

Signal Transduction: NLP Based Information Extraction Using a Multi-agent System

William J. Gillis², Salim Khan¹, Carl J. Schmidt² and K. Vijay-Shanker¹,
¹Department of Computer and Information Sciences and ²Department of Animal and Food Sciences, University of Delaware, Newark, DE. 19717

William J. Gillis: wgillis@cis.udel.edu

Salim Khan: skhan@cis.udel.edu

Carl J. Schmidt: schmidtc@udel.edu

K. Vijay-Shanker: vijay@cis.udel.edu

Keywords:

Data Mining

Signal Transduction

Software Agents

Knowledge Base

Natural Language Processing

Abstract: Metabolism is the machinery of life and signal transduction provides the regulatory mechanisms to control that machinery. Due to the complexity of signal transduction pathways, computational approaches are needed to aid the biologist in integrating available knowledge and in the formulation of testable hypotheses. Our objective is to apply multi-agent systems to build a comprehensive system for study of signal transduction. The agent driven system draws inferences and hypothesizes pathways, evaluates predictions and identifies information relevant to signal transduction from both web based and literature resources. We describe here a system for agent directed natural language processing to extract information from journal articles. An interface was developed to permit curation of the NLP results and deposition of accepted results into a knowledge base.

Motivation: The advent of high-throughput methods has revolutionized the field of biology, creating an information explosion in their wake. The daily accretion of experimental data from sequencing projects, scientific literature, gene array experiments and other high volume data pipelines, has prompted calls (Karp, 2001) to encode scientific theories into symbolic form so that inference engines may be employed to generate promising hypotheses and flag potentially erroneous results. In this paper, we describe just such a formalism driven by a multi-agent system to collect heterogeneous information from multiple web-based resources. The objective is to populate a knowledge base (KB) that can facilitate the generation of hypothetical pathways, which can then be tested, queried and, quantitatively and qualitatively simulated (Khan et al., 2003a, Khan et al., 2003b). While our research thus far is limited to gathering information related to signal transduction (ST) pathways the discussion is relevant to the generation of other types of biological pathways, such as those of metabolism and gene regulation.

Living systems exhibit great robustness and versatility in adjusting their intracellular molecular machinery to changes in the external environment. The cellular processes by which cells detect, convert and internally transmit information regarding the external environment are collectively referred to as signal transduction (ST) pathways. The study of ST pathways is vital to our understanding of life, as these pathways control normal processes of reproduction, development and homeostasis. In addition, errors in ST pathways are responsible for many diseases including cancer, diabetes and neural disorders. ST is a complex process, integrating many different types of signals with multiple pathways frequently responding to individual signaling events. A major challenge confronting biology is understanding how these pathways become integrated into a living system. The vast scale of ST makes it difficult for individual researchers to be cognizant of all the different ways changes in gene expression, protein or metabolite levels can impact an organism. To facilitate such understanding, computational approaches are necessary to mine information and place it in a context that can be queried by both humans and machines.

There are two major types of resources for information mining relevant to ST: web accessible databases and published literature. Ultimately, these disparate sources of

information need to be integrated under a single computer-accessible framework so that automated reasoning mechanisms can utilize this information when hypothesizing pathways. One approach to integrate such resources is the use of multi-agent information gathering systems, which have several attractive features (Decker et al., 2001) including ability to deal with:

1. information that is available from many distinct locations
2. information content that is heterogeneous
3. information content that is constantly changing
4. new types of analysis and sources of data that are appearing constantly

Web accessible systems relevant to ST include a variety of genomic and proteomic databases, gene ontologies, and many of other resources (see: Nucleic Acids Research Database Issue, 2003). While challenging because of their scale, such sites are typically amenable to electronic query, which is essential to the application of multi-agent systems to knowledge base development. More challenging is the integration of knowledge that is recorded in the literature, either in the form of abstracts, via PUBMED, or complete journal articles that are available by subscription via publishers web sites. This work describes our efforts to integrate multi-agent systems with Natural Language Processing (NLP) for the extraction of information from the literature into a ST knowledge base.

Multi-agent System: We have used DECAF, a multi-agent system toolkit based on RETSINA (Sycara et al, 1996, Decker et al,1997a, Decker et al., 1997b]: and TAEMS (Decker et al., 1993, Wagner et al., 1997], to construct a prototype multi-agent system for the automated extraction, annotation and knowledge base storage of ST pathway related data. The RETSINA approach consists of three general classes of agents (Sycara et al., 1996, Decker et al., 1997b):

- Information Extraction Agents, which interact directly with external data sources, i.e. wrapping sensors, databases, and web pages.
- Task Agents, which interact only with other agents to handle the bulk of the information processing tasks. These include both domain-dependent agents that take care of filtering, integration, and analysis; also domain-independent “middle agents” that take care of matchmaking, service brokering, and complex query planning.
- Interface Agents that interact directly with the end user.

In our current context, information extraction agents (IEA) are responsible for targeting queries to either PUBMED, or specific journals, relevant to a topic. As a prototype for development of this system, we have populated a knowledge base for information relevant to phosphorylation events in response to EGF. The IEA was used to search six publicly available life sciences journals for information relevant to this topic, extracted appropriated sentences, and returned the extracted information to a database. Once the IEA is complete, task agents activate NLP software to parse the sentences, identifying noun phrases that define both agents (in this case proteins, such as kinases, which phosphorylate other proteins) and targets (which get phosphorylated). When parsing is complete, a task agent then notifies a curation expert that new information is available. Subsequently, the expert uses an interface agent to examine the results of the

NLP analysis, make any necessary corrections, and deposit this information in a PARKA-DB. Using a PARKA-DB knowledge base allows efficient, modern relational data storage on the back end and query as well as limited KB inferencing (Hendler et al., 1996). The interface agent also records corrections and the original NLP output to a log. This log can be used to apply machine learning approaches to improve the accuracy of the NLP software. A second interface agent then permits queries of the resulting database.

The curation interface agent provides a fundamental link in the development of the ST knowledge base. In general, there is a need for quality control to ensure that the information extracted and used to populate the KB is correct. This is particularly so in the current state of NLP and text mining technologies. For human curation purposes, the information extracted from the NLP component is displayed in a form that is easy to read. Figure 1 shows the screenshot of our interface for curation. The interface shows the

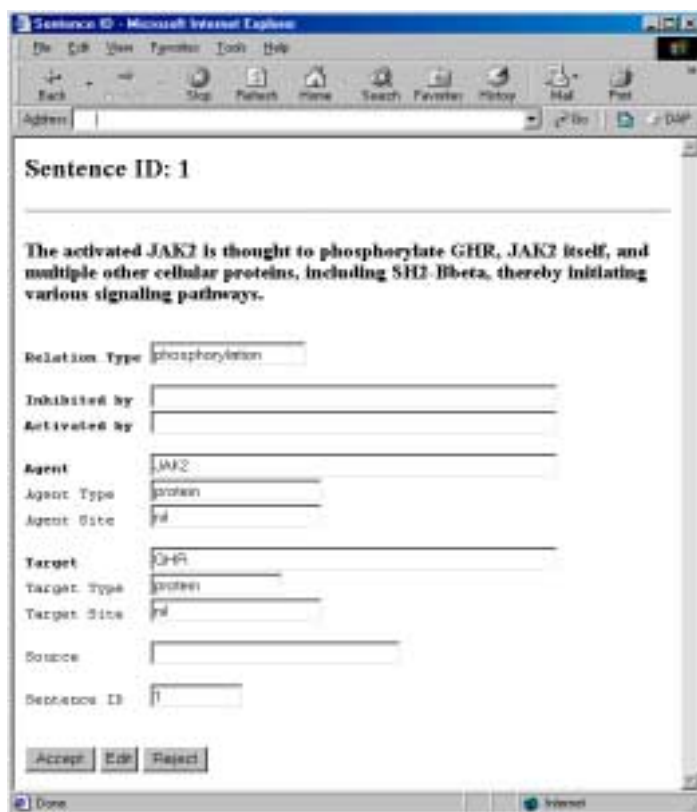


Figure 1. Curation Interface. This interface displays the sentences to be validated along with the NLP assignments relevant to agents and targets. The source will provide a direct link to the article from which the sentence was obtained.

information extracted together with the sentence from the article from which this data was obtained.

The curator can correct or add information in any of the slots. For example, for the sentence shown in Figure 1, the NLP component identified only one target and missed the other targets that are conjoined in the object phrase for the verb phosphorylate. The curator can add the new information in such cases. The curator can also reject the information extracted. For example, in the sentence shown in Figure 1, the authors are not necessarily stating a fact but rather expressing a belief. The NLP system is not capable of recognizing when sentences express beliefs, hypothesis and conjectures, and the curator must decide whether information extracted from such sentences should be

accepted or rejected. The extracted information is stored in a temporary KB and stored in a permanent KB only after the curator hits the accept button. Each extracted fact is given

a Fact-id which remains linked to the Sentence-id . This permits subsequent users to link back to the journal article that yielded the fact of interest.

ST Ontology Development: It is through multi-agent systems and curator interfaces that a ST knowledge base is being developed. However, a knowledge base also must incorporate an ontology linking the resident data into a framework that provides for complex queries and for computation based hypothesis generation. To this end, we have

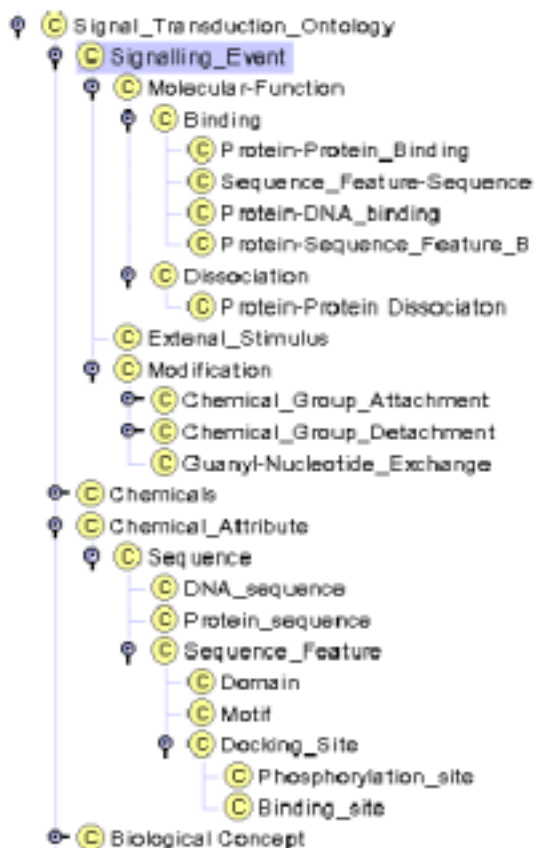


Figure 2. Screen shot from Signal Transduction ontology.

been developing a ST ontology (Figure 2). This ontological framework allows us to capture and relate information with different depths of biochemical detail, as well as guiding the integration of heterogeneous data spanning a vast spectrum of sources. However, in designing the ST ontology, we did not have to start from scratch. The ST Ontology has been built on the concepts provided by a variety of accepted bio-ontologies. Among these are Tao, a generalized ontology for bioinformatics applications used by Tambis group (Baker, et al., 1999), an ontology developed for the analysis of metabolic pathway information by the Biocyc (Ecocyc) group (Karp et al., 1994), a biological process modeling ontology, developed by the Stanford Medical Informatics group (Peleg et.al., 2002), and an ontology for Gene Annotation from the Gene Ontology

Consortium [Consortium 2000]. The ontology is divided along the following principles:

- *Signaling Event:* Captures all the processes that participate in ST pathways and help transfer the signal, including post-translational modifications, external events, and levels of various interaction information.
- *Chemicals:* Provides a class hierarchy for the biomolecules found to participate in the ST domain.

Slot name	Documentation	Type	Allowed Values/Classes	Cardinality	Default
has_agent		Instance	Chemicals	0..1	
activated_by		Instance	Signaling_Event	0..1	
has_function		String	Protein	0..1	
attachAs		Class	Protein	0..1	
is_go_process		Class	Protein	0..*	
at_activation_site		Integer		0..1	
target_site		Instance	Sequence_Feature	0..1	
agent_site		Instance	Sequence_Feature	0..1	
has_target		Instance	Macromolecules	0..1	
activated_by		Instance	Signaling_Event	0..1	
at_go_location		Class		0..1	

Figure 3 Phosphorylation Event slots within the Signal Transduction Ontology.

slots in the class template contain information about appropriate values and class information. This information can ultimately be used to refine the text mining,

NLP Algorithm: To identify the participants of ST pathways that are mentioned in text, we have to locate the noun phrases that refer to those components. To locate such noun phrases, we have developed a noun phrase chunker which is integrated with the module that recognizes biological named entities in the text. This name detector has been described in detail in Narayanaswamy et al. (2003). One of the main attributes that differentiates this name detector from others in the biomedical domain is that it recognizes various types of named entities. These entities include *protein*, *protein parts* (names of different subunits, domains, motifs etc. of proteins/genes), *chemicals*, and *source terms* (such cell names, tissues etc.). Work is currently under progress to extend this list of categories.

The ability to recognize a variety of classes helps in at least two ways. For example, to identify the participants of a phosphorylation reaction, we need to recognize not only protein names, but also that of chemical compounds (as they too can serve as a participant) along with protein parts so that we can identify the site of phosphorylation in the target protein. The other benefit of recognizing names other than just proteins is that the precision of protein name recognition can potentially improve (Narayanaswamy et al. 2003). This is because most name detectors identify name occurrences by using surface clues such as use of numerals, capitalization or Greek characters. Of course, names of

- *Chemical Attributes:* Conceptual decomposition of chemicals into functional units that help propagate the signal despite lacking independent physical existence. Includes protein constituents such as domain, motif, and active sites.
- *Compartmentalization:* Provides a controlled vocabulary for the distinct sub-compartments into which the cells are divided.

We have created this framework (Figure 2) with the aid of Protégé (Yeh et al., 2003), and have implemented this framework within the Mysql relational database. Figure 3 displays a view of the event class phosphorylation as represented in the ontology. Clearly, the slots in Figure 1 are to be related to the slots in this class.

As can be seen from Figure 3, the

other classes can also share these characteristics. Hence, because of the ability to identify when a name belongs to a class other than protein, we are able to improve the precision of protein name detection.

The name detector is a rule-based method that works by locating three types of words: c-terms, f-terms, and h-terms. C-terms are those words that bear surface clues such as capitalization and serve to locate the presence of names but rarely ever offer clues to the class of the names. Exceptions to these (i.e., ones that do help in identification of class information) are some c-terms that are identified by using a list of chemical roots and suffixes obtained from the IUPAC naming conventions for chemical compounds. F-terms are a generalization of the notion of function terms used in the PROPER (Fukuda et al., 1998) name recognizer. F-terms help in locating as well as classifying names. For example, one f-term for proteins is the term receptor. After the identification of c-terms and f-terms, concatenation rules are used to concatenate individual terms and more generally extend words to the full name. These rules also incorporate English syntax by using words to the right to provide the class information (as the head nouns of English noun phrases appear to the right). In many cases, a detected name need not have been classified. e.g., because they may not contain any f-term. In such cases, the can provide information about the name's category. The informative words in the context are called h-terms. These words are not part of the names but are often located nearby. When these terms do appear near a name, they provide strong clues regarding the class of the name. For example, if "expression of" immediately precedes a name, we assign the class protein/gene to the name if it had not previously been classified. While the original list of c-terms, f-terms, suffixes, and h-terms were hand-compiled, publicly available annotated corpora and databases have been used to build up a comprehensive list of terms.

Canonical Names: Authors seldom use official or standard names of proteins in their papers and synonyms abound for many protein names. These issues raise some important questions as to how the participants of ST pathways that are mined from text should be stored. If the names as they appear in the text are used in storing the information, then issues of replication of data must be addressed. Two different papers may mention the same interaction using different synonyms for the participants in question. Noting that this is replication of information is of much importance, not only to eliminate redundancy, but also to bolster the confidence we might have in the validity of that piece of information. This is one of the motivations that have led us to consider storing the proteins using a canonical scheme. We have built a system that has extracted all names and synonyms from Swiss-Prot (Boeckmann et.al, 2003). Any name that we detect from the text can be compared with this list. In making these comparisons, we allow for some minor mismatches, such as in capitalization or dropped hyphens. If the name in the text can be matched with a name in the compiled list, then the official name (as listed in Swiss-prot) and Swiss-prot accession numbers are retrieved. This circumvents the problem of multiple names for protein entities, but the accession number now provides an invaluable resource. The user of our system can use it to manually access information from Swiss-Prot or other sources.

NLP and Information Extraction: We are currently focusing on extracting information about post-translational modifications, and on phosphorylation in particular. However, patterns we use for extracting information about phosphorylation can be, without any major changes, used to extract information about other post-translational modifications such as acetylation, or methylation. At the most basic level, the NLP system operates in the following manner. Given a journal article or an abstract, the system scans for the verb phosphorylate or one of its morphological variants. These variants include phosphorylates, phosphorylated and even the nominalized noun form, phosphorylation. For each of these variants, a number of patterns are specified. Given a sentence with one of the inflected forms of phosphorylate, the appropriate patterns are used for matching and based on the match, the appropriate arguments are identified. Among the inflected forms, phosphorylated requires the most number of patterns. These patterns capture the fact that phosphorylated may be a (active) past tense verb, or in past participle form in which case patterns for passivized forms also need to be explored. In addition, phosphorylated can also have an adjectival form, as in “the phosphorylated protein”. In the adjectival form, it can be noted that only the target of the phosphorylation is being mentioned. Similarly, we have noticed that the participants are often only partially specified with the use of the noun form phosphorylation and quite frequently either the agent or the target site are not specified in the same sentence.

The patterns that we use are characterized by their simplicity and their use of class information. The patterns specifically identify the participants by mentioning noun phrases of appropriate type (protein, chemical, source, and protein parts) and their relative positions with respect to the inflected form of phosphorylate and each other. Thus, the type information assigned by noun phrase and name detector play a critical role here. The coverage of these patterns is greatly enhanced by detecting various types of phrases. We detect appositives, prepositional phrases in addition to noun and verb groups. Thus, for example, “A phosphorylates B” and “A, a member of the MAP kinase family, has been shown to phosphorylate B” are recognized by the same pattern. In the latter sentence, the appositive “a member of the MAP kinase family” is detected and eliminated from consideration for the purpose of matching of the patterns. Also, as far as matching the patterns are concerned, the verb group “has been shown to phosphorylate” is treated identically with the simpler verb group in the first of the two sentences. Note that while “has been shown” is in passive form, the phosphorylate verb is not. In contrast, our system will detect the passivization in “A has been shown to be phosphorylated” and use this information to tag A now as a target rather than the agent of phosphorylation.

The system is also capable of extracting information about cell and tissue mentioned in the same sentence. As can be expected for a manually designed rule based system, the system enjoys high precision. Clearly, there is scope for improvement of recall. However, as outlined above, this is not mainly due to insufficient number of patterns. Many of the missed arguments are due to the fact that we do not recognize relative clauses (which need to be treated like appositives as discussed above, but differ that they often themselves contain the information being stated about phosphorylation). Another cause for missing participants is that we are only now treating conjunction of noun phrases. Currently, even for a straightforward sentence pattern such as “A

phosphorylates B and C”, we note that A is the agent and that B is the target, skipping information about C. Finally, because our patterns are tied so closely with noun phrase classes, if the system is unable to judge that a name is one of the protein class and leaves it as unclassified, currently our system will not choose it as a argument even if it is clear from its syntactic position (such as subject). Recent enhancements to our name detector and in particular the recent development of a dictionary of protein names and synonyms that we have built should greatly alleviate this problem.

Knowledge Base and Text Mining: So far, we have discussed using NLP techniques to populate the knowledge base so that this information may be used to draw inferences and hypothesize pathways etc. However, the ontology and the populated knowledge bases can also be used to improve the text mining. For example, the knowledge encoded can be used to improve the recall by suggesting possible participants when the patterns used in NLP-based text mining are not sufficient. As an example, consider the use of the word phosphorylation. As mentioned earlier, many occurrences of phosphorylation do not mention the agent within the same sentence and the NLP agent then returns an incomplete record with the agent slot unfilled. However, it is also the case that frequently the agent is mentioned in the neighboring context. By locating the names of proteins nearby, identifying their type by using the protein ontologies and using the information in our phosphorylation ontology (e.g., that kinases are likely to be agents of phosphorylation, or that member of some families are not likely to be agents) can allow the text mining agent to propose the agent (for inspection by human curator).

More generally, the NLP data mining agent is responsible for using the ontology and data of the KB to capture specific slot information and disallow erroneous mismatches that would otherwise occur from using a purely NLP agent. For example, one common error during NLP extraction was the assignment of EGF as an agent (or kinase). However, EGF does not function as a kinase, but rather stimulates kinase activity of other proteins. Based upon information available in Swiss-Prot, the DM agent understands that EGF cannot function as a kinase, and will disallow assignment of EGF as a agent. This improves the accuracy of the final NLP output. Greater integration between the KB and NLP component will allow us to use patterns that are more flexible. Usually such looser patterns allow for higher recall at the expense of precision, but with the integration with the KB, we believe that many errors can be caught and hence precision does not have to be sacrificed.

Future Objectives: This approach can be extended beyond simple protein modifications to incorporate other types of interactions that mediate signaling events. For example, protein-protein interactions play an important role in many steps of ST, and such information could be extracted from the literature. Furthermore, protein accession numbers provide access to a variety of databases that can be used for information extraction. In combination, both web based resources, and NLP based literature data extraction, should provide important assets for computational analysis of ST pathways.

References:

Baker PG, Goble CA, Bechhofer S, Paton NW, Stevens R, Brass A. An ontology for bioinformatics applications. *Bioinformatics*. 1999 Jun;15(6):510-20.

Boeckmann B., Bairoch A., Apweiler R., Blatter M.-C., Estreicher A., Gasteiger E., Martin M.J., Michoud K., O'Donovan C., Phan I., Pilbout S. and Schneider M. The Swiss-Prot protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res*. 31:365-370(2003).

Decker, K.S. and Victor R. Lesser. Quantitative modeling of complex computational task environments. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 217–224, Washington, July 1993.

Decker, K.S., A. Pannu, K. Sycara, and M. Williamson. Designing behaviors for information agents. In *Proceedings of the 1st Intl. Conf. on Autonomous Agents*, pages 404–413, Marina del Rey, February 1997a.

Decker, Keith S. and Katia Sycara. Intelligent adaptive information agents. *Journal of Intelligent Information Systems*, 9(3):239–260, 1997.

Decker, K., X. Zheng, and C. Schmidt. A multi-agent system for automated genomic annotation. In *Proceedings of the 5th Intl. Conf. on Autonomous Agents*, Montreal, 2001

Fukuda, K., Tsunoda, T., Tamura, A., and Takagi, T. Information Extraction: Identifying Protein Names from Biological Papers. *Proceedings of the Pacific Symposium on Biocomputing*. Hawaii, January 1998.

The Gene Ontology Consortium. 2001. Creating the gene ontology resource: design and implementation. *Genome Research* 11: 1425-1433.

Hendler, J., and Merwyn Taylor Kilian Stoffel. Advances in high performance knowledge representation. Technical Report CS-TR-3672, University of Maryland Institute for Advanced Computer Studies, 1996. Also cross-referenced as UMIACS-TR-96-56.

Karp, P. D. 2001. Pathway databases: A case study in computational symbolic theories. *Science* 293:2040–2044.

Karp PD, Paley SM. Representations of metabolic knowledge: pathways. *Proc Int Conf Intell Syst Mol Biol*. 1994;2:203-11.

Khan, S.; Gillis. W.; Schmidt, C.; Decker, K. A multi-agent system driven AI Planning approach to biological pathway discovery. To appear in 13th Intl. Conference on Automated Planning and Scheduling, Trento, Italy, June 2003a.

Khan, S.; Makkena, R.; McGeary, F.; Gillis, W.; Schmdt, C.; Decker, K. A Multi-Agent System for the Quantitative Simulation of Biological Networks. To appear in 2nd Intl. Joint Conference on Autonomous Agents and Multi-agent Systems, Melbourne, Australia, July 2003b.

Narayanaswamy, M., Ravikumar, K., and Vijay-Shanker, K. A Biological Named Entity Recognizer. In Proceedings of Pacific Symposium on Biocomputing, 427-438, Hawaii, January 2003.

Nucleic Acids Database Issue: The Molecular Biology Database Collection: 2003 update
Nucl. Acids. Res. 2003 31: 1-516.

Peleg M, Yeh I, Altman RB. Modelling biological processes using workflow and Petri Net models. *Bioinformatics*. 2002 Jun;18(6):825-37.

Sycara, K., K. S. Decker, A. Pannu, M. Williamson, and D. Zeng. Distributed intelligent agents. *IEEE Expert*, 11(6):36–46, December 1996.

Wagner, T., A. Garvey, and V. Lesser. Complex goal criteria and its application in design-to-criteria scheduling. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, Providence, July 1997.

Yeh I, Karp PD, Noy NF, Altman RB. Knowledge acquisition, consistency checking and concurrency control for Gene Ontology (GO). *Bioinformatics*. 2003 Jan;19(2):241-8.