SOFTWARE MODULES FOR OPTIMIZATION OF SPECIFICATION OF HETEROGENEOUS MULTI-ROBOT SYSTEM

Vitalijs Komasilovs
Latvia University of Agriculture
vitalijs.komasilovs@gmail.com

Abstract. Economic benefit of an industrial company depends on forethought deployment of an industrial production system. Configuration of the robotic system affects the whole production process, and selection of inappropriate option can lead to financial losses of the company. The author proposes a formal procedure for finding an optimal specification of a heterogeneous multi-robot system. The paper presents the concept of the optimization procedure and describes the design of software modules used for implementation of the procedure.

Keywords: multi-robot system, specification optimization, software architecture.

Introduction

Robotic systems are becoming prevalent in many industrial fields, such as intelligent manufacturing [1; 2] or process automation [3]. During the last decade new prominent applications of robotic systems were reported in medicine [4], space exploration [5], disaster management [6] and warfare [7; 8]. Researchers have been noticing tremendous potential of heterogeneous multi-robot systems already for several years [9; 10] and report successful application of such systems [11-13]. The complex task of design and development of effective heterogeneous multi-robot system arises and opens the opportunity of investigating of formal methods for optimization of the system. An elaborate design and deployment of an industrial system increases effectiveness of the production system which, in turn, is one of the means for improving the economic benefit of a company.

Impartial evaluation of design of the heterogeneous multi-robot system is an open research topic. Several applications of formal methods are reported in adjacent fields, for example, industrial robot selection [14; 15] and multi-agent team collaboration [16; 17] problems. The author proposes an advanced optimization procedure [18] for heterogeneous multi-robot systems which support formal evaluation of large number of solution candidates taking into account defined objectives and constraints.

The paper briefly defines an optimization task for a specification of a heterogeneous multi-robot system, describes a concept of a solution for the optimization task and introduces the proposed optimization procedure. A core of the paper presents software used to implement various steps of the optimization procedure. It describes architectural design and implementation peculiarities of each module.

Materials and methods

The scope of the research is limited to domain of heterogeneous multi-robot systems. It aims to development of a formal method for finding the best configuration of the system in terms of defined objectives and constraints - specification optimization procedure. A set of optimization parameters includes the types of agents, their functions and a number of instances used in the system.

The author defines three major concepts of specification optimization task (see fig. 1).

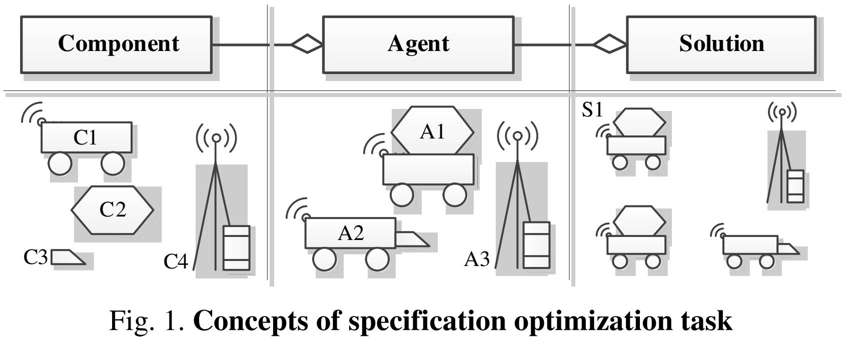

Fig. 1. **Concepts of specification optimization task**

The component defines the ability (function) of the robotic system without an explicit declaration of its realization. Agents are composed from the components and they are either mobile robots or stationary units of a robotic system. Solution is obtained by combining instances of various types of agents in a specification of a heterogeneous multi-robot system [19].

The specification optimization procedure [18] has iterative nature and it consists of eight steps (see fig. 2). First of all, the industrialist defines the business requirements for the system which are subsequently converted to formal optimization objective function. Depending on the analysis of a solution domain various evaluation approaches are applied to find the solution. The main goal of each step of the procedure is to exclude non optimal branches of the solution domain at early stages thereby reducing the number of considered solution candidates.

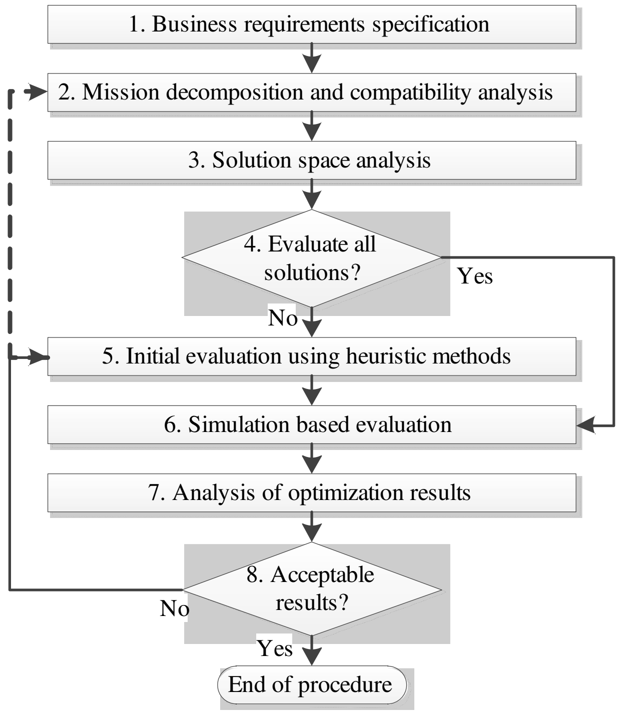

Fig. 2. **Specification optimization procedure**

Results and discussion

The following chapters describe software modules used in steps 3, 5 and 6 of the specification optimization procedure.

CoMBot-Gen – solution space analysis utility

The third step of the specification optimization procedure stands for analysis of a solution domain. Its aim is to estimate the number of feasible solutions based on the provided objective function. In general the solution domain contains valid combinations of possible agents, which in turn are combinations of components. The number of feasible solutions is calculated using Formula (1) [19].

$$g(n) = 2^{2^n - 1} - 1 - r(n) \qquad (1)$$

where n – number of defined components for the mission;
$g(n)$ – number of solutions;
$r(n)$ – number of combinations, which are not solutions.

Function $r(n)$ represents the number of combinations, which cannot be considered as solutions because of missing components. Software was developed in order to support experimental calculations of the function – *CoMBot-Gen*, which stands for Combination Generator for Multi-Robot system specification problem. The main feature of the software is the use of a special algorithm for generation of solutions. The combinations are organized in a tree structure, which in turn allow generation of combinations on demand. According to the algorithm the list of possible agents is sorted at first, and then added to the tree structure on root level. Next, child nodes are populated recursively on demand, when particular parent node is expanded. Only agents positioned behind the current agent within the ordered list are considered as child nodes. This rule ensures that the tree does not contain repeating branches. The solution (the combination of agents) is obtained by selecting the particular node of the tree and recursively fetching agent information from its parent. Fig. 3 shows an example tree for mission defined using three components. The yellow nodes indicate valid solutions, while the grey nodes stand for incomplete agent combinations.

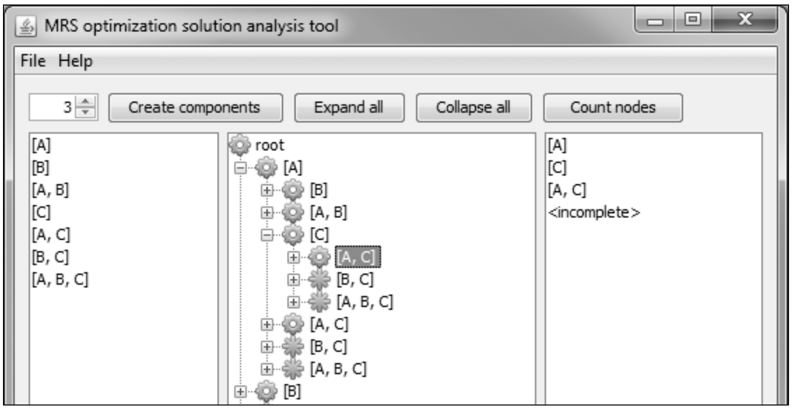

Fig. 3. **Screenshot of *CoMBot-Gen* software**

The values of $r(n)$ function were experimentally obtained for cases with up to 5 components and the trend of the function is obtained. Absolute values of $r(n)$ function grow slightly faster than pure exponential function, which means that the function lags behind $g(n)$ function.

The relative amount of invalid combinations was analysed in the context of all combinations: $r(n)/g(n)$. Despite the fact that absolute amount of invalid combinations grows exponentially, the amount of invalid combinations compared to the total number of combinations converges to zero percent. Thereby, the relative value of $r(n)$ function goes below 0.01 % at five components.

GAMBot-Eva – genetic algorithm based specification evaluation tool

The fifth step of the procedure corresponds to the initial evaluation of specifications of the multi-robot system [20] and is implemented within *GAMBot-Evasoftware*, which stands for Genetic Algorithm based Evaluation of MultiRobot System Specification. The software performs genetic algorithm based heuristic search in wide solution domain. The software uses *JGAP – Java Genetic Algorithms and Genetic Programming Package* [21] as a kernel for genetic processing.

The architectural design of the system includes three concepts: processing module, which executes genetic algorithm and manages evolution of its population; presentation module, which provides the user interface for viewing the processing results; storage module, which ensures persistent data storage and exchange between the first two modules.

The software is designed in a way to allow asynchronous processing of multiple populations of genetic algorithm inspired by island model [22]. The software also provides access to the intermediate results allowing the user to follow the processing of the genetic algorithm online.

Within *JGAP* framework configuration singleton is used to setup the genetic algorithm. The configuration defines the use of technical and low level objects in genetic processing, for instance, breeders, random number generators, event managers. Next, the list of parameters is defined, including population size, allowed variations in population, chromosome pooling options and others. In addition the configuration defines genetic functions, such as natural selection, genetic operators, and sample chromosome, which is used for breeding population. Finally, the fitness function is defined in the configuration.

Additional concepts (see fig. 4) are defined in order to support genetic algorithm processing and to store the results according to the solution design. The project is used as wrapping entity for all other concepts and it stores parameters of genetic algorithm configuration and cost estimation model. The project holds a list of components and their properties which are relevant to the defined mission. The inter-component requirements are defined removing initially invalid solution candidates from processing.The definitions of mission tasks include general parameters, references to mandatory components and their performance indicators.

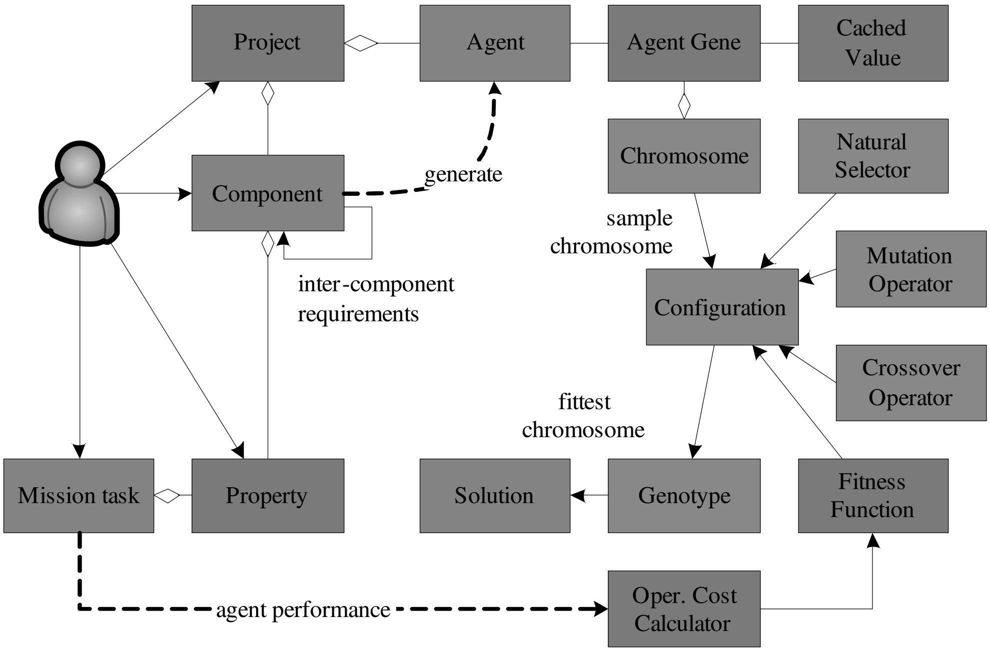

Fig. 4. **Conceptual design of *GAMBot-Eva* processing module**

The presentation module of *GAMBot-Eva* software provides a graphical user interface for monitoring and analysis of asynchronously running evaluation processes. Three different views are defined in the user interface of the module (see Ffig. 5). The process view provides an overview of the running or already finished processes on the computational server. The evolution view provides a graphical representation of the evaluation process and shows the chart of solutions (fittest chromosomes) of particular generation. The solution view displays the details of a particular solution selected in the evolution view, it shows the specification of the multi-robot system.

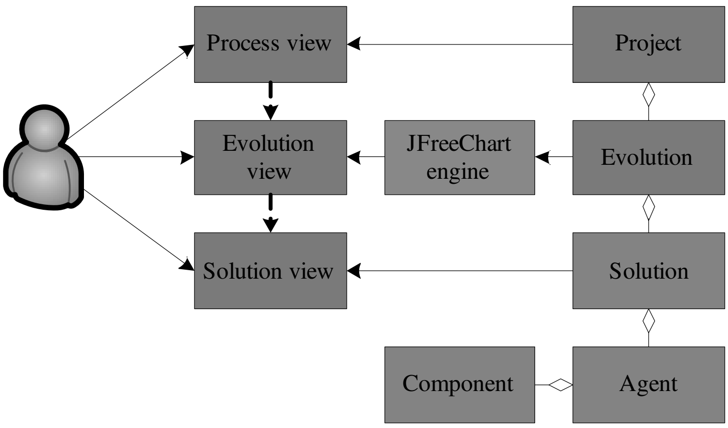

Fig. 5. **Conceptual design of *GAMBot-Eva* presentation module**

GAMBot-Eva software is developed as a tool for the implementation of step 5 of the specification optimization procedure which implies initial evaluation of a specification using heuristic methods. The kernel of the software is an implementation of the genetic algorithm which is a heuristic search approach. The software has modular design and it is the main processing module intended to be executed on the dedicated computational server.

SiMBot-Ctr – control system framework for simulated multi-robot systems

The sixth step of the specification optimization procedure stands for precise simulation based evaluation, which implies model development using the tools provided by simulation environment and execution of simulations to assess the performance of a particular specification of the multi-robot system.

In order to support the development of the control system the author proposes *SiMbot-Ctr* framework, which stands for Simulated Multi-Robot Control System. The framework provides abstract tools for creating control systems based on subsumption architecture [23].

Node is a basic concept, which defines a computational entity of the control system. The list of node types includes behaviours, which define unified action scenario performed by the robot, input and output nodes used to define the interface for the external entities, and signal control nodes used to direct signals through the control system (see Ffig. 6).

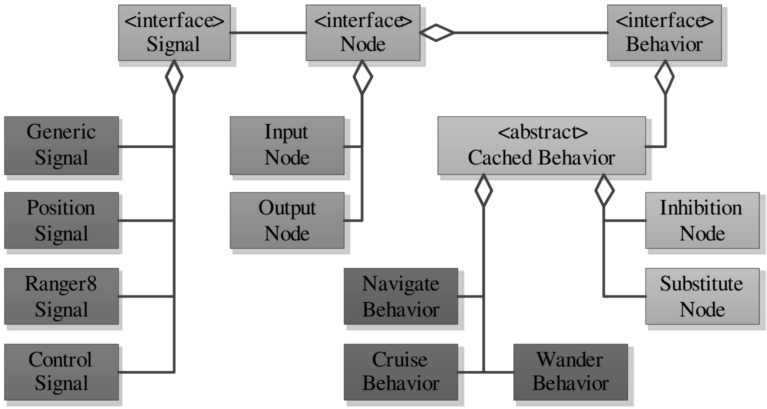

Fig. 6. **Conceptual design of *SiMBot-Ctr* framework**

The framework allows quick and easy setup of a control graph using generic and abstract classes. It provides parameterized implementations of the most common behaviours of robots like cruising, wandering, obstacle avoidance, homing and others.

Execution of the control system implies direct requesting of its outputs. The internal nodes automatically propagate processing requests if their predecessor nodes are not processed. The output of the control system is obtained by a single call, and at the same time only nodes, which are required for obtaining the output, are processed leaving unnecessary lower importance nodes unprocessed.

Conclusion

The paper describes software entities used for implementation of specification optimization procedure for heterogeneous multi-robot systems. Agent combination generation utility *CoMBot-Gen* is used for solution space analysis. Genetic algorithm based tool *GAMBot-Eva* is used for initial evaluation of large number of specification candidates for the multi-robot system. Control system framework *SiMBot-Ctr* is used for controlling the simulated robots during precise evaluation of a relatively small number of specification solution candidates. The software is distributed under GNU General Public License and its source code is available on the project site [24].

The architectural design of the software modules is substantiated by application peculiarities and the specification optimization problem in general. Many features of the software are implemented in order to speed up processing or to improve the precision of the evaluation.

Acknowledgments

Academic study financed by the project "Support for doctoral studies in LUA" /2009/0180/1DP/1.1.2.1.2/09/IPIA/VIAA/017/ agreement Nr. 04.4-08/EF2.D1.05.

References

1. Rao R. V., Patel B. K. Novel method for decision making in the manufacturing environment, Proceedings of the Institution of Mechanical Engineers. Part B. Journal of engineering manufacture, vol. 225, no. 3, pp. 422-434, 2011.
2. Almeida F. L. F. Designing and implementation of an intelligent manufacturing system, Journal of Industrial Engineering and Management, vol. 4, no. 4, pp. 718-745, Dec. 2011.

3. Jammes F., Smit H. Service-Oriented Paradigms in Industrial Automation, IEEE Transactions on Industrial Informatics, vol. 1, no. 1, pp. 62-70, Feb. 2005.
4. Rogozea L., Leasu F., Repanovici A. Ethics, robotics and medicine development, in Proceedings of the 9th WSEAS international conference on Signal processing, robotics and automation, 2010, pp. 264-268.
5. Bajracharya M., Maimone M. W., Helmick D. Autonomy for Mars Rovers: Past, Present, and Future, Computer, vol. 41, no. 12, pp. 44-50, Dec. 2008.
6. Hambling D. Japan sends robots into Fukushima nuclear plant, New Scientist, 2011. [Online] [05.04.2011]. Available: http://www.newscientist.com/blogs/onepercent /2011/03/japanese-send-robots-into-fuku.html.
7. Sullivan J. M. Revolution or evolution? The rise of the UAVs, in International Symposium on Technology and Society, 2005. Weapons and Wires: Prevention and Safety in a Time of Fear., 2005, pp. 94-101.
8. Singer P. W. Wired for War: The Robotics Revolution and Conflict in the 21st Century. Blackstone Audiobooks, 2010.
9. Shen W., Norrie D. H. Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey, 1999. [Online] [19.01.2012]. Available: http://www.eurescom.eu/~public-webspace/p800-series/P815/web/Papers/Agents/abm.htm#sec410..
10. Bi Z. M., Lang S. Y. T., Shen W., Wang L. Reconfigurable manufacturing systems: the state of the art, International Journal of Production Research, vol. 46, no. 4, pp. 967–992, Feb. 2008.
11. Kiener J., Stryk O. Towards cooperation of heterogeneous, autonomous robots: A case study of humanoid and wheeled robots, Robotics and Autonomous Systems, vol. 58, no. 7, pp. 921-929, Jul. 2010.
12. Medvidovic N., Tajalli H., Garcia J., Krka I., Brun Y., Edwards G. Engineering Heterogeneous Robotics Systems: A Software Architecture-Based Approach, Computer, vol. 44, no. 5, pp. 62-71, May 2011.
13. Wong C. Y., Seet G., Sim S. K. Multiple-robot systems for USAR: Key design attributes and deployment issues, International Journal of Advanced Robotic Systems, vol. 8, no. 1, pp. 85-101, 2011.
14. Chatterjee P., Manikrao Athawale V., Chakraborty S. Selection of industrial robots using compromise ranking and outranking methods, Robotics and Computer-Integrated Manufacturing, vol. 26, no. 5, pp. 483-489, Oct. 2010.
15. Devi K. Extension of VIKOR method in intuitionistic fuzzy environment for robot selection, Expert Systems with Applications, vol. 38, no. 11, pp. 14163-14168, May 2011.
16. Lau H. C., Zhang L. Task allocation via multi-agent coalition formation: Taxonomy, algorithms and complexity, in Proceedings of 15th IEEE International Conference on Tools with Artificial Intelligence., 2003, pp. 346-350.
17. Service T. C., Adams J. A. Coalition formation for task allocation: theory and algorithms, Autonomous Agents and Multi-Agent Systems, vol. 22, no. 2, pp. 225-248, Feb. 2010.
18. Komasilovs V., Stalidzans E. Procedure of Specification Optimization of Heterogeneous Robotic System, in IEEE 10th Jubilee International Symposium on Applied Machine Intelligence and Informatics (SAMI), 2012, pp. 259-263.
19. Komasilovs V., Stalidzans E. Functional decomposition method reveals the number of possible specifications of multi-robot system, in 2011 IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI), 2011, pp. 161-165.
20. Komasilovs V., Stalidzans E. Genetic algorithm used for initial evaluation of specification of multi-robot system, in 13th International Carpathian Control Conference, 2012, pp. 313-317.
21. Meffert K., Meseguer J., Martн E. D., Meskauskas A., Vos J., Rotstan N. JGAP - Java Genetic Algorithms and Genetic Programming Package, 2012. [Online] [19.01.2012]. Available: http://jgap.sf.net.
22. Whitley D., Rana S., Heckendorn, R. B. The island model genetic algorithm: On separability, population size and convergence, Journal of Computing and Information Technology, vol. 7, pp. 33-48, 1999.
23. Toal D., Flanagan C., Jones C. Subsumption architecture for the control of robots, IMC-13, no. 1, 1996.
24. 24. Komasilovs, V., Multi-robot system specification optimization tool, Google Project Hosting, 2012. [Online] [14.11.2012]. Available: http://code.google.com/p/mrs-optimization/.