Towards Safe Autonomy in Exploration Using Reconfigurable Multi-Robot Systems

Thomas M. Roehr, Ronny Hartanto

DFKI GmbH Robotics Innovation Center, Germany
thomas.roehr@dfki.de
ronny.hartanto@dfki.de

Abstract

This paper presents an approach for applying reconfigurable multi-robot systems to allow for more and safer autonomy in upcoming space missions. It outlines a concept that extends an existing space control architecture relying on the Functional Reference Model for the application of reconfigurable multi-robot systems. The concept involves four main elements: an organization model, a planning and optimization component, an arbiter component to evaluate generated plans and an execution engine. While reconfigurability of the multi-robot system allows for general flexibility the organization modelling intends to support the deliberate exploitation of this flexibility in order to reach mission goals in a safer way.

1 Introduction

A number of space missions have proven the effectiveness of applying robots for planetary exploration. Those missions are usually handicapped by long distances and limited resources on the communication network which making such operations less efficient. One possible approach to overcome this problem is by increasing the level of autonomy of the deployed robotic systems. However, this is only cautiously being accepted as a tool for existing space missions and thus applied in a very limited fashion. Some reasons for these are limited experience with this technology in space missions, development costs, and a low to no risk tolerance in space missions. Meanwhile, the "pressure to reduce manpower during routine mission operation" [1] is real, though thus such robotic missions are far from becoming routine missions and demanding further research on novel mission design concepts.

The main motivation for applying reconfigurable systems is derived from natural systems and targeting an improvement in the areas of safety, efficiency, and efficacy. A multi-robot system represents an organization consisting of multiple actors. This organization can adapt to increase its safety properties, e.g., by maintaining redundancy. Furthermore, actors can specialize for handling specific tasks, since they are part of a larger organization. Thus, actors can focus on performing fewer tasks but with higher precision or efficiency. In addition, specialized actors can cooperate with each other and build a team for providing new capabilities, which is not possible on a single individual actor. This holds even more when the resources can be physically exchanged. We believe that space missions can highly benefit from an application using such a reconfigurable system. However, it demands an effective resource distribution management for allowing an optimal exploitation of the existing limited resources. As a result of this effort an increase in operation safety can be expected. Reconfiguration can only be exploited if the actors are capable of sharing (physical) resources and thus are designed modularly. Resources such as energy modules can be moved within the network of actors, so that an overall increase of safety can be achieved by balancing resources and actively managing the level of redundancy during mission operations. At the same time, the ability of sharing resources and the redundancy increase the flexibility for solving tasks, since it has a larger solution space compared to a monolithic system.

The benefits of a reconfigurable system can be clearly foreseen, however, exploitation of a reconfigurable system comes with a number of challenges such as its application in existing space architectures and handling a much larger state space for the planning tasks. Firstly, to get a realistic chance for a future application, our proposed concept must be compatible with existing space control architectures which embed the Functional Reference Model (FRM) [2],[3].

This paper builds upon the experiences made in [4] and [5] and assumes a similarly capable multi-robot systems to introduce a concept for autonomous operations within safety constraints. [4] describes multi-robot systems for lunar sample collection in unknown environment using heterogeneous robots. However the robots are fixed and not reconfigurable, so that a failure in one system might endangers the overall mission. [5] presents the successful creation of a reconfigurable system, but it did not fully exploit the newly developed hardware capabilities for autonomy. This paper outlines a strategy for improving an existing space control architecture relying on the FRM and presents an approach for applying reconfigurable multi-robot systems to allow for more and safer autonomy in upcoming missions. Firstly, this paper illustrates how a reconfigurable system can be embedded into the control architecture that relies on the FRM and accounting for: (1) multiple robots, (2) recombination of existing ones, and (3) guaranteeing safety for autonomous mission. Secondly, this paper describes our approach towards embedding planning capabilities which exploit the full capabilities of a reconfigurable multi-robot system.

2 A multi-robot architecture based on the Functional Reference Model

The multi-robot architecture discussed in this paper is motivated by an application scenario defined in the project TransTerrA. The project develops and evaluates technology and architecture concepts in order to establish and maintain a logistic chain to perform lunar exploration. The team of robots which is deployed on the lunar surface is capable of reconfiguration similar to the system illustrated in [5], i.e., each robot comprises at least one Electro Mechanical Interface (EMI) which allows to connect to a so-called payload element and also to other robots. A payload element (aka payload-item) is a general purpose container which can host equipment for extending or hardening the multi-robot infrastructure, e.g., by providing additional capabilities for communication, sensing, probing, storing data or energy. Payload elements are modular and exchangeable and extend the functionality of multi-robot systems by providing a standardized interface. This interface enables transport, deployment and exchange across the members of the multi-robot team, and even allows to extend the teams functionality in subsequent deployment phases if needed.

As previously mentioned the mission for the team of robots is defined as planetary exploration of the lunar surface: one scouting robot will be supported by a number of shuttle robots. Furthermore, the logistic chain can increase its outreach by using base-camps – immobile systems serving as hubs to support energy harvesting and payload element exchange. Though base-camps are immobile they comprise an EMI and thus can be relocated using the EMI as link for transport, e.g., by a scouting robot.

The control architecture implementation follows the guidelines of the FRM, cf. Figure 1. This architecture template suggests three distinct layers for mission control, task control and action control. Each layer encapsulates functionality and abstracts or enriches information of lower layers. The FRM provides a clear template, but working with the FRM in [6],[5] has shown some practical challenges, e.g., to design the contents of the bottom

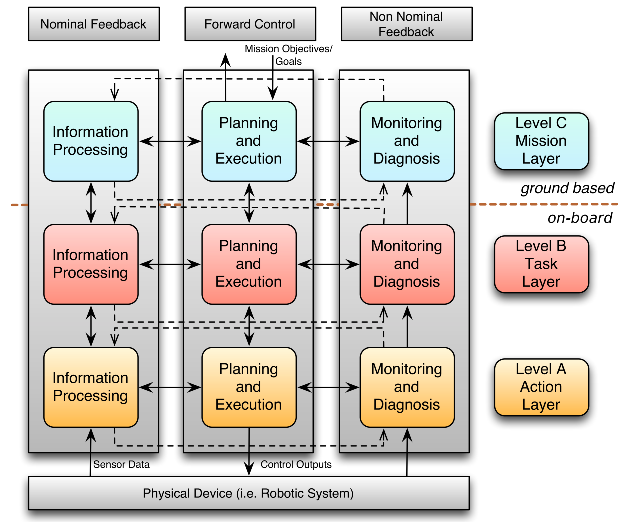

Figure 1. Functional Reference Model

two layers it requests to cluster activities of similar complexity and assign them to one these layers. In practice this separation can look arbitrary for an autonomous system and is rather a heuristic than an exact science. However, this separation can have a great influence on the overall control layout, since it has to be decided along with the Operations Reference Model (ORM) where the actual control has to reside: on-ground or on-board. While the task layer in the mentioned projects has been often been seen an on-board component this settings seems to dictate a centralized control schema for mission and task control. In this context the task control layer can easily be bound to the device that holds the lunar uplink to the groundstation, e.g., a lunar lander.

In general, implementation of control algorithms is straight forward for a centralized control schema. However, centralization means a single point of failure by design. The central control element can be engineered to an increased level of robustness, but the general architecture will remain rigid and prone to the outage of individual members of the multi-robot team. In order to make the mission success less dependent on individual elements of the control architecture, the authors advocate for increasing decentralization instead. This increases the flexibility of missions to adapt to unforeseen events by allowing for more autonomy on the individual robots when needed.

Individual robots represent at least the Level A in the FRM and the concept presented in this paper aims at introducing decentralized control aspects into the task layer while maintaining the existing control approach and properties of the FRM. The FRM does not enforce a purely centralized control on implementations so that this paper presents a concept that embeds distributed control mechanisms. Embedding decentralization aims at two main

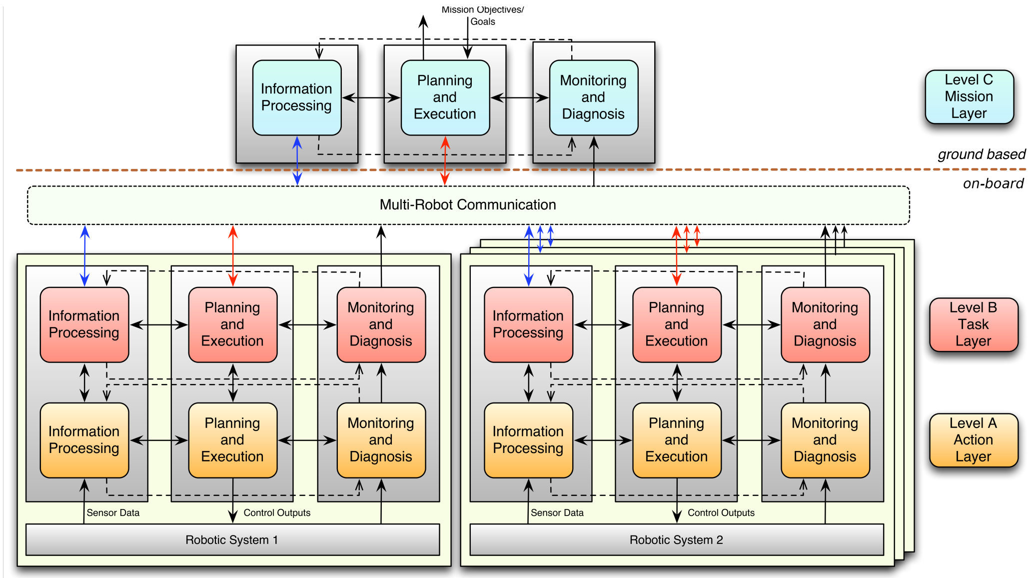

Figure 2. Extending the FRM

goals: 1. allow sub-groups of a deployed robotic team to continue working when individual members fail 2. maintain centralized control by human operators throughout the mission. To account for decentralization, the task layer is therefore extended horizontally while maintaining its capability to interact with Level A and C, cf. Figure 2. This allows to increase the redundancy of the task layer by providing its functionality on different physical devices though maintaining the task layer's capability to operate in a centralized fashion.

The mission control layer acts as the main driver in the whole architecture and human operators can be part of the mission control layer responsible for setting global goals which affect all involved systems. Human operators, likely supported by planning tools, will also deal with the constraints on how a mission has to be performed, i.e., taking resource limitations and scientific goals into account. Planning for a mission can be performed on-ground, i.e. a task sequence can be generated and forwarded to an on-board machine-based controller. Eventually, there is no need for Level A systems to known the overall mission goal - they just need to perform their actions. Information in the FRM flows from top to bottom layers, but account for less intelligent systems at the bottom layers. While this general control schema is practical, it does not allow a direct response in time-critical situations. Ground-to-robot communication is limited and also inter-robot communication can be disrupted or infeasible if the line of sight to a communication relay cannot be maintained; without atmosphere, line of sight is a requirement for communication on the lunar surface. Thus, multi-robot missions can become either very inefficient or arbitrarily complex to plan. To allow for more capable autonomous systems, global information should be accessible for robots that have to expose intelligent behaviour and operate towards the global goal, even when centralized control fails.

In order to dynamically orchestrate the reconfigurable multi-robot system and account for safe execution of the overall exploration mission, an architecture is being developed that relies on the following four main elements: (1) an organization model, (2) a planning and optimization component, (3) an arbiter component to evaluate generated plan, and (4) an execution engine, cf. Figure 3.

This should provide an additional tool for risk mitigation by allowing to modulate redundancies so that resources are shifted to where they provide the greatest benefit for the overall mission.

2.1 Organization modelling

The organization model describes the reconfigurable multi-robot system in a static form and along a number of dimensions (as suggested by [7]). The most important dimensions for the presented application are structure and function. Function depends directly on the existing and active structure of the system, e.g., some functionality will require a mechanical connection of actors and other functionality might require only an established communication link between actors. The organization model serves as basis for the planning and optimization component, and will at runtime be augmented with telemetry data for that purpose. The presented concept uses a resource centric organization model, i.e. every element can be seen as a resource that is part of the organization. Further differentiation from general resources is made between actors,

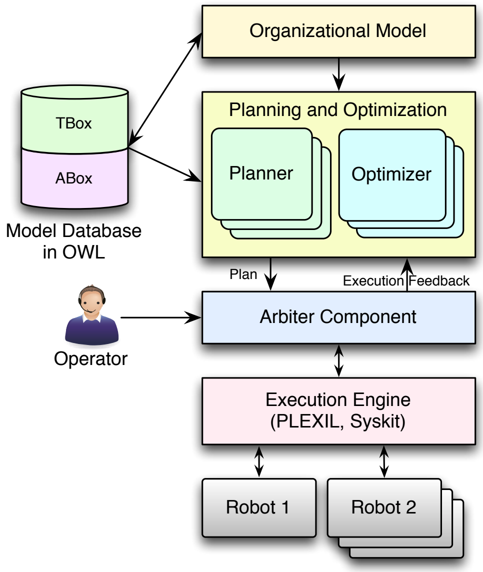

Figure 3. Planning architecture based on an organization model

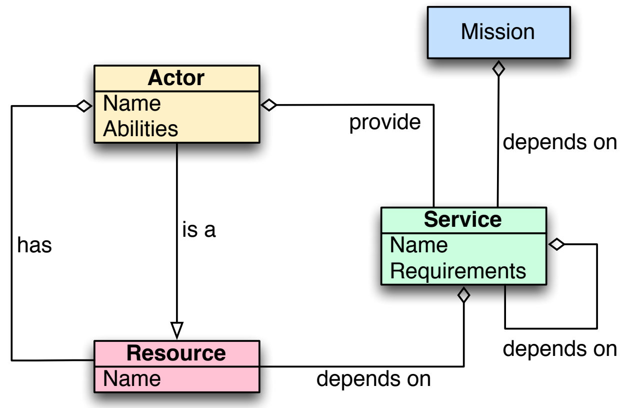

Figure 4. UML diagram of the organization meta model

services, and interfaces to describe the organization structure. Figure 4 illustrates the underlying organization meta-model.

The modelling the reconfiguration of multiple actors into a single actor can be seen as the most significant difference to existing approaches of organization modelling. Functionality of individual actors is inferred from the set of available resources. However, reconfiguration will enable and disable certain actors in the overall organization, i.e. the model needs to partition the overall set actors into an operating set and a dormant set of actors. A previously dormant actor can become operative after joining multiple operative actors – turning the joined actors into dormant actors upon activation.

The model provides a tool to assess an organization's state, e.g., evaluating resource distribution and connection properties. This way the organization model provides an offline tool to analyse feasible recombinations and an online tool to select usable, reachable recombinations.

2.2 Planning with reconfiguration

A reconfigurable multi-robot systems can use any recombination of its actors and modular resource elements to perform actions. Thus, the inferred the set of actor from the organization model serves as partial description of the planning domain. The planning system can account for reconfigurability by relying on a planning domain that takes the information about organization states, the current world model, and mission description into account.

The planning system is responsible for providing feasible plans to execute the overall mission. The overall mission is modelled with the chosen actors, current states of the environment, functional actions of the actors, and the goal set or tasks to achieve. [4] presents how the overall mission is being planned for multi-robot systems by using a Hierarchical Task Network (HTN) planner. The plan can be generated in a centralized fashion either by mission control or by a single actor, e.g., after dynamically assigning an actor to temporarily act as planning instance.

In order to order embed a set of planners, the planning domain is described in the most wide-spread input language for planning, namely the Planning Domain Description Language (PDDL). The planning problem can be represented as a finite-domain planning task as a tuple $\mathcal{P} = (X, A, S_0, G)$. X is a set of variable, where each $x \in X$ is associated with a finite domain $dom(x)$. A is a set of action, where each $a \in A$ takes the form (pre_a, eff_a) with pre_a (the precondition) and eff_a (the effect) each being a partial variable assignment. S_0 is a variable assignment representing the initial state, and G is a partial variable assignment representing the goal [8]. The first two members of the tuple (X, A) are defined in the planning domain.

The result of a planning system is a plan π that consists of a sequence of actions $\pi =< a_1, ..., a_k >$, where $k \geq 0$ representing the length of the plan. By following the π, the state will change from S_0 to G. $k = 0$ means that the plan is empty or no-solution. With no cost assigned to the actions, k denotes the cost of the plan. Additionally, each action a_i can be assigned with a specific cost c_i. As a result, the plan's cost is represented by $C = \sum_k c_i$.

Several planning problems are needed for describing the reconfigurable multi-robot systems that fulfil the required functions for the overall mission. These planning problems are described as $\mathcal{P}_i$, where i is the number of configurations that relevant for the mission from the organization model. The results of the planning system are a pair (π_i, C_i), where π_i and C_i are the sequence of actions and the total cost for the given planning problem in $\mathcal{P}_i$. Hence, computing the reconfigurable multi-robot systems means solving several different planning problems. Based on the results of these planning problems, a degree of safety can be guaranteed by selecting appropriate reconfigurable multi-robot systems.

As already known the complexity of a planning task can become intractable by introducing unused or irrelevant states in the planning problem (cf. [9]). It is important that the planning system can filter the states and consider only the relevant ones for a certain planning problem. To do this, the planning domain generation is supported by a reasoning system that retrieves relevant information from the model database that are represented in Web Ontology Language (OWL). [9] presents an approach for the HTN planning that produces a sound planning problem while maintaining its validity and reducing its search space by including relevant states only.

2.3 A plan arbiter

A set of planners and optimizers will be applied sequentially or in parallel to generate candidate solutions. Each planner can use all actors and resources available to solve the given planning problem, and the planner will use the best selection automatically. The arbiter component selects the best plan from the resulting solution set that fulfils given selection criteria, e.g., a plan that minimizes the use of resources, or a plan that acts under the highest level of redundancy. The selection criteria can be adapted by an operator, e.g., prioritizing on resources, setting policies on action sequences and overall safety levels that rely on organizational metrics.

2.4 An execution engine for a multi-robot architecture

The team of robots executes a selected plan by distributing the corresponding actions to the relevant actors. However, actors have to synchronize their activities whereas the synchronization points are set in the plan, e.g., as preconditions of further actions. Since synchronization is needed between systems, the current status of action execution of a single actor has to be accessible from any other actor.

3 Implementation

This section describes the steps taken towards developing the overall concept architecture. The selected starting point uses an example that is simple, yet illustrates multiple aspects which the overall implementation has to cover.

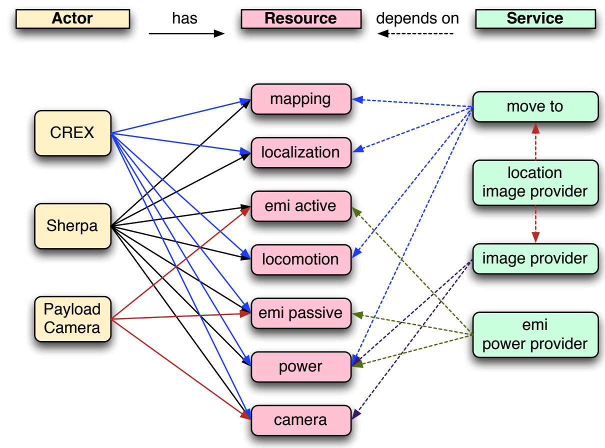

Figure 5. Modelling actor, resource and service relationships

3.1 Scenario

The scenario assumes a mission which depends on taking an image at a given location. The resources available to perform the mission are two single robots and a single payload element. Here, the robot types described in [5] namely Sherpa and CREX will be used; the payload element is hosting a camera. In subsequent work the complexity of the scenario can be increased, e.g., by putting constraints on the viewing angle and the image quality or by limiting the delivery time.

3.2 Organization model

The organization model defines the available set of atomic actors, associated resources, and resource requirements of services. Sherpa and CREX are mobile systems that each have a camera and a power supply. They can serve as location image provider. In contrast, the payload camera is immobile and requires an external power supply to serve as image provider. Each of these actors have an EMI, though CREX is limited to having a passive EMI only. The relationships are illustrated in Figure 5.

Actors are themselves resources that can be bound by other actors, i.e. once an actor is bound it turns into a dormant actor. Given the information of atomic actors all dormant actors of the system are inferred along with the services that such a dormant actor provides. In the given scenario all recombinations with Sherpa and/or CREX can serve as location service provider.

The resource centric view is taken and motivated by an application of semantic web technology. Thus, modelling of the system is done by describing relationships using triples of subject, predicate, and object, e.g., 'CREX has camera'. OWL naturally supports a graph-based modelling approach which is combined with tools for network-analysis [10],[11]. Modelling the organization model in a graph-based structure prepares the next step, e.g., performing percolation [12] to compute advanced metrics for the organization.

3.3 Planning domain and problem

The planning domain is built upon available actors and resources from the organization model, which are Sherpa, CREX, and a payload camera in this scenario. Based on this information, only relevant actions that can be performed by an actor are included in the planning domain. The possible configurations include Sherpa, CREX and payload camera as standalone systems and recombinations of these systems. It needs to be considered that reconfiguration can only be performed at a cost including additional coordination, e.g., Sherpa and CREX can actively move but require a rendezvous point for the reconfiguration process to make Sherpa+CREX operative. As a result, three additional location dependent planning problems are needed, namely at-sherpa-loc, at-crex-loc, and at-rendezvous-loc. The rendezvous location is calculated based on the time and costs from both actors, e.g., movement of CREX is more expensive and slower than that of Sherpa. As a result, the rendezvous location is a location between current location of Sherpa and CREX, where the Sherpa and CREX meet each other at the same time. This information is needed to extend the current planning domain in order to plan with reconfiguration.

Fed with information about the current cost for reconfiguration, the planning problem for solving the overall mission can be modelled. This problem consists of the set of actors including possible transitions to dormant actors and tasks to achieve. To reduce complexity and depending on the chosen safety level, the set of actors can be selectively filtered for the mission planning problem.

3.4 Selecting plans

Selecting plans shall be done according to safety requirements that can be set by an operator. The state of the organization has to be dynamically adapted depending on the current states of actors and resources, e.g., location, energy level and potential wear-out. The current approach focuses on the redundancy level to create a safer organization and neglects additional costs for activating the back, i.e. assuming a low to no-cost switch over when a resource fails and a backup is available. A system comprising CREX, Sherpa and payload camera will provide the highest level of redundancy. However, the setting might prove to be highly inefficient and in a real scenario less capable for some mission, e.g., when moving in extreme terrains. These properties of actors given the current mission are the main input for the plan selection process. When requesting the highest level of redundancy, the selection will neglect the reconfiguration costs but still check the resource availability for achieving the goal. On the other extreme case, the selection is done based on minimizing the use of resources as much as possible.

3.5 Execution engine

Cooperation of the multi-robot system relies on inter-robot communication and an action command protocol that facilitates integration (cf. [5]) of any type of execution engine. In this implementation, the execution engine of an individual system is based on a plan managing component (cf. [13]). This so-called supervision not only allows to perform action execution in local control loops, but also to respond to internal failures. Failure handling strategies can be seen as a normal part of action execution and where an error handling is not available a corresponding error notification will be forwarded via the nominal or asynchronous error channel to trigger error handling a higher levels.

NASA's Plan Execution Interchange Language (PLEXIL) [14] has been selected as high-level controller for executing selected plans on a set of reconfigurable systems. PLEXIL is combined with the mentioned plan management system. Each action on the planning domain corresponds to an action on a target system, i.e., the abstract planning action translates directly into system specific commands. PLEXIL commands are validated against the language syntax and tested on the simulator for ensuring the reliability of the plan. PLEXIL can access the current state of the environment using the inter-robot communication and forwarding this state through PLEXIL's variables. Finally, PLEXIL coordinates the global plan while the supervision manages the actual execution of individual actors including a monitoring of the action execution process.

4 Background

A number of reconfigurable robotic systems exists. However, the majority of these systems focuses on studying cooperation using rather simple robots which can perform low-complexity task. In addition, reconfigurable robots that perform complex tasks are mostly limited to single systems. Reconfiguration of complete robotic systems and cooperation leads to the more general idea of organization modelling and organization management. Organization modelling is not a novel idea and various approaches have been developed (cf. [15]). Organization modelling approaches cover a range from highly theoretical approaches such as [16] to practical implementations [7],[17]. While reconfiguration has been recognized as a key element for organizations, it has not yet been essential to the organization models available. However, this will be a requirement to fully exploit the flexibility of a reconfigurable system.

Both [17] and [7] focus on an application in the robotics domain and provide means to model reconfiguration in terms of group and team building. [7] considers three dimensions of modelling: structural, functional and deontic. Whereas the structural dimension deals with the specification of relationship relying on roles and group, the functional dimension allow to embed organizational goals and missions that can contribute to reaching a goals. The deontic dimension allows to specific (socially) permissive activities of individual members of the team. [17] provides a more agent centric approach and embeds a quantification of agents' capabilities. Thus, it allows to find the optimize agent to role mapping given a goal of the organization. [17] considers reconfiguration of the agents and teams and also illustrates the application of metrics in order to analyse a system's flexibility [17].

The mentioned research lays ground for developing and applying an organisation model in the context of a system as illustrated in [5]. However, reconfiguration has a greater effect in that context, since existing entities are temporary given up to allow for the existence of another. Though, this feature can also be derived from team building, none of the model seen so far takes this feature seriously into consideration, i.e. a recursive definition of agents which also allows for a set of temporary or rather virtual agents. A recursive definition of agents allows to form agents by merging two or more other agents and allows to activate a different set of capabilities while disabling a subset of capabilities provided by former individual actor that are now merged into the newly formed one.

Other neighbouring research domains to the problems illustrated in this paper can be found in risk assessment when engineering and resilience research for ecological and economic systems [18]. Research results in these neighbouring fields support the idea that a holistic view needs to be taken and a generic measure has to be identified to allow a true monitoring and description of state of a complex systems. Only broadly applied dense monitoring of systems allows for appropriate responses that make the systems more resilient to unforeseen impacts [19]. Furthermore, "[s]leeping functional groups" [20] have been identified as a natural phenomenon which can be found in individual actors of a system to increase resilience, giving further motivation to maintaining a set of dormant actors in a multi-robot team.

5 Conclusions and Future Work

This paper presents a general concept aligned with an existing space architecture to exploit a reconfigurable multi-robot team using an organizational model at its core. This paper does not provide the details of a full implementation, but outlines the set of critical architecture elements from the authors' point of view.

In the long run, this research aims at identifying organization properties that allow to identify valid and feasible metrics to quantify the current capabilities of a multi-robot organization to cope with outages of individual components. After identification of significant and vital safety metrics the multi-robot organization can be dynamically adapted to maintain a state of maximum safety relying on its capability of reconfiguration. The results might lead to designing simpler and cheaper systems using a modular approach, while still allowing for execution of complex tasks. However, reconfiguration requires complex and costly cooperative maneuver and this aspect needs to be taken into consideration in the planning part as a next step.

5.1 Acknowledgment

The presented work is part of the project TransTerrA – a project supported by the German Space Agency (DLR) and conducted under grant agreement 50RA1201 (BMWi).

References

[1] European Cooperation for Space Standardization, "Space engineering spacecraft on-board control procedures," August 2010, retrieved 6 August 2010, from http://www.ecss.nl.

[2] P. Putz and A. Elfving, "Control techniques 2," Dornier GmbH, Tech. Rep. CT2/CDR/DO/BL, August 1991, issue: 1.1, ESTEC Contract 9292/90/NL/JG (SC).

[3] G. Visentin, "Autonomy in ESA planetary robotics missions," PPARC KITE Club/Sensors KTN. Workshop on Autonomous Sensors, 2007, retrieved March 8 2013, from http://www.scitech.ac.uk/Resources/PDF/GianfrancoVisentin.pdf.

[4] M. Eich, R. Hartanto, S. Kasperski, S. Natarajan, and J. Wollenberg, "Towards coordinated multirobot missions for lunar sample collection in an unknown environment," *Journal of Field Robotics*, vol. 31, no. 1, pp. 35–74, 2014. [Online]. Available: http://dx.doi.org/10.1002/rob.21491

[5] T. M. Roehr, F. Cordes, and F. Kirchner, "Reconfigurable integrated multirobot exploration system (rimres): Heterogeneous modular reconfigurable robots for space exploration," *Journal of Field Robotics*, vol. 31, no. 1, pp. 3–34, 2014. [Online]. Available: http://dx.doi.org/10.1002/rob.21477

[6] F. Cordes, I. Ahrns, S. Bartsch, T. Birnschein, A. Dettmann, S. Estable, S. Haase, J. Hilljegerdes, D. Koebel, S. Planthaber, T. Roehr, M. Scheper, and

F. Kirchner, "{LUNARES}: lunar crater exploration with heterogeneous multi robot systems," *Intelligent Service Robotics*, vol. 4, no. 1, pp. 61–89, 2010. [Online]. Available: http://dx.doi.org/10.1007/s11370-010-0081-4

[7] J. F. Hübner, J. S. a. Sichman, and O. Boissier, "MOISE+," New York, New York, USA, p. 501, 2002. [Online]. Available: http://doi.acm.org/10.1145/544741.544858http://portal.acm.org/citation.cfm?doid=544741.544858

[8] J. Hoffman, I. Weber, and F. M. Kraft, "Sap speaks pddl: Exploiting a software-engineering model for planning in business process management," *CoRR*, vol. abs/1401.5858, 2014.

[9] R. Hartanto, *A Hybrid Deliberative Layer for Robotic Agents: Fusing DL Reasoning with HTN Planning in Autonomous Robots*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer-Verlag, 2011, vol. 6798.

[10] B. DezsÅ, A. Jüttner, and P. Kovács, "LEMON an Open Source C++ Graph Template Library," *Electronic Notes in Theoretical Computer Science*, vol. 264, no. 5, pp. 23–45, July 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1571066111000740

[11] D. A. Bader and K. Madduri, "SNAP, Small-world Network Analysis and Partitioning: An open-source parallel graph framework for the exploration of large-scale networks," in *2008 IEEE International Symposium on Parallel and Distributed Processing*. IEEE, Apr. 2008, pp. 1–12. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4536261http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4536261

[12] M. Newman, *Networks*. Oxford University Press, Mar. 2010. [Online]. Available: http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780199206650.001.0001/acprof-9780199206650

[13] S. Joyeux, F. Kirchner, and S. Lacroix, "Managing plans: Integrating deliberation and reactive execution schemes," *Robotics and Autonomous Systems*, vol. 58, no. 9, pp. 1057–1066, 2010. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0921889010001090

[14] G. Dowek, C. Muñoz, and C. Rocha, "Rewriting logic semantics of a plan execution language," *Electronic Proceedings in Theoretical Computer Science*, vol. 18, pp. 77–91, 2010.

[15] V. Dignum, Ed., *Handbook of Research on Multi-Agent Systems: Semantics and Dynamics of Organizational Models*. IGI Global, 2009. [Online]. Available: http://books.google.de/books?id=-UMAEGsMvfYC

[16] B.-J. Putten, V. Dignum, M. Sierhuis, and S. Wolfe, "OperA and Brahms: A Symphony?" in *Agent-Oriented Software Engineering IX*, ser. Lecture Notes in Computer Science, M. Luck and J. Gomez-Sanz, Eds. Springer Berlin Heidelberg, 2009, vol. 5386, pp. 257–271. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01338-6_19

[17] S. DeLoach and V. Kolesnikov, "Using Design Metrics for Predicting System Flexibility," in *Fundamental Approaches to Software . . .*, L. Baresi and R. Heckel, Eds. Springer Berlin Heidelberg, 2006, vol. 3922, no. 0347545, pp. 184–198. [Online]. Available: http://link.springer.com/chapter/10.1007/11693017_15http://people.cis.ksu.edu/~sdeloach/publications/Conference/fase06-final.pdf

[18] F. Berkes and C. Folke, Eds., *Linking social and ecological systems: management and practices and social mechanisms*. Cambridge University Press, 2000. [Online]. Available: http://lib.icimod.org/record/16819/files/JO130.pdf

[19] G. B. Giannakis, V. Kekatos, N. Gatsis, S.-J. Kim, H. Zhu, and B. F. Wollenberg, "Monitoring and Optimization for Power Grids: A Signal Processing Perspective," *IEEE Signal Processing Magazine*, vol. 30, no. 5, pp. 107–128, Sept. 2013. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6582699

[20] D. R. Bellwood, T. P. Hughes, and A. S. Hoey, "Sleeping functional group drives coral-reef recovery." *Current biology : CB*, vol. 16, no. 24, pp. 2434–9, Dec. 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960982206023670http://www.ncbi.nlm.nih.gov/pubmed/17174918