Middleware Challenges in Robotic Fleets for Precision Agriculture

Domagoj Drenjanac, Slobodanka Dana Kathrin Tomic
The Telecommunications Research Center Vienna (FTW)
Vienna, Austria
{drenjanac, tomic}@ftw.at

Abstract—**The past decade has witnessed a huge increase in the number of proposed middleware solutions for robotic fleets operating in unstructured environments. As a result, it has become difficult to decide which middleware is the most appropriate for a specific application or application domain. In this paper we first extract a set of common and specific challenges that middlewares address, and group them according to the source domain they have originated within. These challenges are derived from a specific precision agriculture use-case based on the robotic fleet for weed control elaborated within the European Project RHEA-Robot Fleets for Highly Effective Agriculture and Forestry Management. Furthermore, the paper provides an analises of a number of different middlewares and suggests a set of criteria for systemizing representative solutions. The aim of this analysis is to assist the process of finding an adequate middleware for a specific application domain.**

Keywords—Robotic Middleware; Robotic Fleets; Complex Environment; Multi-Robot System

I. INTRODUCTION

An important current trend in robot-assisted work is a transition from pure industrial robotic systems, where robots carry out work in controlled environments, towards robotic systems and fleets that autonomously work or support humans in unstructured and volatile environments. This entails numerous challenges related to the complexity of robotic mission tasks. Another manifestation of this trend, is emergence of a vast number of different robotic middleware solutions each being customized for and focusing on some specific aspects of autonomous robotic work or human-robot interaction in robotic fleets.

When a new application is being developed, understanding the challenges of a specific application domain is the first important step towards making the decision which middleware to use. Systematization of challenges on a general level has been already undertaken in [1-2], where the focus is on the challenges imposed by unpredictable environments. This work has been extended in [3-4] where authors discuss some more specific issues and compare several middleware implementations against them. Our work further extends this effort with a focus on extracting additional, less covered issues imposed on the middleware for robotic fleets. Through the analysis of a motivating scenario, presented in Section 2, we aim at broadening the perspective introduced in the existing work. Based on the scenario analysis and the extensive review and comparative study of robotic middleware, we propose a classification of middleware challenges according to their specific domains of concern: (1) general issues as related to a fleet realization, (2) issues imposed by operating in unstructured environments, and (3) challenges resulting from the task complexity. This systematization can be used to identify a suitable middleware for a specific robotic fleet application.

This paper is structured in the following way. Section 2 introduces motivating precision farming scenario and Section 3 presents an overview of common and specific challenges extracted from it. Section 4 provides the review of some existing middleware solutions accompanied with their use-case description. Section 5 discusses in more detail several selected middleware solutions and compares them against challenges identified in the Section 3. Section 6 concludes the paper.

II. PRECISION FARMING AS A MOTIVATING SCENARIO

Precise management of agricultural land is possible due to the availability of new technologies, i.e., global positioning systems (GPS), geographic information systems (GIS), sensors, automation of agricultural machinery, and high resolution image sensing. The aim is to diminish the use of chemical inputs and improve crop quality, humans' safety, as well as to reduce the production costs. Sustainable precision crop management can be based on the use of a fleet of heterogeneous robots equipped with advanced sensors and actuators as developed within the RHEA project [5]. It provides motivation and requirements for the presented work.

The core of the RHEA concept is a centralized fleet management system that assists the system operator in choosing a suitable strategy for field treatment taking into account weed infestation map and available robots, their implements and sensors [6]. The selection of the treatment strategy, i.e., building a mission, takes into account many parameters, e.g., the type of tasks to be performed, the number and features of available robots and field information [7]. After the mission is defined, it is decomposed in the number of tasks and mapped to corresponding robots. Centralized control system is responsible for both task-robot mapping and robots coordination during the mission execution. During the mission, heterogeneous and distributed robots report their status to the base station (user) who supervises the mission and acts as a central point in the system. The base station is a place that manages, coordinates, makes decision, collects data,

---

This paper is supported within the European Research Project RHEA.
The Telecommunications Research Center Vienna (FTW) is supported by the Austrian government and the City of Vienna within the competence center program COMET

instructs, and monitors all robots in the network. In addition, a user (operator) in the field can establish remote control over the fleet, complementing the central user when necessary.

In this basic RHEA scenario the robots´ autonomy is limited to a narrow set of basic functions like small adjustments related to, e.g., path correction. Our interest is to understand how autonomy, i.e., the robot self-awareness, can be extended, and the implications of this extension on the robotic middleware selection. In this respect we are particularly interested in the aspects pertaining to the modelling of the tasks, and the resources and services that the robots embody.

Fig. 1 illustrates a precision farming concept of the RHEA project including the mission that comprises tasks, the mission monitoring and centralized system control. This concept is extended with the blocks that represent models of tasks (task ontology) and robotic coordination, as well as self-awareness knowledge model, and models of robotic resources and services (ontology for service description). We refer to the concept illustrated in Fig. 1 as the extended RHEA scenario.

III. MAIN CHALLENGES FOR ROBOTIC MIDDLEWARE

This section summarizes middleware challenges posed by the RHEA scenario where robotic fleet operates in unstructured and volatile environment. We complement the challenges from the related work [1-4] with the challenges specific to the scenario from Fig. 1. The result is a classification of middleware design challenges, addressing three domains of design decisions: (1) general fleet realization, (2) dealing with the environment, and (3) dealing with the specificities of fleet tasks. This classification provides a scope in which we map existing robotic middleware solutions in Section 5, and identify potential for new developments.

A. *General Fleet Realization Challenges*

We identified five pervasive problem statements common for the RHEA scenario as well as for the reviewed middleware [8-27].

1. How to distribute control? Centralized versus Distributed Control. The control organization is a concern critical in the early stages of the system design [1], [4]. It influences other system decisions that are to be made in later phases, e.g., the autonomy level and collaboration patterns. As also shown in Fig. 1, the control component dominates the system as it manages the communication and coordination. While centralized control adopted in the basic RHEA scenario simplifies coordination, the distributed control provides for more flexibility (extended RHEA).

2. How to distribute functionalities? Homogeneous versus Heterogeneous Robots. There are two types of fleets: homogeneous and heterogeneous. The robotic fleet application directly drives a decision on the type of robots in a fleet [1], [2]. The systems exploiting parallel, and in time and space distributed tasks, often use large scale fleets of interchangeable homogeneous robots. On the other hand, more demanding applications may require teams of individuals with specific sensors or actuators, like robots in the RHEA scenario.

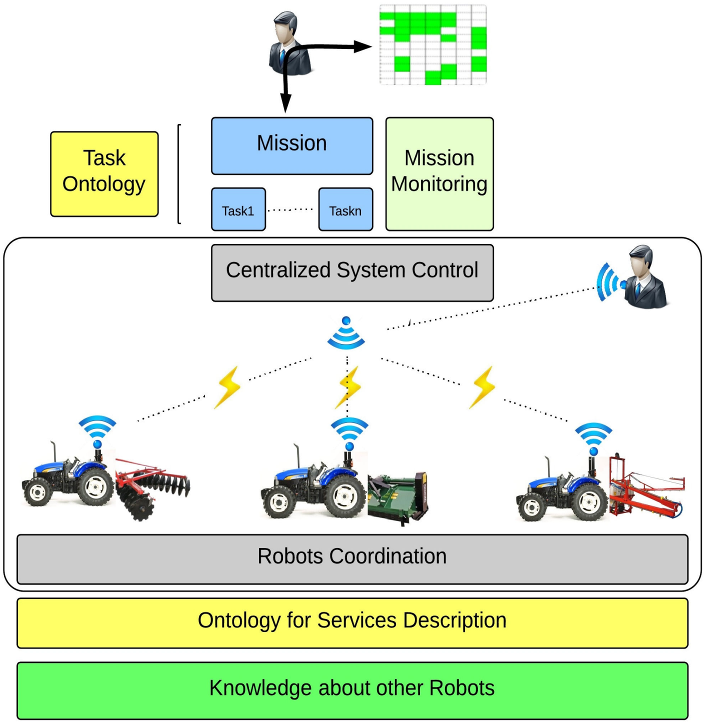

Fig. 1. Extended RHEA scenario

3. How much autonomy is needed? Communication, Collaboration and Coordination as Autonomy Drivers. Information exchange is vital for collaboration and coordination, which are complementary processes running over a communication backbone of multi-robot systems [1-4]. The information that supports robots in achieving their goals can be obtained in different ways, e.g., by sensing the environment in which robots are operating, by observing actions of the peers, or by an explicit message exchange with the peers. Based on the acquired information, robots can build their collaboration and coordination patterns. Furthermore, autonomy is significant feature of robotic systems [3]. Autonomous robots utilize communication infrastructure and sensors for automated local or group decisions and actions, with no human intervention. Current research in autonomy is focused on developing different levels of autonomy and thus providing the robots with *adaptive* autonomy functions. Adaptive autonomy enables human-robot fleets to incorporate advanced coordination and collaboration mechanisms with different levels of robots and humans involvement. The communication in basic RHEA is centralized with a hub spoke topology. The central user (base station, or hub), creates, coordinates and supervises a complete mission. Spokes are heterogeneous tractors that wirelessly communicate with the base station, execute commands received from the user, and report their status to the hub. End-to-end communication is established only between the operator in the base station and each particular tractor.

4. How to specify a mission? Mission Definition and Task Allocation. A robot task can be decomposed into independent subtasks, hierarchical task trees or roles [1]. The mission tasks can be designed either by an autonomous planning function, or by the human designer [1], [2]. Centralized fleet management system in RHEA assists the system operator in building a mission for the field treatment [7]. The mission is decomposed in tasks which are mapped to corresponding robots. In general, a common way of defining a mission is by defining a set of tasks that have to be completed within a specified time. Each task can be processed by a variety of different robots, and each robot can work on different tasks. The mapping between available robots and generated tasks is the solution to the task allocation problem.

5. How to design for self-organization? The Knowledge-based design and Semantic-based Resource Presentation. Semantics can be used to model resources provided by ubiquitous heterogeneous devices distributed in the environment. Describing functions of cameras, sensors, actuators, etc., with a common ontology eases the process of finding appropriate resources in the environment. Semantic technology is underutilized in robotic fleets but has a large potential due to the flexibility it offers. As shown in Fig. 1, we extend the initial assumptions of the RHEA scenario with semantic blocks for describing tasks and services that reside on the robots and the base station respectively. In this new scenario, robots are aware of their local context and capabilities of other robots in the fleet.

We identified these core challenges as both imposed by RHEA scenario and imminent to the design of every middleware. They have to be continuously addressed in different phases of the middleware design, and build a basis for extensions according to special needs of robotic application.

B. Environmentally Imposed Challenges

This class comprises of three problem statements characteristic for robots employed in RHEA scenario, which have to cope with unstructured and volatile environment that introduces additional complexity in system implementation.

1. How to deal with uncertainty in communication? Disconnected versus Connected Operation Mode. Fig. 1 shows the robots operating outdoor where communication links are subject to environmental impacts and therefore unreliable. Although the basis RHEA robots do not possess modules for operating in disconnected mode, this may be a crucial requirement for a robotic middleware applied in unstructured environments. In most scenarios with intermittent communication robots have a connection with others, but only for a limited, unknown time.

2. How to deal with dynamics due to faults? Robustness, Fault Tolerance and Adaptability. Robustness and fault tolerance are relevant features of every multi-robot system that executes time critical tasks [1], [4]. In RHEA they are embedded within the reliability concept that includes a remote user in a filed who can at any time manually take over a control of a robot and solve aggregated issues. In general, robustness and fault tolerance guarantee operation in the presence of malfunctioning components, which requires that the system has autonomy capabilities to continue the work with reduced resources [4]. Adaptability is a design feature that enables robots to change behavior according to the dynamically changing requirements posed by the environment, e.g., as triggered by mission customization, changing resources of teammates, or the need to prevent performance degradation [2], [4], [8].

3. Can a new behavior emerge? Emerging Behavior and Dynamic Team Formation. Behavior-based system design enables robots to perform tasks without having explicit set of instructions for their execution [1]. In this context, we extend the initial assumptions of the RHEA scenario with the cooperation and collaboration capabilities at robots, assuming that robots are aware of their local context and capabilities of other robots in the fleet. This means that they can autonomously select to perform tasks within a complex mission, which requires e.g., a combination of different implements (skills), such as spraying and flaming implement. In general, robots may use knowledge of the current state of the robot mission, robot team member capabilities, and robot actions, to decide, in a distributed fashion, which robot should perform which task. Dynamic team formation may occur either when one or more robots move away and lose connectivity to others, or when a robotic fleet is split in groups according to assigned tasks. The former is the physical, and the latter is the logical separation.

Due to the volatile environment where RHEA scenario takes a place, support for the operation in the disconnected mode is perceived as a comparative advantage in a robotic middleware. Ability to operate in this mode, complemented with an adaptive behavior and dynamic team formation, increases system robustness.

C. Task Dependent Challenges

Task specific challenges are related to mission or task requirements that may be different in each application domain. Here we identify four common problem statements:

1. How big the fleet may be? Scalability. The scalability support is an essential feature of a fleet of robots operating outdoor in unstructured environments [4]. In RHEA, the dimensioning of the fleet as a task specific challenge greatly depends on a size of the field that has to be treated. In general, scalable, open systems, e.g., systems comprising of different types and numbers of components, such as agents, computers, humans, have to support dynamic joining and leaving of components. Scalability relates to the ability of the system to accept new components without significant change in architecture and design.

2. How much knowledge shall be shared? Knowledge about other Robots and Resource Discovery. Shared knowledge is a driver for successful coordination between robots [4]. To attain knowledge about other robots in a fleet, a single robot does not have to contact a centralized knowledge repository. Instead, the local (distributed) knowledge can be maintained and used in tasks where robots have to combine their services in order to successfully accomplish a given task. In this context, discovery mechanisms are essential components of dynamic computing environments [2]. During environment exploration, mobile robots discover external resources, like cameras, sensor networks, and configure themselves to interact with them. While the basis RHEA scenario does not involve information sharing among robots, this feature is perceived as a main driver for advanced and distributed coordination mechanisms in the extended RHEA.

3. What is the human role? Human-Robot Interaction. Robotic fleets are designed for limited autonomous operation in the field and thus require interaction with humans. The requirements on the human-robot interaction capabilities depend on the challenges of specific applications. A control concept exploiting different levels of autonomy is an important trend in human-robot interaction research [16]. There are two human operators in RHEA, one at the base station and the other in the field, and both of them can manually control each robot. This type of adjustable autonomy increases the robustness and therefore is important in the system modeling.

4. Shall fleet resources be dynamically allocated based on preferences? Context and Costs Awareness. The context and costs awareness are important task-specific challenges [14], [19]. To support cost-based decision making each task has to be assigned with an objective function to minimize the cost of resources and maximize the benefit gained by performing that task. In this way a fleet operates as a system where actions are driven by business objectives. A concrete RHEA fleet is a result of a centralized ahead-of mission optimization; RHEA robots are not able to autonomously and dynamically select desired, most appropriate tasks, rather, they are assigned to them. In the extended RHEA scenario cost awareness is a function that each robot will use to estimate a gain it receives from executing a specific task.

It is hard to identify one specific task dependent challenge as the most relevant because the requirements depend on the application domain. Thus a middleware designer needs to decide which should receive more attention, based on requirements of the specific use-case.

IV. REVIEW OF MIDDLEWARES SOLUTIONS

In this section we review a number of prominent existing middleware solutions with a focus on a particular use-case in which the system was used. We consider only solutions that deal with higher layer functionalities including ALLIANCE [8], MARTHA [9], Collaborative tasking middleware [10], PEIS [11-13], a market based approach [14] which we refer to as MarketE, and HART [15]. Solutions controlling hardware components on robots are out of the scope of this review.

We restricted ourselves to this selection from a large number of middleware solutions we reviewed, because each of them addresses a larger number of challenges we identified as important and their use-cases are similar to RHEA. This is not the case for the majority of other existing implementations which address more restricted number of challenges. Also, we do not include solutions whose application is limited to a specific purpose, such as DIRA [17], LIME [18], SOLD [19] and AuRA [20].

A. Alliance [8]

ALLIANCE [8] defines a framework that allows teams of robots, where each is equipped with a variety of high-level functionalities, to individually select appropriate actions based on the mission requirements, the environment, activities of other robots and an internal state. The middleware framework is distributed and implements behavior-based architecture which enables robots to act based upon their current state.

In [8] authors test this framework in the hazardous waste cleanup mission, which requires two artificially waste spills in an enclosed room to be cleaned up by a team of three robots. The mission is divided into following tasks required to be executed by the robots: the robot in the team locate two waste spills, move the spills to a goal location, and periodically report the progress to the human monitoring system.

B. MARTHA [9]

MARTHA [9] focuses on the control and management of autonomous fleets for transshipment tasks in harbors, airports and marshaling yards. The focus is on the increase of robots' autonomy as a key solution for decentralization, which allows robots to efficiently cope with unexpected environmental issues, e.g., obstacles and other robots. The central station does not intervene in the robot coordination tasks, nor does it calculate precise trajectories robots have to take. Thus, the required communication bandwidth between robots and the central station is very low.

The testbed in [9] consists of an experimental room, dimension 10 x 7 meters, divided into two areas including six docking stations and two lanes. Three indoor robots conduct transshipment experiments in the testbed environment demonstrating how coordination and synchronization issues that emerged during a mission are being handled.

C. Collaborative Tasking [10]

Collaborative Tasking [10] middleware supports market-based task allocation through the implementation of the standard Contract Net Protocol called Collaborative Tasking Protocol (CTP) for a group of heterogeneous unmanned vehicles. When a task is injected, each vehicle estimates its cost to perform the task, taking into account remaining consumables, required effort, its other pending tasks, and user preferences.

In a military use-case presented in [10] an observer reports the coordinates of an enemy tank and requires an air strike. To begin, an operator adds the target to the workspace and initiates the air strike requiring three tightly coordinated units: first verifies a target, second bombs the target, and third performs damage assessment. It is a challenging task since close coordination between units is required to prevent collisions. The mission is decomposed into three tasks broadcasted to the units. Upon receiving a task, each unit computes its cost function to execute a task and sends a bid. After bids are collected, minimization algorithm is used to select best unit for each task.

D. *PEIS (Physically Embedded Intelligent Systems) [11-13]*

The concept of Ecology of Physically Embedded Intelligent Systems or PEIS-Ecology aims at building intelligent robots in the service of people [11]. In general, PEIS is defined as a set of connected PEIS components that reside in the same physical place. PEIS-kernel enables each PEIS component to communicate and participate in a PEIS-Ecology by implementing distributed tuplespace. PEIS components use cooperation model based on the linking of functionalities: each PEIS component is able to use services provided by other PEIS components complementing its own functionalities.

The scenario simulating PEIS environment consists of two robots (one robot supervises a mission and other executes it) and a tracking system (cameras) installed in a 25m^2 apartment. First robot (supervisor) receives a goal to wake up a person and it generates a plan and delegates it to the second robot (executor). The plan consists of three actions: goto bed, wakeup (talk), goto sofa. While executing each action, the executor utilizes information provided by the tracking system in order to better keep track of own position in the environment. When the executor reaches the final position, it notifies the supervisor. It is worth to notice that all communication and synchronization is enabled by the exchange of the tuples through the distributed tuplespace.

E. *MarketE [14]*

This middleware exploits market architecture to maximize information gain while minimizing incurred costs [14]. It uses the concept of market economies, which are distributed systems where individuals exchange goods and services by establishing contracts. Here, multiple robots interact in a distributed fashion to attain global goals in the efficient way by maximizing their profits.

Authors in [14] utilize 10 robots equipped with a ring of 16 ultrasonic sensors to construct occupancy grids of the environment as the robot navigates. Robots aim to build a map of an indoor space (45m x 35m) cluttered with many obstacles. The success of mapping is quantified as a metric that is proportional to the area covered by a particular robot and inversely proportional to the cost incurred (traveled distance). Results show that robots were more efficient when they coordinate using the market model to exchange maps; they covered wider area than in the case where they were concurrently exploring the area.

F. *KAoS HART (Human-Agent-Robot Teamwork) [15]*

KAoS HART [15] is a middleware that supports coordination in a mixed-team with adaptive autonomy where human-robot interaction takes place. Due to its hierarchical organizational structure, control is centralized and performed by the team leaders. A team leader defines a common goal and monitors its execution. Other team members follow the leader. Team members register at centralized directory service where they publish a description of capabilities they provide. This enables them to perform a lookup for desired services and match them against their own requirements. Coordination among team members is based on a set of policies that manage the organizational structure among the agents.

Scenario in [15] encompasses a mixed human-robot team exercised to isolate, find and apprehend a human intruder on a pier. Commander and lieutenant roles are assigned to two humans and they were accompanied by 5 robots. The commander is responsible of establishing subteams and managing the whole mission relying on a combined speech and graphical interface. The lieutenant is assigned to a team just like robots. Mission starts when the commander states a task of finding an intruder and uses utterances such as “what resources are available” and “who has a laser” to build two teams, one autonomous and the other with the lieutenant as a leader. After the teams were created, the commander defines an area of search interest and uses natural language to task each team to secure a particular side.

V. COMPARATIVE STUDY

We conducted a comparative study in order to detect gaps which existing middleware do not address, indicate caveats in a design of a distributed middleware, and help developers to avoid them. We use it to drive further development of a middleware framework for the extended RHEA scenario.

Three tables presented in this section summarize mapping between challenges (columns) from the Section 3 and selected middleware implementations (rows). Fleet realization challenges discussed in the Section 3.A are reviewed in Table 1. Environment specific are included in Table 2 and task entailed are shown in Table 3.

In Table 1, the column names stand for: CS - Control Structure referring to centralized or distributed control, RD - Robots Diversity referring to heterogeneous or homogeneous robots, 3C - Communication, Collaboration and Coordination, AUT for autonomy, MD & TA - Mission Definition and Task Allocation, and SEM - Semantics.

ALLIANCE is a distributed software architecture (Table 1, CS = D) that facilitates fault tolerant cooperative control (Table 2, RS) of heterogeneous (Table 1, RD = Het) mobile robots. The control is distributed and supported via control mechanisms deployed on all robots. Robots have different abilities, e.g., different sensors and actuators. Information sharing occurs when each robot broadcasts (Table 1, 3C = Broadcast) a state of its current actions on which other robots are listening. Collaboration is attained through the common work on same tasks. If one robot cannot finish a task it is in charge of, within the required time, or is not able to finish it at all, other robots will be notified (Table 1, AUT). Due to the implemented features, especially of the support for distributed and heterogeneous robots, ALLIANCE partially satisfies the requirements of the extended RHEA scenario, however it lacks support for fine-granular mapping of the mission into a set of tasks and for task allocation.

In MARTHA the Central Station coordinates and monitors robots which perform transshipment tasks in a harbor. Due to the centralized control (Table 1, CS = C), the framework is able to control both heterogeneous and homogeneous robot (Table 1, RD = Gen). Whenever a robot produces a plan,

TABLE I. MIDDLEWARE COMPARED AGAINST FLEET REALIZATION CHALLENGES

Name	CS	RD	3C	AUT	MD & TA	SEM
ALLIANCE	D	Het	Broadcast	✓		
MARTHA	C/D	Gen	Plan Merging	✓	Local task planning	
Collaborative Tasking	C	Het	Broadcast, Bidding		Market-based	
PEIS	D	Het	Tuplespaces	✓	Tuple collection	✓
MarketE	D	Gen	Price-map exchange	✓	Tasks trading	
KAoS HART	C	Het	Policies	✓	Uusing utterances	✓

which uses some shared resources, cells or trajectories, it advertises it, and collects from other robots their resource usage plans. Then it produces a coordinated plan and informs the other robots of events like cells exit or particular point traversal on trajectory (Table 1, 3C = Plan Merging). Furthermore, different strategies for plan coordination are proposed, with the local scope of planning actions, instead of a global. This means that robots plan their actions locally instead of receiving them from a centralized place (Table 1, AUT, MD&TA = Local Task Planning). Instead of planning a whole mission from the beginning, an incremental approach provides for continuous and fluid tasks execution. In this way robots can better react on hazardous events. The communication between robots has higher priority than the communication with a Central Station, which itself requires a high bandwidth and reliable communication link. Considering the system architecture and the use-case, this system resembles both basis and extended RHEA scenario more than the others do. Introducing local and incremental approaches for a mission planning on each robot could increase autonomy of the robots in RHEA scenario.

Collaborative Tasking approach automatically designs and re-designs tasks for a group of heterogeneous (Table 1, RD = Het) unmanned vehicles. It uses a central agent (Table 1, CS = C), taskLead, who runs the bidding processes (Table 1, 3C = Bidding). The taskLead agent runs the Collaborative Tasking Module which decomposes a high-level mission tasks into executable tasks and broadcasts (Table 1, 3C = Broadcast) these to the robots in charge for performing them. Robots are heterogeneous with complementary services that are combined together towards successfully achieving mission goals. Each robot knows a set of tasks it is capable to perform, as well as the cost of performing a specific task under current circumstances. The middleware supports market-based task allocation (Table 1, MD & TA = Market based) through the implementation of the extended version of the standard Contract Net Protocol called Collaborative Tasking Protocol (CTP). General features of the framework can be applied to basis RHEA scenario because the framework has a central place which makes a final decision on a mapping between tasks and robots, while robots cannot autonomously select tasks.

PEIS solution addresses all challenges summarized in Table 1. Decentralized control (Table 1, CS = D) is identified as a main requirement in the peer-to-peer PEIS-Ecology. The solution to decentralization is based on advertisements by each PEIS-component in the same space where this component runs. Distributed tuplespaces enable each component to make a decision locally (Table 1, 3C = Tuplespace), within its own decision space rather than having a central decision making system. Furthermore, to establish information sharing between heterogeneous components (Table 1, RD = Het), each PEIS component displays its services and functionalities by sending XML based messages. Components use cooperation model based on the linking of functionalities: each PEIS component is able to use services provided by other PEIS component complementing its own functionalities (Table 1, AUT). The semantics is introduced to overcome the issue of heterogeneous distributed components (Table 1, SEM). PEIS approach can be used to address all basic challenges in the RHEA and therefore can serve as a solid basis for development of additional, more complex coordination and collaboration algorithms. Semantic resource description is a comparative advantage over the other frameworks which makes a whole system easy extendable.

MarketE distributes control mechanisms over the robots assigned to explore a certain area (Table 1, CS = D). Each robot, which implements negotiating algorithms can participate on a market (Table 1, MD & TA = Tasks trading) and therefore compete for announced resources. Thus, the framework supports both heterogeneous and homogeneous robots (Table 1, RD = Gen). The robots make decisions by communicating price information and continuously negotiating with others to improve their plains (Table 1, 3C = Price-map exchange). In addition, robots explicitly share their maps of visited, explored, areas in exchange for revenue (Table 1, AUT). Developed framework exposes results that show how the collaboration between robots increases task and mission efficiency. Therefore, the extended RHEA scenario could benefit from the concepts of MarketE middleware.

KAoS HART is a middleware with centralized control (Table 1, CS = C) that supports coordination in a mixed-team where human-robot interaction takes place. Mixed human-robot teams introduce heterogeneity in the framework (Table 1, RD = Het). The framework uses policies implemented in OWL (Web Ontology Language) as rules for dynamically regulating behaviors imposed by different components (Table 1, 3C = Policies, SEM). Policies are used for a mapping between natural language and commands which are basis for a successful coordination between team leaders and team members. The commander uses same policies to decompose a mission into tasks, and by utilizing utterances it delegates those tasks to specified team (Table 1, MD & TA = Utterances). RHEA can benefit from utilizing policies approach in coordination mechanisms that could be distributed over the robots.

In Table 2 column names stand for: DM - Disconnected Mode, RS - Robustness, ADAPT - Adaptability, EB - Emerging Behaviors and DTM - Dynamic Team Formation.

TABLE II. MIDDLEWARE COMPARED AGAINST ENVIRONMENT SPECIFIC CHALLENGES

Name	DM	RS	ADAPT	EB	DTM
ALLIANCE		✓	✓	✓	
MARTHA		✓	✓		
Collaborative Tasking		✓			
PEIS					
MarketE	✓	✓			
KAoS HART		✓	✓		✓

TABLE III. MIDDLEWARE COMPARED AGAINST TASK SPECIFIC CHALLENGES

Name	SCL	AK	RSD	HRI	CA
ALLIANCE		✓		✓	
MARTHA			✓	✓	
Collaborative Tasking				✓	✓
PEIS	✓		✓		
MarketE	✓				
KAoS HART	✓			✓	

ALLIANCE has control mechanisms that rely on different sets of behaviors (Table 2, EB) where the activation of a certain behavior depends on: (1) the efficiency of performing a local task, and (2) how efficiently the teammates are performing their tasks. The framework tackles adaptability (Table 2, ADAPT) and emerging behavior as prominent challenges imposed by the extended RHEA scenario. Since robots in MARTHA plan the mission locally, instead of communicating it with the Central Station, they expose certain level of robustness and fault tolerance (Table 2, RS). Furthermore, the local mission planning system increases adaptability and autonomy levels (Table 2, ADAPT) and thus could be a pertinent feature to implement in the RHEA scenario. Collaborative Tasking Module integrates mechanisms for handling unsuccessful task allocation processes as a means for robustness (Table 2, RS). PEIS framework does not directly address any of the challenges related to the open environment because it is designed to operate in the structured and controlled environments. However, it supports semantic resource modeling that makes it easy adaptable to various use-cases.

Due to the communication uncertainties in MarketE, the robots are equipped with mechanisms to retain system's functionalities with zero communication (Table 2, DM). Robots´ actions are triggered by arrival of messages that contain goals the robot is going to execute. If for some reason the robot did not receive a message it expected, either due to a communication problem or due to its peer's failure, it has to be able to proceed with a task rather than indefinitely wait. Hence, unreliable wireless communication does not disable a team to perform tasks, but can reduce its efficiency (Table 2, RS). Retaining operation capabilities in a case of a reduced or even broken communication is a desired feature of robotic fleet systems and very relevant in the extended RHEA scenario. KAoS HART supports dynamically changing policies and therefore can accommodate changes imposed by volatile environment (Table 2, RS). Furthermore, using utterances that are mapped to commands utilizing policies, the system can easily and dynamically form teams (Table 2, ADAPT, DTM).

In Table 3 column names stand for: SCL - Scalability, AK – Aggregated Knowledge, RSD – Resource Discovery, HRI – Human-Robot Interaction and CA – Cost Awareness.

Robots in ALLIANCE share information about the status of a task they are performing with teammates establishing mutual support and enhancing mission's efficiency. Information sharing encourages robots to learn about actions and knowledge of peers (Table 3, AK). At the same time, exchanged information is also presented to the users building a basis for human-robot interaction (Table 3, HRI). The fact that robots are enriched by having knowledge about teammates, would enhance the efficiency of coordination mechanisms also in the RHEA scenario. Consequently, coordination can be transferred from a central place to distributed robots.

MARTHA proposes incremental resource discovery (Table 3, RSD) and acquisition process as a more flexible way of adaptation to dynamically changing environment. Human-robot interaction in MARTHA is bidirectional: on the one hand, robots receive a high-level mission from an external user and on the other, they continuously send their status to the Central Station (Table 3, HRI). External user in RHEA has a wider set of functionalities on a disposal, which can be utilized for a robot control as well. In Collaborative Tasking when a task is injected, each vehicle estimates its cost to perform the task taking into account remaining consumables, required effort, its other pending tasks, and user specified preferences (Table 3, CA). A high-level mission definition is a product of a remote operator who uses a specific device with a customized user interface (Table 3, HRI). Costs awareness is a feature desired also in the RHEA scenario.

PEIS implements dynamic join and leave of components (Table 3, SCL). Each PEIS-component describes in a formal way services it offers, together with required input and output ports, dependencies, type of data, etc. By having uniform semantic-based services description, all heterogeneous components are able to communicate and cooperate (Table 3, RSD). MarketE easily scales and the new robots only have to implement desired negotiation algorithms in order to participate in a mission (Table 3, SCL). An additional feature of the policies in KAoS HART is to provide a scalability support (Table 3, SCL). When a new robot joins, it automatically acquires the intelligence possessed by the others. User interface provides a user with information from robots, such as position, state, video, thereby, enabling monitoring of the system behavior (Table 3, HRI). As policies support a wide set of features the RHEA scenario can benefit from implementing them. Policy-based automatic mapping between task requirements and available resources in the RHEA makes the coordination process more flexible.

VI. CONCLUSION AND FUTURE WORK

Designing middleware for application in complex robotic systems requires that many challenges related to the fleet operation are being addressed. In this work we focus on a robotic fleet operating in an unstructured environment, in particular in precision agriculture scenario introduced by the RHEA project, aiming at extracting middleware challenges common for similar use-cases.

The paper presents identified challenges classified in three domains of concern: (1) general fleet realization, (2) environmentally entailed challenges, and (3) task specific challenges. The latter two are specific for the fleet application. The benefits of some of the design options are already well understood: the solution that enables decentralized control of autonomous and heterogeneous robots has advantage over the one which lacks those features. Specific tasks and environments introduce a set of additional design constraints that are critical for operation in missions with high uncertainty.

The review of existing solutions revealed to us that the semantic-based task and service description is an approach that only a small number of solutions pursue. This motivates our future work on the extended RHEA scenario to be steered in that direction, in particular focusing on extension of a semantic solution that combines space-based middleware with semantic modeling, processing and reasoning.

REFERENCES

[1] L.E. Parker, "Multiple Mobile Robot Systems", Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics. Springer, Heidelberg, 2008, pp. 922-941.

[2] V. Kumar, D. Rus, G.S. Sukhatme, "Networked Robots", Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics. Springer, Heidelberg, 2008, pp. 943-958.

[3] N. Mohamed, J. Al-Jaroodi, I. Jawhar, "A Review of Middleware for Networked Robots", International Journal of Computer Science and Network Security 9, 2009, pp. 139-148.

[4] P. Inigo-Blasco, F. Diaz-del-Rio, C. Romero-Ternero, D. Cagigas-Muniz, S. Vicente-Diaz, "Robotics software frameworks for multi-agent robotic systems development", Elsevier Robotics and Autonomous Systems 60, 2012, pp. 803-821.

[5] RHEA project, http://www.rhea-project.eu/

[6] C. Fernandez-Quintanilla, J. Dorado, C. San Martin, J. Conesa-Munoz, A. Ribeiro, "A Five-Step Approach for Planning a Robotic Site-Specific Weed Management Program for Winter Wheat", 1st International Workshop on Robotics and Associated High Technologies and Equipment for Agriculture, September 9, Montpellier, France, 2011.

[7] J. Conesa-Muniz, A. Ribeiro, "An evolutionary approach to obtain the optimal distribution of a robot fleet for weed control in arable crops", EFITA 2011,11-14 July, Prague, CZ, 2011.

[8] L.E. Parker, "ALLIANCE: An Architecture for Fault-Tolerant Multi-Robot Cooperation", IEEE Transactions on Robotics and Automation. 14, 2008, pp. 220-240.

[9] R. Alami, S. Fleury, M. Herrb, F. Ingrand, F. Robert, "Multi-Robot Cooperation in the MARTHA Project", IEEE Robotics & Automation Magazine, 36-47 (1998)

[10] D.C. MacKenzie, "Collaborative Tasking of Tightly Constrained Multi-Robot Missions", 2nd International Workshop on Multi-Robot Systems, Washington, 2003, pp. 39-50.

[11] M. Broxvall, B. Seo, W.Y. Kwon, "The PEIS Kernel: a Middleware for Ubiquitous Robotics", IROS-07 Workshop on Ubiquitous Robotic Space Design and Applications, San Diego, 2007.

[12] A. Saffiotti, M. Broxvall, "PEIS Ecologies: Ambient Intelligence meets Autonomous Robotics", sOc-EUSAI (Smart Object and Ambient Intelligence), Grenoble, 2005.

[13] M. Gritti, M. Broxvall, A. Saffiotti, "Reactive Self-Configuration of an Ecology of Robots", IEEE International Conference on Robotics and Automation, Rome, 2007.

[14] R. Zlot, A. Stenz, M.B. Dias, S. Thayer, "Multi-Robot Exploration Controlled by a Market Economy", ICRA '02 IEEE International Conference on Robotics and Automation, Washington, 2002.

[15] M.J. Johnson, K. Intlekofer, H. Jung, J.M. Bradshaw, J. Allen, N. Suri, M. Carvalho, "Coordinated Operations in Mixed Teams of Humans and Robots", International Conference on Distributed Human-Machine Systems 2008 (DHMS 2008), Athens, 2008.

[16] D. Drenjanac, S. Tomic, "User Interactions with Robotic Fleets for Intelligent Agriculture", 1st RHEA International Conference on Robotics and associated High-technologies and Equipment for Agriculture, Pisa, 2012.

[17] R. Simmons, S. Singh, D. Hershberger, J. Ramos, T. Smith, "First Results in the Coordination of Heterogeneous Robots for Large-Scale Assembly", ISER 7th Int. Symp. Exp. Robot. Springer, New York, 2000

[18] A.L. Murphy, G.P. Picco, G.C. Roman, "Lime: A Middleware for Physical and Logical Mobility", 21st International Conference on Distributed Computing Systems ICDCS, IEEE Computer Society, Washington, 2001, pp. 524-533.

[19] B.P. Gerkey, M.J. Matarić, "Sold!: Auction Methods for Multirobot Coordination", IEEE Transactions on robotics and Automation, 2002, pp. 758-768.

[20] T. Balch, R.C. Arkin, "Behavior-based Formation Control for Multi-robot Teams", IEEE Transactions on Robotic and Automation, 1999, pp. 1-15.

[21] S. Yuta, S. Premvuti, "Coordinating Autonomous and Centralized Decision Making to Achieve Cooperative Behaviours Between Multiple Mobile Robots", IEEE/RSJ International Conference on Intelligent Robot Systems, Raleigh, 1992.

[22] W. Burgard, M. Moors, C. Stachniss, F. Schneider, "Coordinated Multi-Robot Exploration", IEEE Transactions on Robotics, 2005, pp. 376-386.

[23] J. Wang, L. Michael, P. Scerri, "Cooperating Robots for Search and Rescue", AAAMAS ´06, Hakodate, Japan, 2004.

[24] C. Castelpietra, L. Iocchi, D. Nardi, "Coordination among Heterogeneous Robotic Soccer Players", IEEE/RSJ International Conference on Intelligent Robots and Systems,Takamatsu, Japan, 2000.

[25] I. Navarro, J. Pugh, A. Martinoli, F. Matia, "A Distributed Scalable Approach to Formation Control in Multi-robot Systems", 9th International Symposium on Distributed Autonomous Robotic Systems, Ibaraki, Japan, 2008.

[26] A. Hristoskova, C. Agüero, M. Veloso, F. Turck, "Personalized Guided Tour by Multiple Robots through Semantic Profile Definition and Dynamic Redistribution of Participants", 26th AAAI Conference on Artificial Intelligence, Toronto, 2012.

[27] S. Mokarizadeh, A. Grosso, M. Matskin, P. Kungas, A. Haseeb, "Applying Semantic Web Service Composition for Action Planning in Multi-Robot Systems", 4th International Conference on Internet and Web Applications and Services, Stockholm, 2009, pp. 370-376.