An Intelligent Workflow Construction Support System for MATLAB

UTC for Computational Engineering
Hakki Eres and Andy Keane: CEDG School of Engineering Sciences

Introduction

In the process of intelligent workflow support there are two distinct roles. The first role is that of the expert, who possesses deep knowledge in the domain, and in our case design optimisation, the other is the role of the end-user, who still needs help and guidance from an automatic and intelligent workflow support system in order to develop complex workflows that require some knowledge in design optimisation.

The overall system architecture of the SIMBAT intelligent workflow construction support system is given in Fig. 1, and the distinct boundary between the domain expert (or the knowledge engineer) and the end user is shown with a dotted line. The domain expert uses a publicly available, open-source, and widely used ontology editor called Protégé. This program allows the domain expert to construct expert-domain knowledge in a graphical environment.

As the knowledge model grows, it gets more and more difficult to maintain the consistencies of the model manually. Therefore, automatic reasoners are needed to check the consistencies of the knowledge model. These reasoners can further be used to infer additional knowledge based on the asserted knowledge model. A snapshot of the engineering design search and optimisation ontology model is given in Fig. 1.

When the domain expert is satisfied with the consistencies and the usefulness of the knowledge model they can save the model in an OWL file and upload the file to a triple store, in our case Sesame server, depending on the installation, actual data is stored in a local database application like MySQL, in memory, or in a local filesystem. The access to the knowledge repository is password protected, and by using a very simple user interface the domain expert can use the OWL model in the repository. During this process the Sesame server checks the consistencies of the model. If one more time, and warns the domain expert if inconsistencies exist.

MATLAB Interface

We assume that the end user needs to create an optimization workflow or script in the MATLAB environment, but they have somewhat limited knowledge on existing methods and techniques to apply to their design problem at hand. Therefore, we use MATLAB’s Jazz support along with the Sesame API to implement MATLAB functions to provide the end-user an intelligent workflow construction environment. At the simplest level this will involve a basic help system that offers solutions on the details of different algorithms and optimization techniques, and at a more complex level the user might need to know what would be an appropriate pop-out site for a genetic algorithm, or whether there are any methods for a specific problem that would provide a better convergence. The MATLAB interface consists of some basic functions that perform ontology upload, test the status of Sesame triple store, and more complex functionality to perform various database queries.

SeSQL Queries

When users are sure that the knowledge server is up and running, they can perform various queries on the Sesame triple store by using implemented MATLAB query functions. In Fig. 3, the user first searches for a comment attached to a specific concept called “Currency,” and then they use a wildcard to search for all the comments of all concepts.

For the next example we have created two repositories, the first one only holds the asserted ontology which is not capable of using reasoning techniques, and the second repository can use automatic reasoning techniques to infer additional knowledge on the same ontology. When the user queries the concept “Currency” from the asserted repository, the query returns only one result. However, the inferred repository returns more knowledge to the end user as shown in Fig. 4.

In the optimization ontology generated by the Computational Engineering Design Group there are specific equivalent names that are used to describe the same-optimisation algorithm. For these type of concepts the user can search for another equivalent MATLAB function, called as, QuerySeSQLseSQLfunctionSeSQLendfunction, that allows the end user to perform equivalence queries as shown in Fig. 5.

Note that the inferred repository returns automatically generated knowledge.

Conclusions

An intelligent workflow construction support system has been implemented in order to aid domain experts to construct a useful ontology and to help end-users to query and retrieve this knowledge easily by using open-source software packages. This work demonstrates that by integrating only open-source software, it is possible to implement a useful environment that allows ontology creation and reasoning, as well as knowledge maintenance and querying capabilities within a workflow construction environment.

Acknowledgements

This work is supported by the SIMBAT project (European Commission contract number IST-2004-504696).