Enabling World Information Exchange in a Heterogeneous Team of Robots for Mobile Manipulation

Ryo Sakagami[1], Martin J. Schuster[1], Riccardo Giubilato[1], Alin Albu-Schäffer[1], Freek Stulp[1], and Andreas Dömel[1]

***Abstract*— Exchanging information about the world among robots is fundamental for them to cooperate. However, challenges still persist when the robots exhibit heterogeneity in their hardware/software components, and different tasks such as observing and interacting with the environment are given to them. In this paper, we propose a new communication architecture and describe how it addressed these challenges with our team of heterogeneous robots developed for and demonstrated during the four-week Moon-analogue exploration mission on Mt. Etna, Italy. Our approach is to make each robot have its egocentric world model and exchange information only by remotely accessing the interfaces of other robots' world model. We offer key findings on the different communication strategies to trigger the remote interfaces as well as insights into how these information exchanges could compose a complex conversation. Through in-depth analyses of the real-world demonstration, we evaluate the contribution of such a communication design, and derive general design guidelines for information exchange about the world between robots in heterogeneous multi-robot systems.**

***Index Terms*— Multi-robot cooperation/collaboration; Intelligent and autonomous space robotics systems; Planetary exploration**

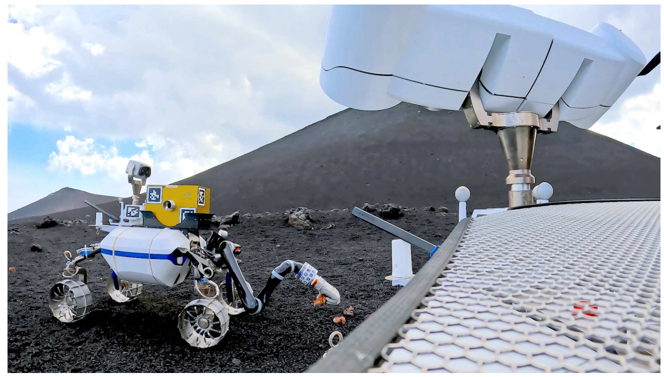

Fig. 1: Impression of an experiment on multi-robot information exchange with our two Lightweight Rover Units (LRUs) on Mt. Etna, Italy. LRU1 (in the front) selects one of the stones via its science camera and tells LRU2 (in the back) to collect it with its manipulator.

I. Introduction

Heterogeneous teams of robots have advantages regarding efficiency, robustness, and versatility over a single robotic system. Possibility of parallel task execution improves *efficiency*, while overlapping and complementary capabilities increase *robustness* and *versatility*, respectively. Therefore, utilization of different, heterogeneous robots as a team is expected to be useful especially for scientific exploration in harsh environments, e. g., extraterrestrial surfaces.

We have been developing such a robotic team and demonstrated it during the four-week ARCHES mission in a Moon-analogue environment of Mt. Etna [1], [2]. The team, consisting of two lightweight rover units (LRUs: see Fig. 1), a flying robot Ardea, a lander, and various payload boxes, conducted autonomous sample collection and deployment of an antenna array. To distinguish different types of stones and terrains, LRU1 is equipped with a science camera with spectral filter wheels. LRU2, on the other hand, is equipped with a 6-DOF manipulator to fullfil manipulation tasks.

What is crucial for fulfilling such missions is how to exchange information about the world among such a heterogeneous team of robots having different tasks. In this paper, we propose a world knowledge exchange concept, achieving the following three design goals. First, we aim to support heterogeneity in the world model[1] [3] used by robots, to allow for compact, efficient, task-driven representation. Due to the differences in hardware/software and requirements for the tasks, each robot requires a different representation of the world, which is difficult to be simply synchronized.

Second, we target to enable efficient low-bandwidth communication on a need-to-know basis. Even if feasible at all, synchronizing the entire state would be inefficient, since not all information is relevant for every robot.

Third, we aim to allow robots to have inconsistencies between their world models and intend to delegate to each robot how to integrate information from other robots. Robots cannot model the world without having system-specific errors such as calibration and sensor errors. Thus, a merged, single world state for the entire team would not be useful especially for manipulation, which requires high precision.

Our key idea is to make each robot has its own world model suitable for its capabilities and tasks, and let the robots exchange information by remotely accessing the interfaces of the others' world models. We describe how we developed our team of robots with this concept during the ARCHES campaign, and offer key findings and insights from our devel-

This work was supported by the Helmholtz Association project iFOODis (contract number KA2-HSC-06) and by the European Union's Horizon Europe research and innovation framework project IntelliMan (grant agreement No. 101070136).

[1] DLR (German Aerospace Center), Institute of Robotics and Mechatronics, Münchner Str. 20, 82234 Wessling, Germany, {firstname.lastname@dlr.de}.

[1]In this paper, we use the term from a robotics perspective [3]. A world model is an internal representation that reflects relevant parts of the physical world. It is shared by multiple sensors, planners, and actors. It is segregated by a boundary and internally consists of a state and operations.

 DOI: 10.1109/iSpaRo66239.2025.11436782

opment, demonstration, and evaluation, which are applicable to multi-robot systems in general.

The rest of the paper is structured as follows. We first provide an overview of the related work and elaborate the challenge we focus on in Section II. The approach we developed is described in Section III. The concrete implementation and demonstration with the real systems is described in Section IV, and the evaluation and lessons learned are described in Section V. We conclude with Section VII.

II. Related Work

There have been many works around the theme of utilization/collaboration of heterogeneous robots as a team. We particularly review those describing communication framework/design among robots for collaboration.

RoboCup [4] has been fostering technologies for world knowledge exchange among robots. A team could consist of heterogeneous robots from different research institutes or laboratories, and thus a framework to enable knowledge exchange among robots has been essential. Some early work used a global world model on an off-field computer and let every robot access it [5], [6]. Later, approaches to let each robot have its local, egocentric world model and broadcast relevant information were proposed [7], [8]. Although they addressed world knowledge exchange among heterogeneous robots, 1) their systems are homogeneous in tasks and thus 2) each robot always broadcast information in the same format.

The same statement applies to the multi-robot SLAM research, which has been addressing the challenge of mapping and localization among multiple, heterogeneous robots [9], [10], [11], [12]. Each team member has the common task of mapping and localization, assumes to have a common world model sharing coordinate frames and uncertainty representations, and exchanges information in a common format. Such a distributed, synchronized world model approach is also employed by NASA's upcoming Cooperative Autonomous Distributed Exploration Rovers (CADRE) mission [13].

Another research direction is to use a commonly-shared large database, e. g., an ontology as a world model for multiple, heterogeneous robots. For instance, RoboEarth [14] enables different robots to share semantic maps, scene geometry obtained by SLAM, and object models relevant for their tasks. This approach requires each robot to employ a single, synchronized state with the same world model, and thus would not be applicable to our systems due to the heterogeneity in the systems and tasks.

From an application-oriented perspective, there are many notable system works in various environments. The euRobin project demonstrated the parcel hand-over among eight heterogeneous robots from three different domains [15]. The Co4Robots demonstrated logistics tasks with a heterogeneous team of three decentralized systems [16]. The Surface Avatar missions showed successful teleoperation of four heterogeneous robots on Earth by an astronaut from the international space station [17]. They achieved complex collaboration among completely different robots, although no concrete communication framework/design about the world was proposed. Our research focus is to identify an effective world knowledge exchange pattern in such a team of robots where each robot needs to have different world model due to the task and capability heterogeneity, as was the case in our ARCHES demonstration mission.

III. Remote Access of World Models

Each robot does not exactly know what the others know about the world and in which format they store such knowledge. This was the challenge that our team of robots was confronted with. In this section, we describe our approach to address this, which is to enable each robot to access the others' world models remotely (see Fig. 2).

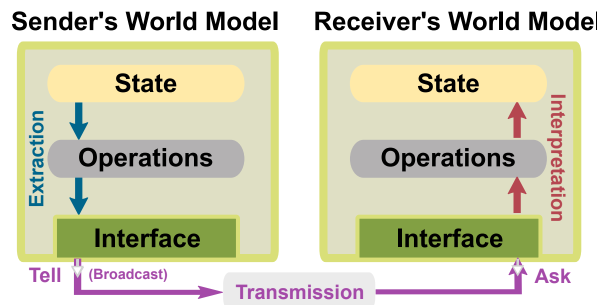

Fig. 2: Graphical representation of our approach to achieve remote access of world models.

A. Exposure of Interfaces

A world model provides interfaces through which other components can provide/acquire information to/from the world state [3]. The core idea of our approach is to expose these interfaces to the external robots as well.

This addresses a part of the challenge that each robot might have totally different world state structures. Operations inside the others' world models bridge the interfaces to the state with extraction and interpretation (see Fig. 2). Thus, knowledge about the state structure of other robots becomes unnecessary for communication by exposing the interfaces.

B. Communication Patterns: Tell, Ask, and Broadcast

The exposed interfaces enable robots to convey/query information to/from each other. However, when to convey/query what information to/from which robots is dependent on situations. We identified the following three communication patterns regarding how to trigger the interfaces.

a) Tell: This type of communication is initiated by the information sender. This pattern is used when a robot considers that a certain piece of information in its world model could be relevant for another robot as well. For instance, when a robot picks/places an object, the robot should tell the new object location to other robots which further manipulate/transport/localize it.

b) Ask: This type of communication is initiated by the information receiver. This pattern is used when a robot knows that others could have more certain information about a specific aspect of the world. For instance, when a robot searches for an object, the robot should ask the latest state of the object to other robots which are physically conveying it or have manipulated it recently.

c) Broadcast: This is a special type of the "tell". This pattern is used only for exceptional cases where a certain piece of information is known to be commonly relevant for the whole team. For instance, in the application scenario of RoboCup [4], the pose of the ball should be broadcasted by the robot which possesses the ball.

C. Towards Complex Conversation

Single, one-directional communication of the tell, ask, and broadcast still faces the remaining part of the challenge that each robot does not exactly know what the others know. Telling fails if the receiver robot cannot interpret the provided information. Similarly, asking fails if the sender robot does not store corresponding information and could not extract requested information. This is because robots can tell/ask/broadcast only based on assumptions on what others would know about the world.

The key idea to address this remaining challenge is to combine multiple tell, ask, and broadcast communications and to make the information exchange both-directional conversation. This is analogous to how humans have conversation; each person has its own understanding of the world and no one exactly knows what others know. Assuming what others know, they tell/ask/broadcast information, and if it is not understood, they initiate a conversation to convert the information into an understandable format.

Although how robots can generally achieve such conversation is beyond the scope of this paper, we would provide several concrete implementation examples in Section IV.

IV. Demonstration with Robotic Systems

In this section, we elaborate concrete challenges that our robotic team was confronted with for fulfilling the ARCHES mission, and how they were overcome by our approach described in Section III.

A. The ARCHES Moon-Analogue Exploration Mission

In the ARCHES demo mission, the team of our heterogeneous robots cooperated semi-autonomously and collected sand and stone samples from unvisited locations [2], performed laser-induced spectroscopy (LIBS) analyses on rocks [18], and deployed payload boxes to construct the low-frequency antenna array (LOFAR) [19] (see Fig. 3). Instruments and sample containers are housed in a standardized payload box module, which LRU2 can manipulate and transport and LRU1 can detect as a SLAM landmark.

To fulfill these tasks autonomously and collaboratively, the world knowledge exchange among the heterogeneous robots played a crucial role. Particularly, LRU2 needed to know from which lander's storage it could pick a certain payload box, and LRU1 needed to distinguish which particular box was placed on the terrain and served as a fixed landmark.

B. Implementation of Exposed Interfaces

World states of different robots show substantial and topological heterogeneity due to difference in their components and tasks. Nevertheless, all of our robots use the AIMM-WM (Autonomous Intelligent Mobile Manipulator's World Model) [20] as a software component to represent the world. AIMM-WM is a hybrid symbolic/geometric tree-based world model, where nodes represent particular entities such as concrete objects (e. g., boxes) and abstract objects (e. g., approaches and grasps), while relations represent a 6-DOF transformation between nodes (see Fig. 5 for example). The interfaces of AIMM-WM are implemented as ROS services, and thus exposing them to other robots is achieved by using ROS multi-master setup[2]. Fundamental communications are enabled by these exposed interfaces, whose examples are shown in Table I.

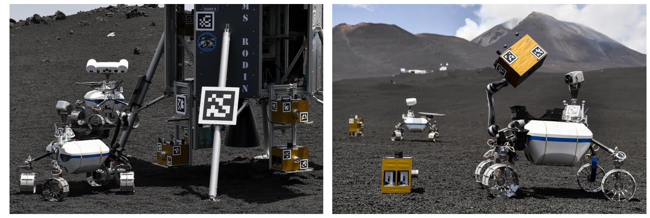

Fig. 3: Snapshots of the low-frequency antenna array deployment mission on Mt. Etna during the ARCHES campaign. LRU1 and LRU2 started at the lander, which serves as a global landmark as well as payload box storages (left). LRU1 navigated around the deployed boxes to improve their localization whereas LRU2 carried and deployed the antenna boxes (right).

TABLE I: Examples of fundamental communication enabled by the exposed ROS service interfaces of AIMM-WM.

Communication Examples	Type	Used ROS Service
Do you know an object called x?	Ask	`get_object_id`
What is inside of x?	Ask	`query_pair_ids`
What is on top of x?	Ask	`query_pair_ids`
In which scene is x located?	Ask	`query_pair_ids`
Where is x from y?	Ask	`get_pose_from_object_to_object`
How heavy is x?	Ask	`get_property`
There is a new object called x.	Tell	`add_item`
The object x is in/on another object y.	Tell	`reassign_object`
The object x is moved to a pose p.	Tell	`update_pose`

We had three specific use cases for having complex conversation. They are implemented as RAFCON state machines [21], combining the above fundamental communication as building blocks. Their concrete implementations are described in the following sections.

C. LRU2 Asks Lander Storage Status

LRU2 is responsible for transporting the modularized payload boxes from the lander to collect samples, to conduct LIBS measurements, and to deploy LOFAR boxes in a certain configuration. LRU2 picks a suitable payload box from one of the six storages of the lander, and the lander itself loads a box from its internal room. To enable LRU2 to know which payload box is placed on which storage, the complex conversation shown in Fig. 4 is implemented.

[2] `https://fkie.github.io/multimaster_fkie/`

The conversation consists of two parts: the first part to ask the lander, and the second part to interpret the provided information and update its own world model. In the first part, using the `query_pair_ids`, LRU2 asks the lander which payload box is located in which storage and retrieve the result as a set.

In the second part, LRU2 iterates on this set and executes three different operations on its world model depending on the acquired information. If a storage should be empty according to the lander, LRU2 deletes any existing payload box from the storage in LRU2's world model. If the storage should have a payload box, LRU2 either 1) re-assigns it to the storage node if the instance of the box already exists, or 2) creates a new instance of the box as a child of the storage if there is no instance of the box yet.

D. LRU2 Broadcasts Payload Status

The payload boxes serve as a landmark for the multi-robot SLAM component [22], [23]. However, they are not placed at fixed places but rather manipulated and transported by LRU2. Therefore, it is essential that LRU2 tells all the other robots the status of the payload boxes after manipulation.

Manipulation affects the physical contact status, and how such status is represented is different from robots. For instance, when LRU2 picks a box, it is represented to be attached at the end effector in the LRU2's world model. However, other robots do not necessarily model the end effector of LRU2, since that is irrelevant for the operation of other robots. Instead, other robots at least model LRU2 itself in a simple form. Therefore, the first step of the conversation is for LRU2 to ask other robots which objects they know in common. By asking other robots iteratively, LRU2 identifies which object is appropriate for a box to be referenced. Afterwards, LRU2 tells others to update the information of the box or to instanciate one if not known yet.

E. LRU2 Asks LRU1 Stones to Collect

LRU1 is capable of distinguishing different types of stones using the science camera. Therefore, the third use case is that LRU2 asks LRU1 which stones are to be collected. First, LRU2 asks LRU1 if it knows any stone. If so, LRU2 further asks where the stone is located. Based on this, LRU2 drives to the location and localize the stone by using the perception component. The pose estimated by LRU2 itself is stored into the world model of LRU2.

V. Evaluation

In this section, we analyze our systems described in Section IV and evaluate the following three advantages that our approach brought: 1) to enable robots to have heterogeneous world models, 2) to reduce unnecessary information exchange among robots, and 3) to allow robots to have world model inconsistencies between each other. The first two are evaluated by using the log of the AIMM-WM states and their modification during more than six hours from the ARCHES demo missions. The third one is evaluated by conducting follow-up sample-collection experiments in our laboratory[3].

A. Evaluation 1: Heterogeneity in World Models

Our approach works irrespective of world models employed by different robots. To evaluate this point, we analyzed the world state after the mission to deploy the low-frequency antenna array was successfully completed. As show in Fig. 3, this mission included a complex manipulation task for LRU2 to stack the payload box with a radio antenna onto the other box with a power-supply capability. In parallel, LRU1 was given a task to improve the localization of the deployed boxes utilizing the fiducial markers on them. Such diversity of the tasks given to the different robots imposed the necessity of the heterogeneous world state per robot.

Each world state after the mission is shown in Fig. 5. One prominent heterogeneity exists in the number and the kind of objects to be represented. While some objects, e. g., the lander and the manipulated payload boxes, were represented in the world state of every robot, others were not. For instance, grasps and approaches for manipulation were only represented in LRU2's word state. We counted the number of object nodes in the world state of each robot and how many were also represented in the others'. As shown in Fig. 6, 70 nodes were commonly represented by every robot, whereas 202 nodes were solely represented in LRU2's world model.

We also qualitatively show how different states have different topology after the aforementioned mission. The lower, magnified section of Fig. 5 shows the representation of the stacked reference and power payload boxes. In the LRU2's state, these boxes are represented as a chain of parent-child relationship, conveying the semantics that these objects have physical dependencies. This is particularly important for LRU2 as a mobile manipulator, since it must be able to plan to manipulate the upper object first before manipulating the lower one. On the other hand, in the world state of the lander and LRU1, these boxes are represented as direct children of the lander (named "rodin" in the figure) itself, since these two robots are not interested in such physical dependencies and thus such simplified representation is sufficient.

B. Evaluation 2: Exchanged Data Reduction

Our communication approach is to use the other robots' world model interfaces and exchange only relevant information at a relevant moment. This approach also improves the communication efficiency.

To evaluate this point, we first counted the total number of interface service calls to the world model of each robot (including those within a robot) from the log file, which is shown in Table II. The number of reading from and writing to the world model for the robot i is represented as r_i and w_i, respectively.

Next, we checked how many of such service calls were triggered from one robot to the others, which was in total

[3]The approach was demonstrated on Mt. Etna as an additional experiment after the main demo missions as shown in Fig. 1. However, the appropriate log for the evaluation was missing due to technical errors.

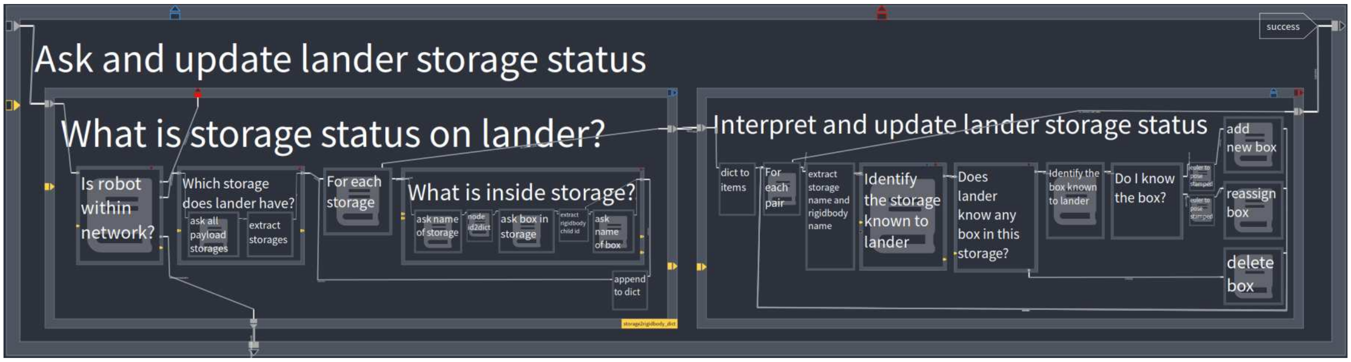

Fig. 4: Screenshot of RAFCON state machine to ask which payload box is placed on which lander's storage, and to interpret and integrate this information into its own world model.

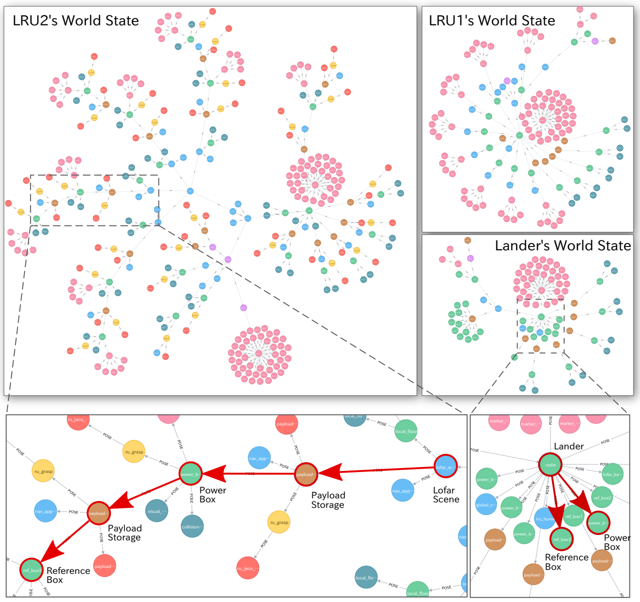

Fig. 5: World state of each robot after the mission to deploy the low-frequency antenna array. Topological heterogeneity is highlighted in the magnified section below.

347 (we name this number rw_{ext}). This is the number of knowledge exchange needed by our approach.

TABLE II: Number of read/write service calls to the world model.

Robot	# of Reading	# of Writing
Lander	26 (r_{lander})	16 (w_{lander})
LRU1	1298 (r_{lru1})	57 (w_{lru1})
LRU2	3460 (r_{lru2})	494 (w_{lru2})

To compare our approach with a baseline, we considered two other possible communication approaches shown in Fig. 7: 1) having a single, central world model at the lander [5], [6], and 2) having a synchronized, distributed world model per robot [7], [8].

With the baseline 1, the two rovers would have always provided/requested information from the lander's world model. The number of dataflow can be reasonably estimated by

$$\sum_{i \in \text{Robots} \setminus \{\text{lander}\}} (r_i + w_i) - rw_{\text{ext}},$$

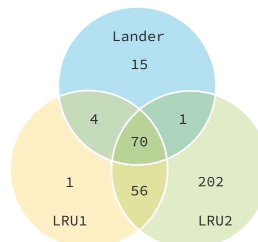

Fig. 6: The number of object nodes represented in each robot's world state after the antenna deployment mission.

since every reading and writing to the world model of LRU1 and LRU2 would have caused the communication to the lander. The value was computed to be 4962. Similarly, the number when the synchronized, distributed world model were used (the baseline 2) can be computed by

$$(|\text{Robots}| - 1) \sum_{i \in \text{Robots}} w_i - rw_{\text{ext}},$$

since every writing into each robot's world model has to be synchronized to the other two robots. The value was computed to be 1134.

The results are summarized in Table III. Overall, the efficiency was improved to be a factor of 14.10 and 3.26 in comparison to the baseline 1 and 2, respectively.

TABLE III: Comparison of the number of world knowledge exchange among robots using the single world model approach (baselines) and our proposed one.

Baseline 1: Single, Central WM	Baseline 2: Synchronized, Distributed WM	Our Approach
4892	1134	347

C. Evaluation 3: Allowing Inconsistent World States

Due to imperfect calibration and sensor errors, robots cannot model the world without introducing errors, which are system-specific. Therefore, the representation of the world is different for each system, which results in inconsistencies between different world models. Our approach allows for each robot to have its local world model, enabling to handle

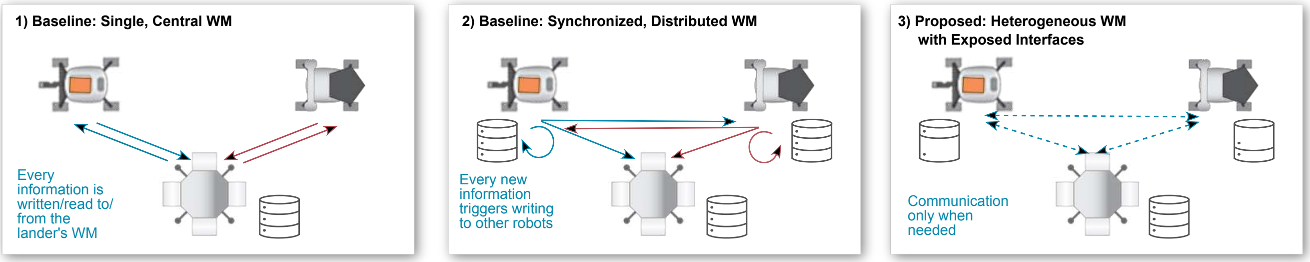

Fig. 7: Schematic communication diagram of the single world model approaches (central and distributed) and our approach.

the inconsistency in a way that makes most sense from the robot's perspective.

Inconsistency also appears due to different interpretation of new information: depending on the tasks and the execution context as well as the purpose of the employed world model, incoming information needs to be interpreted differently. In our use case, for example, LRU1, as a scouting robot to create a global map, interprets object detections made and shared by other robots as landmarks to refine its global pose and map. This means integrating the landmark pose estimates via probabilistic methods into a global model anchored in a robot-external coordinate frame that is to be shared between and used by multiple systems. Such models aim to best approximate the objective reality, e. g., visual and geometrical, properties of the real environment.

In contrast, LRU2, which is assigned to pick up the detected object, might not need such a global model in the context of its manipulation task but instead requires very high local precision to grasp the object. Thus, it might interpret the information shared by other robots only to bring the object into its own sensors' field of view and then use them to estimate a relative robot-to-object transformation in order to compute the target pose for its gripper. For this manipulation task, such a robot only needs its egocentric local estimate and – in contrast to the scouting robot – is not interested in the best approximation w. r. t. a global frame. Using global estimates instead to determine the transformation between robot and object could even lead to manipulation failures as, while these estimates typically are more accurate w. r. t. the global external frame, they also contain errors and biases propagated through the fusion of multiple measurements, estimations of robot movements in between, data from other robot systems, and model approximations, making them less accurate in a local sense.

Through experiments, we show 1) the existence of the inconsistency, 2) that resolving it would introduce new errors to the individual system, and 3) that our concept still allows for world information exchange between the inconsistent models. We conducted five collaborative stone collection experiments with LRU1 and LRU2 in the laboratory (see Fig. 8). A payload box is placed as a landmark at around six meters away from the lander. The task of LRU1 is to find a stone near the box, and LRU2 aims to collect the stone based on the exchanged information by LRU1.

As is shown in Fig. 8, the inconsistency exists in the estimated pose of the stone between LRU1 and LRU2. The pose of the stone estimated by LRU1 does not indicate the right place from LRU2's perspective (Fig. 8-c), and vice versa (Fig. 8-b). Any of the five sample collection trials by LRU2 simply relying on the LRU1's stone pose estimation failed. The difference between the individual representations of the stone pose was 8.18 cm on average with 3.85 cm standard deviation.

If the task were to represent the global pose of the stone, we would have merged these two inconsistent representations, which would result in the offset of around 4 cm in each local world model. For the sample collection task, nevertheless, it would not be precise enough to enable LRU2 to grasp the stone.

For the task of stone picking, we therefore keep the local representation of LRU2. LRU2 asks LRU1 the stone pose, but it is not added to the local world model; instead, the information is used as a hint to identify the correct stone. The pose itself is obtained by using LRU2's perception system. With this approach, LRU2 was able to collect the stone in all five cases, using the information from LRU1 while handling the inconsistencies between the world models.

VI. Lessons Learned and Future Work

In the previous section, we evaluated our approach for our heterogeneous team and showed that it is beneficial to enable heterogeneity in each robot's world model, to achieve efficient communication, as well as to address the limitations of shared world knowledge by augmenting information with local knowledge and interpretation. In this section, we will present the general lessons learned applicable beyond our specific system setup. Furthermore, we discuss the limitations of our current approach and outline future work.

A. Lessons Learned

By applying our approach of exchanging world information by exposing the world model interfaces of the individual agents and using specific communication patterns, we learned some lessons which are relevant for heterogeneous robotic teams in general.

a) Keeping agents independent reduces system complexity: One of the major challenges of heterogeneous robotic teams is the complexity of the overall system. We can address this challenge by keeping individual agents as independent as possible and loosening coupling between them. Rather than maintaining a common world state, sharing

[4]Red lines are key frame matchings and irrelevant to this experiments.

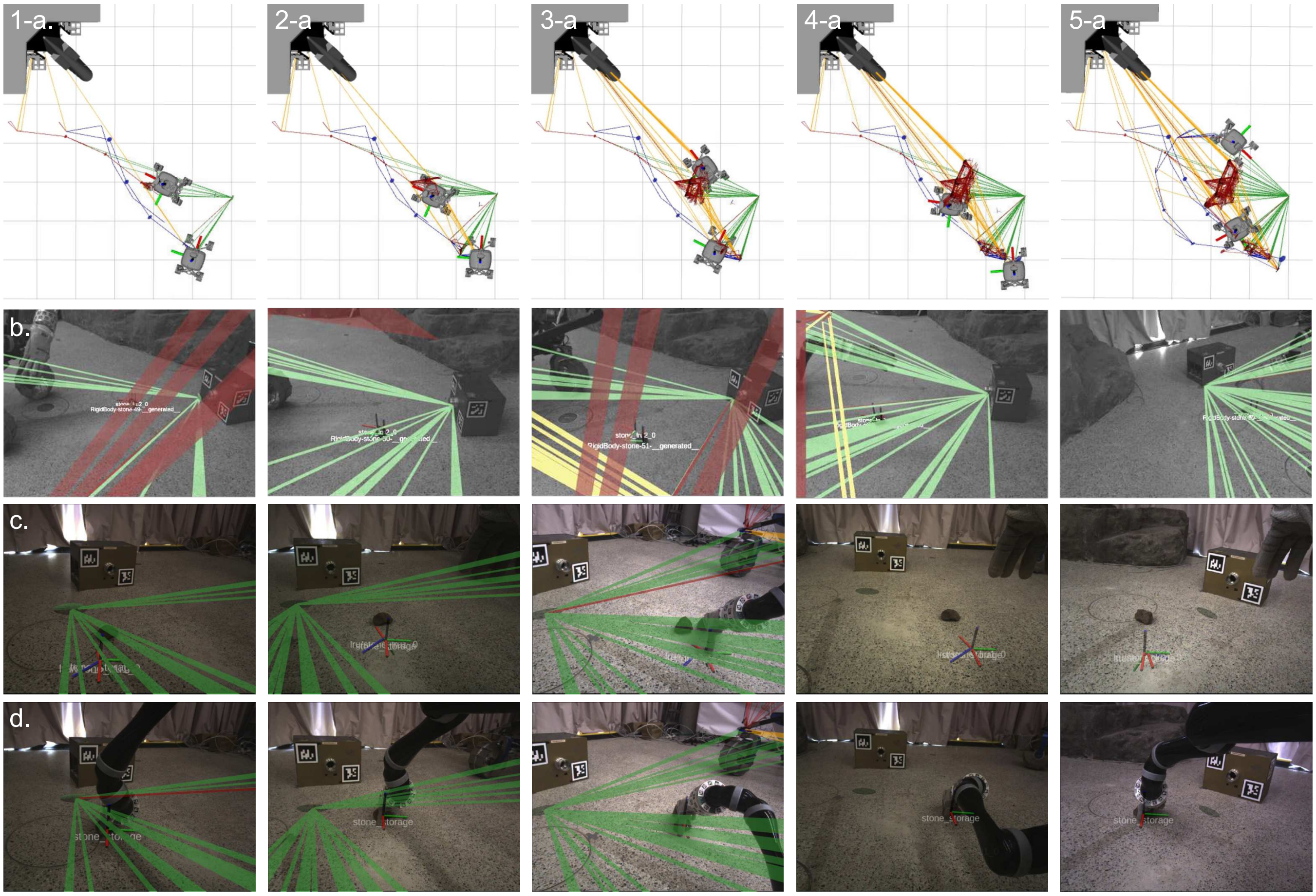

Fig. 8: Snapshots of the five collaborative stone collection experiments. The row a) shows the path of LRU1 (red) and LRU2 (blue) as well as robot detections (orange) and landmark detections (green). The row b) shows LRU1's camera image overlaid with the estimated stone pose of both LRU1 and LRU2[4]. The row c) shows LRU2's camera image overlaid with the stone pose exchanged by LRU1. The row d) shows the pose of the stone detected by LRU2 and stored in its local world model.

only relevant information is highly beneficial to this design goal. Faulty information is only spread over the systems when it is explicitly communicated. This reduces the chance of affecting multiple systems and helps with error isolation.

b) Avoiding a common world model for a team allows for flexible team reconfiguration: Robots are usually designed as individual systems, not specifically for a certain team; they may only temporarily join a team for a particular application or project. Most complex robotic systems use some kind of world model [3], which is often a central building block of the robot's architecture. Exchanging this component with a world model suitable for a specific team greatly impacts the system and requires significant effort, if feasible at all. In contrast, exposing the interfaces of the existing world model and extending it with mechanisms to implement communication patterns has only a limited impact on the system architecture. This is often easy to implement, especially when middlewares are used.

c) Systems in real world need to handle inconsistencies in world states: With a perfect system, each model would accurately reflect the real world, resulting in consistent states. However, in reality, modeling errors, sensor noise, and missing information always affect the state of the model, causing it to deviate from the actual state of the world. Particularly, incoming information to the model contains partially conflicting information, and how these conflicts are resolved depends on the model's purpose. For instance, a world model that aims to minimize global, absolute errors interprets information differently from a model that prioritizes local precision. Consequently, even with the same information, the resulting world model states are inconsistent. The suitable interpretation depends on the agent and the task. For instance, a mapping rover favors the global model, while a sampling rover requires high local precision. Therefore, we learned that, for a heterogeneous team of robots with a variety of tasks, it is necessary to accept inconsistencies in the world models. Furthermore, it is important to develop strategies, such as our communication patterns, that exchange only relevant information, rather than synchronizing data that may differ due to different interpretations of the same information.

B. Limitations and Future Work

With our approach, we were able to solve the problem of information exchange between robotic team members within the ARCHES mission. We believe that this approach can also be transferred to other robotic teams and make valuable contribution to the implementation of the vision of an autonomous robotic team. Nevertheless, due to the given constraints, we did not address certain problems during our implementation that must be taken into account in general application. In the following, we therefore present these limitations and starting points for future work.

a) Necessity of a common language such as Universal Scene Description: To implement our knowledge exchange approach, we used the same software component and a common communication framework of ROS services for each robot. Therefore, there was no technical challenge to interpret data to/from symbolic representation or translate data to another format. However, in general, there need to be either a common language (such as Universal Scene Description [24]) or additional translator nodes.

b) Generalization of who to tell/ask what, when: In our setup, due to the fixed responsibility allocation, it was deterministically implemented which kind of information has to be transferred when to which team member. For instance, the lander was responsible for storing the payload boxes and answering questions about which boxes the lander currently has. When manipulating the payload boxes, LRU2 was responsible for announcing if the box can be used as a new landmark (because it is placed on the terrain), or if it can no longer be (because it is picked up and placed on LRU2 itself). In general, however, roles of robots are not always fixed, and thus another stage in the communication is necessary to determine which robot needs/provides which information and when it is needed.

c) Towards real conversation among robots: In our approach, we combined tell and ask queries for certain tasks, which resembles to a conversation among robots (see Section IV-C and Section IV-D). Nevertheless, the conversation structure in our case was fixed and handcrafted to solve specific problems. Enabling robots themselves to construct such conversations would be very beneficial to achieve a general knowledge exchange capability. There are many open research questions in this respect. For example, robots need to have a way to know which other robots are responsible for certain information. Another challenge is how to deal with inconsistent (or even contradicting) information among different robots, as we discussed in Section VI-A.

VII. Conclusion

In this paper, we presented how we addressed the challenge of exchanging world information among heterogeneous robots including manipulation tasks. Our approach to make each robot have its egocentric world model and let them exchange information by remotely accessible services was demonstrated with our real robotic team. Through in-depth analyses, we evaluated our approach's advantages in allowing for world model heterogeneity of each robot, reducing the data to be exchanged, and enabling interpretation for local belief consistency. We believe the knowledge exchange and world modelling concept presented in this paper as well as the lessons learned provides foundation for achieving general knowledge exchange capability among robots.

References

[1] M. J. Schuster *et al.*, "The ARCHES space-analogue demonstration mission: Towards heterogeneous teams of autonomous robots for collaborative scientific sampling in planetary exploration," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, pp. 5315–5322, Oct 2020.

[2] A. Wedler *et al.*, "Finally! Insights into the ARCHES lunar planetary exploration analogue campaign on Etna in summer 2022," in *73rd International Astronautical Congress (IAC)*, 2022.

[3] R. Sakagami *et al.*, "Robotic world models – conceptualization, review, and engineering best practices," *Frontiers in Robotics and AI*, vol. 10, November 2023.

[4] H. Kitano *et al.*, "RoboCup: A challenge problem for AI and robotics," in *RoboCup-97: Robot Soccer World Cup I*, ser. Lecture Notes in Computer Science, H. Kitano, Ed., vol. 1395. Springer, 1997.

[5] J.-S. Gutmann *et al.*, "The CS Freiburg team: Playing robotic soccer based on an explicit world model," *AI Magazine*, vol. 21, p. 37, 2000.

[6] M. Dietl *et al.*, "CS Freiburg: Global view by cooperative sensing," in *RoboCup 2001: Robot Soccer World Cup V*, A. Birk *et al.*, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 133–143.

[7] H. Utz *et al.*, "Sharing belief in teams of heterogeneous robots," in *RoboCup 2004: Robot Soccer World Cup VIII 8*. Springer, 2005.

[8] P. A. Baer *et al.*, "Communication and collaboration in heterogeneous teams of soccer robots," in *Robotic Soccer*. Citeseer, 2007.

[9] Y. Rizk *et al.*, "Cooperative heterogeneous multi-robot systems: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, pp. 1–31, 2019.

[10] H. Biggie *et al.*, "Flexible supervised autonomy for exploration in subterranean environments," *Field Robotics*, vol. 3, pp. 125–189, 2023.

[11] M. Tranzatto *et al.*, "Cerberus in the darpa subterranean challenge," *Science Robotics*, vol. 7, 2022.

[12] T. Schnell *et al.*, "An efficient scalable autonomy approach for teams of heterogeneous mobile robots," in *19th International Conference on Automation Science and Engineering (CASE)*, 2023.

[13] M. Saboia *et al.*, "CADRE MoonDB: Distributed database for multi-robot information-sharing and map-merging for lunar exploration," in *Int. Workshop on Autonomous Agents and Multi-Agent Systems for Space Applications (MASSpace)*, 2024.

[14] L. Riazuelo *et al.*, "RoboEarth semantic mapping: A cloud enabled knowledge-based approach," *IEEE Transactions on Automation Science and Engineering*, vol. 12, pp. 432–443, 2015.

[15] A. Suarez *et al.*, "Door-to-door parcel delivery from supply point to user's home with heterogeneous robot team: The euROBIN first-year robotics hackathon," *IEEE Robotics & Automation Magazine*, 2024.

[16] M. Logothetis *et al.*, "Efficient cooperation of heterogeneous robotic agents: A decentralized framework," *IEEE Robotics & Automation Magazine*, vol. 28, pp. 74–87, 2021.

[17] N. Y.-S. Lii *et al.*, "Introduction to surface avatar: the first heterogeneous robotic team to be commanded with scalable autonomy from the ISS," in *73rd International Astronautical Congress (IAC)*, 2022.

[18] P. Lehner *et al.*, "Mobile manipulation of a laser-induced breakdown spectrometer for planetary exploration," in *IEEE Aerospace Conference*, 2023.

[19] E. Staudinger *et al.*, "Enabling distributed low radio frequency arrays – results of an analog campaign on Mt. Etna," in *IEEE Aerospace Conference*, 2023.

[20] R. Sakagami *et al.*, "A tree-based world model for reducing system complexity in autonomous mobile manipulation," *IEEE Robotics and Automation Letters (RA-L)*, vol. 9, pp. 6680–6687, 2024.

[21] S. G. Brunner *et al.*, "RAFCON: A graphical tool for engineering complex, robotic tasks," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2016.

[22] M. J. Schuster *et al.*, "Distributed stereo vision-based 6D localization and mapping for multi-robot teams," *Journal of Field Robotics*, 2018.

[23] M. J. Schuster, "Collaborative Localization and Mapping for Autonomous Planetary Exploration: Distributed Stereo Vision-Based 6D SLAM in GNSS-Denied Environments," Ph.D. dissertation, University of Bremen, 2019.

[24] G. H. Nguyen *et al.*, "Translating universal scene descriptions into knowledge graphs for robotic environment," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.