k`
  The student is then told to maximize their length of possession and given 10 feedback trials.
- Don't tell the value of the threshold k.
- Also don't tell how to use the distance calculation.
- For `Near`, hint only that the positions of `self` and `taker` are important.
- For `Near`, hint only that `self` is important.
- Give no information about `Near`.
- Hint only that `Near` should be calculated for `taker0` and `taker1`, but not how to combine them.
- Hint only that `taker0` and `taker1` are important.
- Only tell the student to maximize possession time.
- Only tell the student that possession time is important.

4. IMPLEMENTATION AND DISTRIBUTION

We are implementing these curricula in the Bootstrapped Learning Framework that is being produced by the DARPA Bootstrapped Learning program[2]. This Java-based framework provides an infrastructure for agent-based instruction scenarios, a scripting language for designing lessons and connecting them to form curricula, and a knowledge representation language (InterLingua[4]) that can be used to describe the goal knowledge that a lesson is attempting to teach. A Robocup domain package created by SRI interfaces a standard Robocup simulator into this framework and provides a base player for each team, with spots in its strategy where learned knowledge may be plugged in.

We are making the curricula we produce publicly available on the Open Bootstrapped Learning Project website, `http://dsl.bbn.com/BL/`, along with a copy of the framework, additional documentation, and a semi-competent base learner for researchers to build off. Researchers can use the spectrum curricula on the site as challenge problems or to evaluate their systems. Researchers are also encouraged to submit new spectrum curricula to the collection, either using the provided framework or their own framework, to improve the quality of this community resource.

Figure 4: BBN Open Bootstrapped Learning Project webpage

5. CONTRIBUTIONS

We have demonstrated how spectrum curricula can be designed for investigating a student's teachability, where each spectrum curriculum incrementally varies how a lesson is taught with respect to a property of interest. More particularly, we have presented a set of seven spectrum curricula in the Robocup domain that can be used to investigate three properties of instruction: strength of mutual knowledge assumptions, transfer distance between lesson and use, and lesson detail.

It is important to remember, however, that the actual effectiveness of these curricula at capturing their designed goals has yet to be evaluated. An important future direction of research is to test both humans and machine learners against these curricula in order to verify that useful spectra are actually produced.

The materials we are producing are being made publicly available on the Open Bootstrapped Learning Project website, such that any researcher can test against the curricula available or can contribute their own curricula. As this public resource is improved, we hope that it will prove stimulating to the community of researchers investigating teachability, both providing a useful set of challenge problems and a meaningful standard of cross-system comparison, and thereby advance us toward a future of adaptive human-machine collaboration and predictive models of human teachability.

6. REFERENCES

[1] J. Beal, P. Robertson, and R. Laddaga. Curricula and metrics to investigate human-like learning. In *AAAI 2009 Spring Symposium "Agents that Learn from Human Teachers"*, March 2009.

[2] DARPA IPTO. Bootstrapped learning. http://www.darpa.mil/ipto/programs/bl/bl.asp, (Retrieved Nov. 5, 2008).

[3] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara. Robocup: A challenge problem for ai and robotics. In *RoboCup-97: Robot Soccer World Cup I*, pages 1–19, London, UK, 1998. Springer-Verlag.

[4] L. Lefkowitz, J. Curtis, and M. Witbrock. Accessible research cyc. Technical Report AFRL-IF-RS-TR-2007-204, Air Force Research Laboratory, 2007.

[5] I. S. MacKenzie and S. X. Zhang. The immediate usability of graffiti. In *Graphics Interface '97*, pages 129–137, 1997.