Rafael H. Bordini, Louise A. Dennis and Yves Lespérance (eds)
Pre-proceedings of the 7th International Workshop on Engineering Multi-Agent Systems (EMAS 2019)
13th-14th May 2019, Montreal, Canada
cgi.csc.liv.ac.uk/~lad/emas2019/accepted/EMAS2019_paper_5.pdf

From Goals to Organisations: automated organisation generator for MAS*

Cleber Jorge Amaral[1,2] and
Jomi Fred Hübner[1]

[1] Federal University of Santa Catarina (UFSC), Florianópolis, SC, Brazil
jomi.hubner@ufsc.br
http://pgeas.ufsc.br/en/
[2] Federal Institute of Santa Catarina (IFSC), São José, SC, Brazil
cleber.amaral@ifsc.edu.br
http://www.ifsc.edu.br/

Abstract. The explicit organisational structure allows agent entrants in open multi-agent systems to know their positions in the organisation and cooperate to achieve mutual goals. In spite of its importance, there are few studies on automatic organisation generators that create explicit organisational structures for open systems. This paper introduces *GoOrg*, a proposal for automatically design organisations. Our approach uses as input a goals tree and other features such as necessary skills to achieve the goal, predicted workload and throughput. The output of *GoOrg* is a chart of a well-formed organisation. The generated structure, for instance, can be flatter or taller, accepting matrix connections or not, according to preferences whether there is a need for more coordination levels.

Keywords: Automated organisational design · Organisational chart · Organisation's structure · Multi-Agent Systems

1 Introduction

The organisation structure is a way in which the activities of an organisation are split, organised and coordinated. It allows members to know where they fit relative to others and it reflects authority relations and responsibility for goals, providing a natural way to assign tasks [6]. An organisation structure is a key factor for large-scale Multi-Agent Systems (MAS) and open systems.

Currently, there are a few studies over automation of organisation design process that leads to explicit organisational structures [7, 10]. Although seminal, these works still have limitations to overcome. This paper introduces *GoOrg*, an automated organisation generator that takes the organisational goals tree, looking for opportunities to gather goals into roles giving as output an organisation chart, an explicit organisational structure, according to preferences.

To discuss the problem and to present the proposed generator, this text is structured as follows: Section 2 presents the concept of automatic organisation

* Supported by Petrobras project AG-BR, IFSC and UFSC.

design and the state of art of this research area; Section 3 presents the proposal in details as well as the research method, current results and planned evaluation criteria.

2 Organisational design

Using Pattinson et al's (1987) [9] definition: "Organisation design is the problem of choosing the best organisation class - from a set of class descriptions - given knowledge about the organisation's purpose (goal, task, and constraints on the goal) and the environment in which the organisation is to operate". Given necessary input, an organisational generator is able to give as output organisational aspects, such as, structure, goals definitions, strategy, how leadership will work, which reward system will be used, among others [1].

In the administration area, there are many studies over this matter, including some frameworks that may help companies and other organisations to design their structures [1, 4]. In multi-agent systems, there are what here is being called *manual organisational generators*, i.e., approaches that allow a human to design organisations in a wide variety of structures and other aspects as norms, roles, relations, organisation goals and ontologies (e.g. Moise+ [8] and THOMAS [3]). In spite of having many studies about organisational design, there are still many gaps regarding the wide range of disciplines and high complexity of organisations. Considering only automatic organisation generators, the focus of this research, there are few studies. This paper identified three classes of such generators.

The first class is *automated organisational structure generators*, it is focused "on a specification of desired outcomes and the course of actions for achieving them, analysis of the organisational environment and available resources, allocation of those resources and development of organisational structures and control system" [6]. It processes input such as organisation goals, available agents, resources and performance targets, producing explicit organisations definitions, which may include roles, constraints, assignments of responsibilities, hierarchy and other relations. We found only two works on this class: ODML [7], an algorithm that uses as input mathematical models to predict efforts and create an organisational structure; and, KB-ORG [10] that takes goals and roles to bind agents and create coordination levels. The main drawback of existing studies in this class is the requirement of several parameters including the modelling for each role, reducing its applicability when taking cost-benefit into account.

The second class is the *automated organisational design by task planning.* These planners usually create *problem-driven* organisations, for specific and generally temporal purposes. The organisational structure is not explicit and it usually is a casual result of a task distributing process. Some examples are DOMAP [2] which is a decentralised MAS task planning and Sleight's agent-driven planner [11] using a decentralised Markov Decision Process model. Both models are not suitable for open systems since the tasks are allocated to a particular MAS.

Finally, the third class is *self-organisation generators* [12]. These planners produce emergent organisations which are dynamic, may operate continuously, have overlapping tasks, have no external or central control, and information flows in many directions. The organisational structure is an informal implicit outcome of these bottom-up process. The target of this method is to solve some problem, not exactly to carefully design an organisation [10]. In fact, emergent organisations have advantages over formal structures, which lack the ability to deal with high uncertainty and accommodate individual needs and goals [5]. However, the structure is not carefully designed and not explicit. In addition, in open systems, entries and exits of agents make the system slower due to the necessity of renegotiation.

3 Proposal

The proposal of this research is to develop an *automated organisational structure generator*. This designing class was chosen because: (i) it is suitable to work in open systems; (ii) it helps to develop part of the organisation dimension; and (iii) its outcome may be integrated into other planning techniques, reducing further efforts.

In short, our proposal assigns goals into roles in a structured chart taking advantage of some conditions of the goals such as the ones that have the same parent goal, require the same skills to be performed, have a low predicted workload, etc. Additionally, preferences can also determine whether to combine goals or not, e.g., whether it is preferred a *flatter* or *taller* organisation; if *matrix relations* are allowed or not, maximum *workload* per agent, etc. Moreover, the predicted throughput associated with a goal may inform the need for creation of new hierarchy levels.

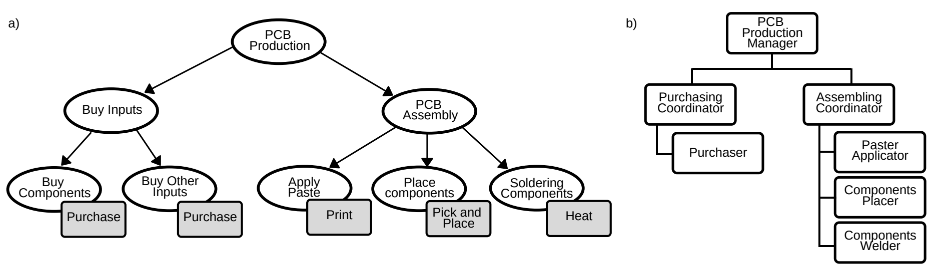

Fig. 1. Automated design for PCB Production. a) Inputs: goals tree and necessary skills. b) Output: the less flat organisation chart

For example, in a goal tree for Printed Circuit Board (PCB) Production it may be necessary to buy inputs and execute the assembling. The *Buy Inputs* goal may have as sub-goals: *Buy Components* and *Buy Other Inputs*. For both sub-goals, the skill *Purchase* could be associated, which means that the agent(s) that will perform both *buy* sub-goals must be able to purchase items. The goal *PCB Assembly* may have three sub-goals: *Apply Paste*, *place Components* and *Soldering Components*. The first is associated with the skill *Print*, the second with the skill *Pick and Place* and the latter with the skill *Heat*.

Fig. 1b shows a possible organisation chart based on the given goals tree configured to be less flat. In this example, the method found that the sub-goals to *Buy Components* and *Buy Paste* could be combined into the same role. One of them was chosen to be the unique child of *Purchasing Coordinator* role. In this sense, the same role will perform both *Components* and *Other Inputs* purchases.

What was discussed until here is the current state of this project (filled shapes of Fig. 2), i.e., using a goal tree associated with skills leads *GoOrg* to create an organisation chart according to preferences (flatter or taller structures). It uses a depth-first search algorithm with cost functions associated with the number of child nodes to check how flat is the structure. The proposal for the next step is to split the process into two phases: the organisational chart design, and the binding phase. With this separation, it is expected to make *GoOrg* suitable to deal with asynchronous changes on the system's resources availability and redesign requests. For instance, with simple changes in the availability of resources, the process can be lighter. However, with bigger changes, for instance on goals tree, a complete redesign process may be necessary.

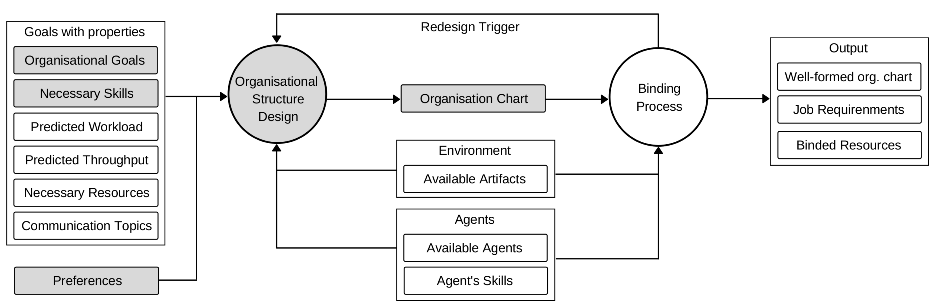

Fig. 2. Goal based Organisation generator (GoOrg).

To enhance the first phase, the next step is to add on each goal the predicted workload, necessary resources, communication topics, and predicted throughput. The expected workload can be used to know how many agents should take the same role or if the same agent can perform more than one role. With communication topics and throughput, the hierarchy levels and departmentalisation can be set. Back to *PCB Production* example, the role *Purchasing Coordinator* would be not necessary, maybe its "child" could be directly associated with the top role *PCB Production Manager*, which would coordinate the processes associated with this role. This can be done according to the number of agents associated with lower roles, and throughput, which affects coordination efforts. Sometimes, instead of reducing the number of levels, the algorithm should increase it, supposing a situation of a large number of agents to be coordinated.

In the second phase, it is proposed to bind resources and roles. The inputs are available agents and skills, available artifacts and organisational design preferences. With available agents and artifacts a binding process can be done, allocating agents and resources to roles. The aim of this binding is to guarantee that the created structure can be well-formed when it runs with the given resources. Finally, the output is an organisational chart with artifacts allocated and agents assigned to roles. The binding process can solve some allocation challenges that does not require a redesign. To illustrate it, consider that *Buy Components* subgoal also needs *Electronics Knowledge* skill, and the chart has created different roles for purchasing, they can be called *Components Purchaser* and *Other Inputs Purchaser*. Consider that *agent A* and *agent B* play, respectively, the referred roles having all the necessary skills to play both. Consider now that *agent A* left the system and *agent C* has joined it, but this agent has no *Electronics Knowledge* skill. The binding process can move *agent B* to *Components Purchaser* role, assigning *agent C* to *Other Inputs Purchaser* role.

3.1 Research method and evaluation

In the adopted research method has 5 steps: (i) bibliography revision under administration basis theory, organisation generators and organisations in multi-agent systems; (ii) defining the problem, the inputs and output; (iii) propose a way to solve the problem; (iv) implement the solution; and, (v) evaluate the solution. The three first steps are done. As a result of the first step, we have found three classes of generators. In the second step, we have positioned our research on *automated organisational structure generator* class. In the third step, we have proposed the use of state space search algorithm to solve this problem.

On the fourth step, we will add to the model new inputs. These data may also allow enhancing the algorithm in the way to decide when a coordination role can be subtracted, maintained or even new ones created. In the sequence, other state space search algorithm and cost functions will be experienced for optimisation purpose and to give more configuration options.

Finally, we will evaluate our solution using existing domains [2, 7, 10]. The goals tree and other aspects for these domains will be manually identified and we will first evaluate the amount of necessary input parameter needed for *GoOrg*. With these inputs, the ability of *GoOrg* to properly design organisations will be evaluated. These situations will be simulated to check if the organisations are able to fulfil the goals, in this sense, the evaluation will be qualitative. The preferences will be varied to evaluate different configurations and their impact on the output.

4 Conclusion

This paper has presented a proposal for an automated generator of organisations based on goals and their properties as inputs. The current status of this research shows that it is feasible to draw an organisational chart only based on organisation's aspects, in other words, it is not necessary to build complex models to use as input to this system. It is intended to enhance the current version adding new input which is expected to bring the necessary information to produce useful organisational charts, taking advantage of opportunities to join goals on the same roles, adding or removing coordination levels.

Besides the organisation chart, an extra outcome of *GoOrg* may be some decentralised task planning input since this study is also expecting to bind agents and roles. The previous allocation of resources is a guarantee that when running this system is able to have a well-formed organisation. About evaluation criteria, it is intended to apply the model in known domains testing if it is able to build suitable structures. These organisations will be simulated in a variety of conditions and tested if goals were fulfilled.

References

1. Burton, R.M., Obel, B., Desanctis, G.: Organizational design: a step-by-step approach (2011). https://doi.org/10.1017/CBO9780511894961
2. Cardoso, R.C., Bordini, R.H.: A modular framework for decentralised multi-agent planning. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems. pp. 1487–1489. São Paulo, Brazil (2017)
3. Criado, N., Argente, E., Botti, V.: THOMAS: An agent platform for supporting normative multi-agent systems. Journal of Logic and Computation **23**(2), 309–333 (2013). https://doi.org/10.1093/logcom/exr025
4. De Pinho Rebouças De Oliveira, D.: Estrutura Organizacional: Uma Abordagem Para Resultados e Competitividade. ATLAS EDITORA (2006)
5. Fink, S., Jenks, R., Willits, R.: Designing and Managing Organizations. Irwin Series in Financial Planning and Insurance, R.D. Irwin (1983)
6. Hatch, M.: Organization Theory: Modern, Symbolic, and Postmodern Perspectives. Oxford University Press (1997)
7. Horling, B., Lesser, V.: Using quantitative models to search for appropriate organizational designs. Autonomous Agents and Multi-Agent Systems **16**(2) (2008)
8. Hübner, J.F., Sichman, J.S.: Organização de sistemas multiagentes. III Jornada de MiniCursos de Inteligência Artificial JAIA03 **8**, 247–296 (2003), http://www.das.ufsc.br/ jomi/pubs/2003/Hubner-jaia2003.pdf
9. Pattison, H.E., Corkill, D.D., Lesser, V.R.: Chapter 3 - instantiating descriptions of organizational structures. In: Huhns, M.N. (ed.) Distributed Artificial Intelligence, pp. 59 – 96 (1987)
10. Sims, M., Corkill, D., Lesser, V.: Automated organization design for multi-agent systems. Autonomous Agents and Multi-Agent Systems **16**(2) (2008)
11. Sleight, J., Durfee, E.H.: Organizational design principles and techniques for decision-theoretic agents. pp. 463–470. AAMAS '13, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2013)
12. Ye, D., Zhang, M., Vasilakos, A.V.: A Survey of Self-organisation Mechanisms in Multi-Agent Systems. IEEE Transactions On SMC: Systems **47**(3) (2016)