Final Report for AOARD Grant FA2386-10-1-4125

"A Unified Architectural Approach to the Hybrid Mixed Challenge of Situation Assessment and Prediction (Task 1: Representing and Processing 3D Imagery)"

October 17, 2013

Name of Principal Investigators (PI and Co-PIs): Paul S. Rosenbloom

- e-mail address: rosenbloom@usc.edu
- Institution: University of Southern California
- Mailing Address: USC Institute for Creative Technologies, 12015 Waterfront Dr., Playa Vista, CA 90094-2536
- Phone: (310) 448-5341
- Fax: (310) 301-5009

Period of Performance: 8/20/2010 – 8/19/2013

Abstract

The scientific goal of the full proposal focused on the development of a new cognitive architecture – which has since been named *Sigma* (Σ) – that is based on *graphical models*, with a specific emphasis on the *hybrid* (combining continuous signal processing and discrete symbol processing) *mixed* (combining probabilistic representations of uncertainty with symbolic representations of knowledge) challenge of supporting robust situation assessment and prediction (SAP). Task 1, which was the one funded, specifically concerned the *representation and processing of mental imagery* in Sigma. The multi-year objectives of this task were to: (1) develop a means of representing mental imagery that leverages Sigma's unique capabilities and that is closely integrated with it (and that extends it to include (mixtures of) Gaussians for noisy continuous images); (2) implement mental imagery transformations – such as translation, scaling and rotation – within Sigma; and (3) produce predictions based on mental imagery, both in isolation and in conjunction with input about external reality.

Except for the extension to Gaussians, these objectives were achieved, with 1D, 2D and 3D mental imagery grounded directly in the multidimensional piecewise-linear functions that are at the core of Sigma, and the standard imagery transformations modifying the locations of the boundaries between the regions of these functions. This combination surprisingly turned out to be general enough to support significant forms of processing that weren't originally conceived of as imagery, but which used numeric (metric) dimensions – such as initializing, and returning results from, subgoals and processing rewards and value functions in reinforcement learning – with the transformations turning out to directly yield a primitive form of mental arithmetic on these dimensions. In conjunction with Sigma's graphical models, we were also able to go beyond the simple image transformations that had been proposed to incremental image composition and extraction of critical spatial properties from these composites. We were furthermore able to demonstrate combinations of prediction and perception in localization tasks – for example, in simultaneous localization and mapping (SLAM) – and to go beyond what was originally proposed in demonstrating learning to predict in the context of mental imagery.

Introduction

The development of Sigma is being driven by three general desiderata: *grand unification* (uniting the requisite cognitive and non-cognitive aspects of embodied intelligent behavior); *functional elegance* (exhibiting a broad set of capabilities while remaining fundamentally simple and theoretically elegant); and *sufficient efficiency* (behaving rapidly enough for anticipated applications). The ultimate goal is an architecture that leverages a small but general set of mechanisms – effectively defining a form of *cognitive Newton's laws* – to span from perception through cognition to action.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **06 NOV 2013**	2. REPORT TYPE **Final**	3. DATES COVERED **20-08-2010 to 19-08-2013**
4. TITLE AND SUBTITLE **A Unified Architectural Approach to the Hybrid Mixed Challenge of Situational Assessment and Prediction**		5a. CONTRACT NUMBER **FA23861014125**
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Paul Rosenbloom**		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Southern California,12015 Waterfront Drive,Playa Vista, CA 90094-2536,United States,CA, 90094-253**		8. PERFORMING ORGANIZATION REPORT NUMBER **N/A**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **AOARD, UNIT 45002, APO, AP, 96338-5002**		10. SPONSOR/MONITOR’S ACRONYM(S) **AOARD**
		11. SPONSOR/MONITOR’S REPORT NUMBER(S) **AOARD-104125**
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES		

14. ABSTRACT

The scientific goal of the full proposal focused on the development of a new cognitive architecture ? which has since been named Sigma (Σ) ? that is based on graphical models, with a specific emphasis on the hybrid (combining continuous signal processing and discrete symbol processing) mixed (combining probabilistic representations of uncertainty with symbolic representations of knowledge) challenge of supporting robust situation assessment and prediction (SAP). Task 1, which was the one funded, specifically concerned the representation and processing of mental imagery in Sigma. The multi-year objectives of this task were to: (1) develop a means of representing mental imagery that leverages Sigma?s unique capabilities and that is closely integrated with it (and that extends it to include (mixtures of) Gaussians for noisy continuous images); (2) implement mental imagery transformations ? such as translation, scaling and rotation ? within Sigma; and (3) produce predictions based on mental imagery, both in isolation and in conjunction with input about external reality. Except for the extension to Gaussians, these objectives were achieved, with 1D, 2D and 3D mental imagery grounded directly in the multidimensional piecewise-linear functions that are at the core of Sigma, and the standard imagery transformations modifying the locations of the boundaries between the regions of these functions. This combination surprisingly turned out to be general enough to support significant forms of processing that weren?t originally conceived of as imagery, but which used numeric (metric) dimensions ? such as initializing, and returning results from, subgoals and processing rewards and value functions in reinforcement learning ? with the transformations turning out to directly yield a primitive form of mental arithmetic on these dimensions. In conjunction with Sigma?s graphical models, we were also able to go beyond the simple image transformations that had been proposed to incremental image composition and extraction of critical spatial properties from these composites. We were furthermore able to demonstrate combinations of prediction and perception in localization tasks ? for example, in simultaneous localization and mapping (SLAM) ? and to go beyond what was originally proposed in demonstrating learning to predict in the context of mental imagery.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

Such an architecture should be a major step forward in developing intelligent agents/robots and virtual humans. It should also yield a new, more integrated and hopefully more effective, approach to complex but more specialized activities such as situation assessment and prediction.

The work funded by this grant ended up reflecting all three of the above desiderata. The intent was to explore grand unification by understanding how to support mental imagery that bridges perception and cognition. Functional elegance was key in determining that mental imagery should be approached via the same core representation – multidimensional piecewise-linear functions – and reasoning algorithm (a message passing approach based on the *summary product algorithm* over *factor graphs*) used for all other processing in Sigma. (Introductions to Sigma and to its use of both piecewise-linear functions and summary product over factor graphs can be found in the attached publications.) Sufficient efficiency came in to the picture with the development and implementation of a new sparse(r) representation for piecewise-linear functions that shows potential for yielding significant speedups in mental imagery tasks. Success in achieving the main objectives of this task brings us closer to systems that can effectively exploit high-level cognition in complex spatial environments.

Results and Discussion

Most of the results produced over the three years of this grant are described in the attached publications. These include representation of 1-3D continuous (and discrete) imagery buffers as piecewise-linear functions; implementation of affine transformations that enable translation, scaling, reflection and rotation (by multiples of 90°); synthesizing multiple images into new composite images, along with adding and deleting specific sub-objects; extracting spatial properties from these composites, such as edges, overlaps and relative directions; leveraging mental imagery in both problem solving and learning (papers on these topics received *Kurzweil Awards* at the annual Artificial General Intelligence (AGI) conference in 2011 and 2012); and the use of mental imagery in both classical cognitive tasks – such as the Eight Puzzle – and in (simulated) robotics tasks that involve perception, localization, mapping and action selection. Because this work is well documented in the attached publications, it won't be described further here. Instead, following a brief discussion of integrating Gaussians into Sigma, a description will be provided of recent, and still very preliminary, unpublished work on the new sparse(r) representation for piecewise-linear functions, before popping back up to explore possible follow-on/future work.

As mentioned in the introduction, Sigma uses a message-passing scheme – based on the summary product algorithm – to structure computations on factor graphs. The messages, as well as the factor functions themselves – except in specially optimized cases, such as are used for affine transforms – are instantiated as piecewise-linear functions. Considerable thought has gone into how to incorporate (mixtures of) Gaussian's into this existing function representation, and into whether other representations for continuous functions, such as particle filters, would be even better. Sigma can already represent continuous functions as closely as desired by approximating them in a piecewise linear manner, but at the potential cost of many regions. The key questions here were whether Gaussians (or other possibilities) would yield a more compact, and thus more efficiently processed, representation for noisy images, and how such a capability could be integrated with the existing representation. Although some conceptual progress has been made on this problem, it did not yield concrete results during the period of this grant.

Progress has instead been made on a sparse(r) representation for piecewise-linear functions that was not originally proposed, but whose importance became obvious as the work on mental imagery was pursued. Because message passing is the main computational workhorse in Sigma, the data structures used to represent these functions are critical. For some time, such functions have been represented as multidimensional arrays of orthotopic regions, each of which is doubly linked along each of its dimensions (Figure 1). This representation allows slicing of functions at arbitrary points – to define regions that extend across a large area of repeated values, saving space and computation time – but requires slices that span the entire dimension. This yields an array of regions at the expense of more partitioning of regions than would strictly be necessary just to represent the function via regions. This may also, and more critically, lead to a large number of regions with the same

value (usually zero), as in Figure 1, which has four zero-valued regions.

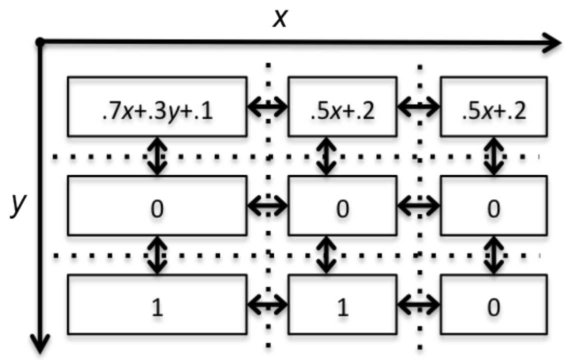

Figure 1: Existing representation of piecewise-linear functions as multidimensional doubly linked arrays of orthotopic regions.

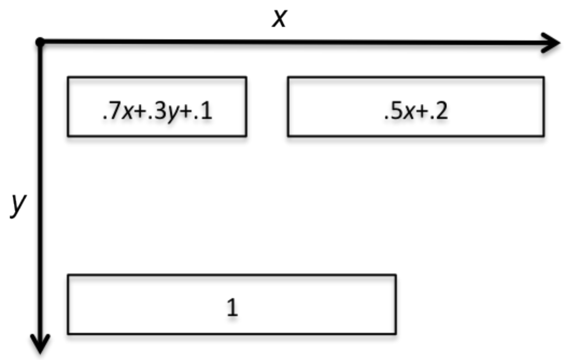

Figure 2: Sparse(r) representation with explicit orthotopic regions for areas with non-default (i.e., non-zero) functions.

What has recently been developed and implemented is a sparse(r) representation[1] that combines a default value for regions, typically zero, with explicit representation of only those regions whose functions differ from this value (Figure 2). This yields efficiency improvements by omitting the explicit representation and processing of default regions; and by eliminating the need for an array of regions, and thus for partitioning regions with uniform functions simply because other regions have more restricted spans. Both of these optimizations are evident in comparing Figures 1 and 2, where the default (zero) valued regions disappear in moving to Figure 2, and pairs of adjacent regions with the same function are coalesced.

Such an optimization is particularly crucial for forms of mental imagery, such as when a composite image is represented as a stack of occupancy planes – one per object – as, for example, shown for the Eight Puzzle in Figure 3. Here, each tile (including the blank) yields one plane, with only one region of a plane non-zero, corresponding to where its tile is located. The existing representation requires slicing this 3D structure – of 9 2D planes – into 81 regions. With the new sparse representation, only 9 regions are required along with a default value of zero.

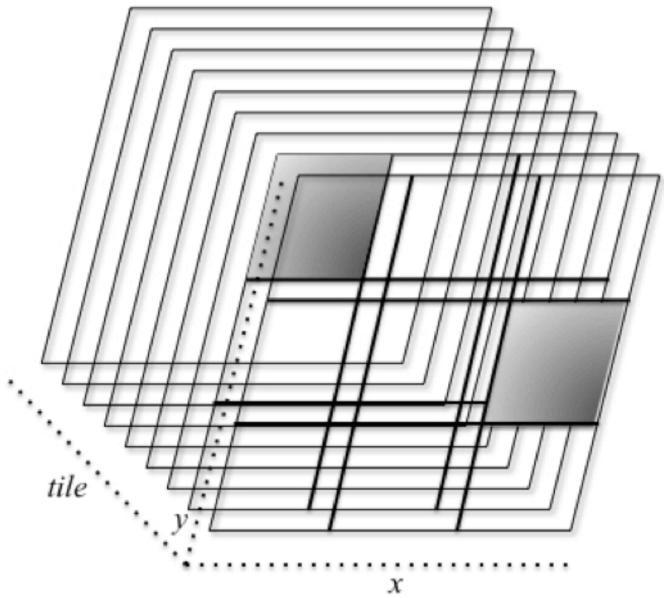

Figure 3: Eight Puzzle board as two continuous dimensions (*x* and *y*) and one discrete dimension (*tile*), yielding a stack of continuous planes, one per tile. Only the region in each plane spanned by its tile has a non-zero (grey) value.

Instead of a doubly linked array of regions, with region boundaries determined by dimension-spanning slices, the sparse representation maintains a simple list of all of the non-default regions plus an ordered list of *projections* along each dimension. Each projection includes the minimum and maximum value along that dimension for a region, along with a pointer to the region.

The two main operations that must be implemented for the summary product algorithm are *combination* (taking two functions and combining their values; typically via product, but sometimes

[1] The existing representation is already somewhat sparse due to its ability to group together large regions with the same value, unlike a truly dense array representation that would represent every single square separately down to some resolution.

via addition or other operations) and *summarization* (taking a single function and eliminating a dimension; typically via integration – or summation for discrete dimensions – or maximum). Using product for combination and integration/summation for summarization yields variable marginals, while using maximum (with combination still via product) yields maximum a priori (MAP) estimation.

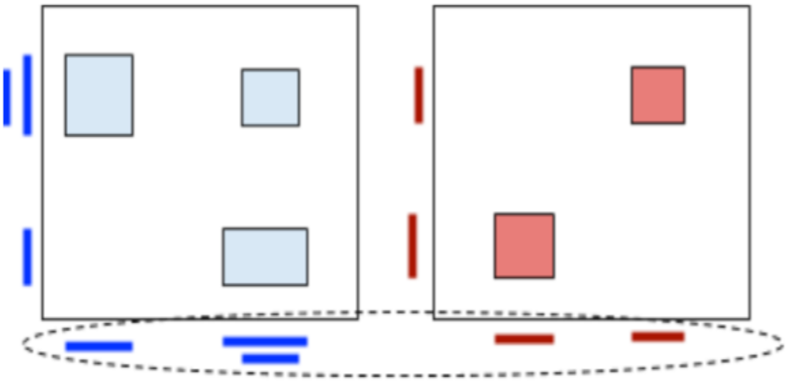

Figure 4: Choosing a dimension along which to traverse the sorted projection lists for generating candidate region overlaps.

In a combination operation, there are two input functions – call them A and B – and a combination function (suppose it's product). Six sets of values must be computed: (1) the default value for the result, which is simply the product of the default values for A and B; (2) the regions obtained by an intersection of a region from A with a region from B, whose value is the product of the two intersecting regions; (3) the regions in A which do not intersect regions in B, so that they can simply be copied and multiplied by B's default value; (4) similarly, regions in B not intersecting anything in A; (5) fragments from regions in A which partially intersect regions in B, forcing us to break off the parts which do not, and give them values as in group 3; (6) similar fragments from B. The two critical processes in computing these sets are finding intersecting (and non-intersecting) regions, and breaking apart regions to deal with partial overlap.

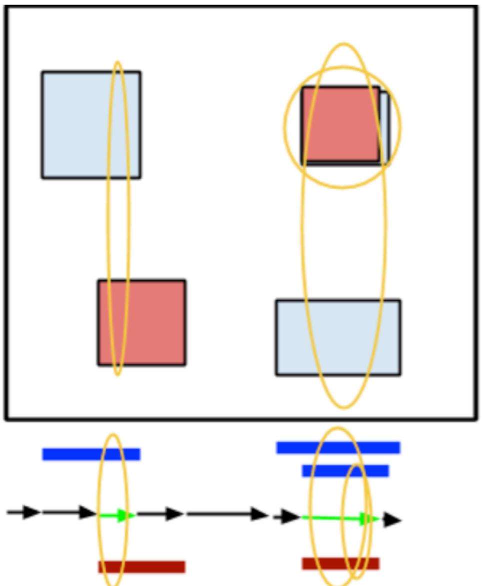

Figure 5: Determining candidate pairs of overlapping regions based on projection lists.

Finding intersecting regions is akin to the problem of detecting collisions in graphics. Here there are two lists of multidimensional orthotopic objects, and we must determine which objects from the first list overlap with those in the second list. The projection index is used to prune the search for intersections. One dimension is chosen and the sorted projection lists for A and B along that dimension are traversed (Figure 4). If a region in A intersects a region in B, then they'll have an intersecting projection, so this can be used as a first pass to find candidate intersections (Figure 5). However, false positives must then be removed by doing a full intersection check (Figure 6).

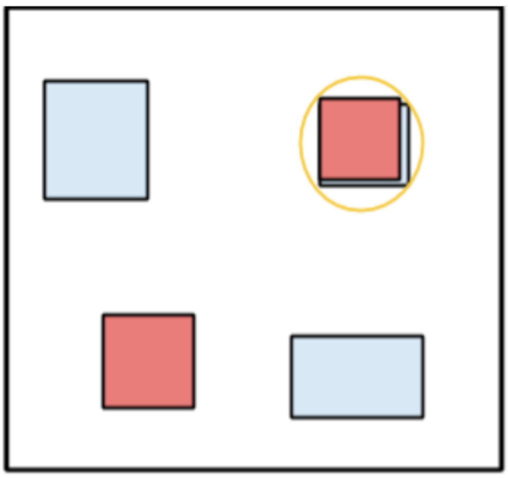

Figure 6: Eliminating false candidates by checking other dimensions.

Once the overlapping regions have been found, they must be split up. Splitting a region from A into parts that intersect with regions in B and non-intersecting parts is not too difficult, but a little care is

needed to ensure the resulting number of regions is linear in the number of dimensions rather than exponential. An easy mistake is to split in all dimensions *at once*. Splitting in each dimension like this results in $O(2^d)$ new regions, where d is the number of dimensions (Figure 7). A better strategy is to break off regions in one dimension at a time, resulting in O(d) regions (Figure 8).

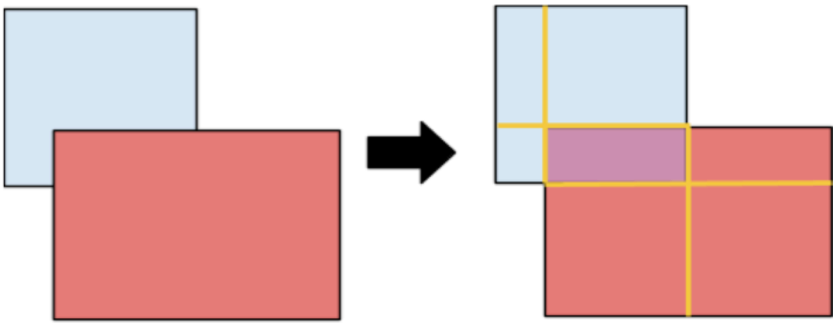

Figure 7: Creating all region splits at once.

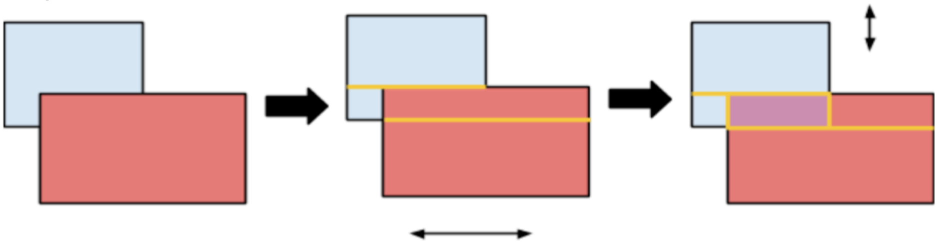

Figure 8: First splitting horizontally, and then splitting the still-overlapping parts vertically.

Summarization is a subtly different process from combination. Given a single piecewise function, the intersections of its regions with itself must be found, ignoring the dimension that is being integrated/maximized away (Figure 9). Having found these intersections-to-be, the regions then need to be split in preparation for combining the intersecting regions along that dimension. So, both major aspects of the combination algorithm are mirrored, except that in the case of summarization, the function is compared to itself rather than to another function, and a specific dimension is ignored. It proved possible to develop general versions of these two operations, so that summarization and combination leverage the same code.

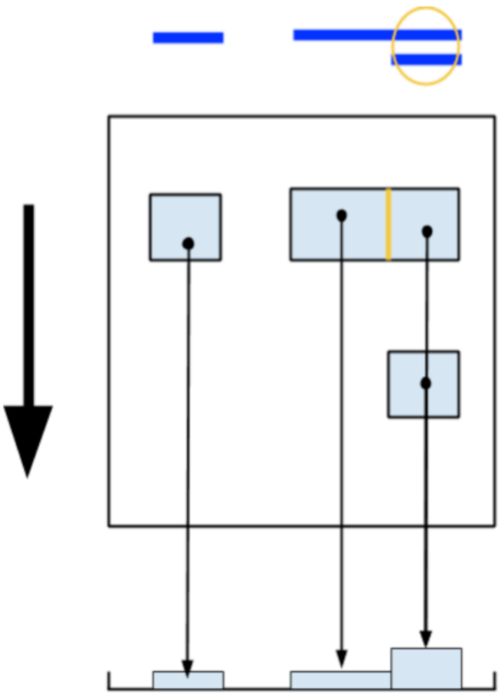

Figure 9: A quick look at integration.

In preliminary results, four existing Sigma models have been tested to provide an initial indication of the speed differences between the sparse and existing formats: a simple naïve Bayes setup, an affine transform test, an Eight Puzzle example, and a shift-reduce parser. The second and third models are directly relevant to mental imagery.

These tests were run within Sigma 12, an older version (dating from October 2012) within which the sparse representation was implemented (a port to Sigma 27, the most recent version, is in progress). Considerable effort has been put into general optimizations of Sigma in the past year, which didn't find their way back into Sigma 12, but the relative comparisons between the two representations within this single version should still be illustrative.

Table 1 shows the preliminary results for the two mental imagery tasks. The table shows the percent of non-empty regions per message in the existing representation – providing a rough maximum on the speed up that is possible with the sparse representation – plus the average runtime (over 3 runs) for the existing and sparse representations, and the percent improvement in runtime. The variance is fairly low, so this gives a decent picture.

Table 1: Preliminary experimental results with the sparse representation in mental imagery tasks.

	Affine	Eight Puzzle
Sparsity (% zero regions)	77%	96%
Existing time (sec)	0.15	2.88
Sparse time (sec)	0.07	0.89
Time Savings (%)	53%	69%

Both of these tasks show a significant speedup – by a factor of 2-3 – although both also fall short of their potential maximum speedup. Our lead hypothesis at this point for why these speedups fall short stems from the sparse representation's higher cost per (explicit) region in determining which regions overlap. There are optimizations under consideration that should significantly ameliorate this, but this is left to future work. In general, the existing representation is more mature, and has thus gone through more representation-specific optimization over the years. With further optimization of the sparse representation, the time percentages may more closely approach the sparsity percentages.

Table 2 shows the preliminary results for the other two tasks, both of which are slower – with the parser being much slower – when the sparse representation is used. The naïve Bayes task is almost twice as slow, with the explanation likely being the same as just discussed for the mental imagery tasks, but with the reduced sparseness here leading to a slowdown rather than just to a reduced speedup. The same issue almost certainly exists in the parser as well, but there must be at least one additional issue causing this rather sparse problem to slow down by a factor of 20. Further analysis has yielded one possibility, concerning the detection of duplicate intersections during summarization, that may account for the excess slowdown. This looks to be fixable, but has also been left to follow on work, as has determining whether there are any other issues involved, and thus optimizations to be investigated.

Table 2: Preliminary experimental results with the sparse representation in two other tasks.

	Naïve Bayes	Parsing
Sparsity (% zero regions)	56%	82%
Existing time (sec)	0.03	5.68
Sparse time (sec)	0.05	115.91
Time Savings (%)	-40%	-95%

As the sparse representation is better understood, it may prove useful to explore hybrid graphs, in which different functions are represented in different manners in distinct parts of the factor graphs. It may also be worth considering other representations for piecewise-linear functions; for example, it may turn out that spatial trees – such as R-trees or BSP-trees – will provide a superior alternative to both of the representations discussed here.

In addition to the potential for speedups, the sparse representation also sets the stage for two further important developments. The first development is a generalization from orthotopic to *polytopic* regions, which should not only enable further coalescing of regions with identical functions – enabling

fewer regions to be used in representing complex objects – but more importantly it should enable rotations that are not limited to multiples of 90° by providing a region representation whose boundaries need not be axially aligned. The second development is the possibility of message passing that is more incremental, just forwarding regions of functions that have changed, and thus yielding even more efficiency. Both of these are good candidates for follow on work.

Beyond the sparse representation, this work as a whole on mental imagery has yielded an approach to incorporating continuous mental imagery into a cognitive architecture without simply bolting on a separate module with an API. Sigma's underlying mechanisms are functionally elegant enough to support mental imagery in a manner that is uniform with its other forms of processing, and thus integratable with them at a very fine granularity. As such, it is an important overall step towards functionally elegant grand unification, with the development of the sparse representation also providing a key step towards sufficient efficiency. One critical future direction along this general path is to return to the issue of a more compact and efficient representation for noisy continuous functions, whether via (mixtures of) Gaussian, particle filters, or some other approach. The other critical future direction – and what was originally proposed as Task 2 on this effort – is integrating true visual perception, including behavior recognition and adaptation, into Sigma in a manner that meets the three desiderata mentioned in the introduction and combines synergistically with both mental imagery and higher level cognition.

Sigma as a whole, through additional funding from the Army Research Laboratory (ARL) and the Office of Naval Research (ONR), is also making progress along other critical paths towards a cognitive architecture that meets the three desiderata. This includes developing social capabilities within Sigma, such as Theory of Mind; broadening Sigma's learning capabilities; developing models of speech recognition and language understanding that are integrated tightly with each other and with cognition; and developing prototype virtual humans. We are constantly seeking functionally elegant paths towards increased grand unification and optimizations that lead it closer to sufficient efficiency. We are also now increasingly looking for useful applications of Sigma.

Publications and Significant Collaborations that resulted from this project:

b) papers published in peer-reviewed conference proceedings,

Rosenbloom, P. S. (2011). From memory to problem solving: Mechanism reuse in a graphical cognitive architecture. *Proceedings of the 4th Conference on Artificial General Intelligence* (pp. 143-152). Mountain View. CA: Springer. **Kurzweil Award for Best AGI (Artificial General Intelligence) Idea**

Rosenbloom, P. S. (2011). Mental imagery in a graphical cognitive architecture. *Proceedings of the 2nd International Conference on Biologically Inspired Cognitive Architectures* (pp. 314-323). Arlington, VA: IOS Press.

Chen, J., Demski, A., Han, T., Morency, L-P., Pynadath, D., Rafidi, N. & Rosenbloom, P. S. (2011). Fusing symbolic and decision-theoretic problem solving + perception in a graphical cognitive architecture. *Proceedings of the 2nd International Conference on Biologically Inspired Cognitive Architectures* (pp. 64-72). Arlington, VA: IOS Press.

Rosenbloom, P. S. (2011). Bridging dichotomies in cognitive architectures for virtual humans. *Proceedings of the AAAI Fall Symposium on Advances in Cognitive Systems*.

Rosenbloom, P. S. (2012). Graphical Models for Integrated Intelligent Robot Architectures. *Proceedings of the AAAI Spring Symposium on Designing Intelligent Robots: Reintegrating AI*.

Rosenbloom, P. S. (2012). Deconstructing reinforcement learning in Sigma. *Proceedings of the 5th Conference on Artificial General Intelligence* (pp. 262-271). Oxford, UK: Springer. **Kurzweil Award for Best AGI (Artificial General Intelligence) Paper**

Rosenbloom, P. S. (2012). Extending mental imagery in Sigma. *Proceedings of the 5th Conference on Artificial General Intelligence* (pp. 272-281). Oxford, UK: Springer.

Rosenbloom, P. S., Demski, A., Han, T. & Ustun, V. (2013). Learning via gradient descent in Sigma. *Proceedings of the 12th International Conference on Cognitive Modeling* (ICCM 2013). Ottawa, Canada.

c) papers published in non-peer-reviewed journals and conference proceedings,

Rosenbloom, P. S. (2013). The Sigma cognitive architecture and system. *AISB Quarterly*, *136*, 4-13.

d) conference presentations without papers,

Rosenbloom, P. S. (2011). Mechanisms and Levels. Invited panel presentation at the *AAAI 2011 Fall Symposium on Advances in Cognitive Systems*. Arlington, VA.

Rosenbloom, P. S. (2011). Cognitive Architectures for Virtual Humans. Presentation at the *31st Soar Workshop*. Ann Arbor, MI.

Rosenbloom, P. S. (2012). Accelerating Architecture Evolution by Striving for Grand Unified Architectures via Functionally Elegant Implementation Levels. Invited panel presentation at the *21st Annual Conference on Behavior Representation in Modeling and Simulation (BRiMS 2012)*. Amelia Island, FL.

Rosenbloom, P. S. (2013). Leveraging Graphical Models in Support of Cognitive-Architecture-Based Robotics. Invited panel presentation at the *12th International Conference on Cognitive Modeling (ICCM 2013)*. Ottawa, Canada.

Rosenbloom, P. S. (2013). The Sigma Cognitive Architecture. Invited Keynote talk at the *IJCAI 2013 Workshop on Intelligence Science*. Beijing, China.