Research with Collaborative Unmanned Aircraft Systems

P. Doherty J. Kvarnström F. Heintz D. Landen
P-M. Olsson
Department of Computer and Information Science
Linköping University, SE-58183 Linköping, Sweden
{patdo,jonkv,frehe,davla,perol}@ida.liu.se

Abstract

We provide an overview of ongoing research which targets development of a principled framework for mixed-initiative interaction with unmanned aircraft systems (UAS). UASs are now becoming technologically mature enough to be integrated into civil society. Principled interaction between UASs and human resources is an essential component in their future uses in complex emergency services or bluelight scenarios. In our current research, we have targeted a triad of fundamental, interdependent conceptual issues: delegation, mixed-initiative interaction and adjustable autonomy, that is being used as a basis for developing a principled and well-defined framework for interaction. This can be used to clarify, validate and verify different types of interaction between human operators and UAS systems both theoretically and practically in UAS experimentation with our deployed platforms.

1 Introduction

In the past decade, the Unmanned Aircraft Systems Technologies Lab[1] at the Department of Computer and Information Science, Linköping University, has been involved in the development of autonomous unmanned aerial vehicles and associated hardware and software technologies [13, 11, 12]. The size of our research platforms range from the RMAX helicopter system [14, 39, 36, 33, 7] (Figure 1) developed by Yamaha Motor Company, to smaller micro-size rotor based systems such as the LinkQuad (Figure 2)[2] and LinkMAV [23, 34] (Figure 1), in addition to a fixed wing platform, the PingWing [8] (Figure 1). The latter three have been designed and developed by the Unmanned Aircraft Systems Technologies Lab. Previous work has focused on the development of robust autonomous systems for UAV's which seamlessly integrate control, reactive and deliberative capabilities that meet the requirements of hard and soft realtime constraints [14, 32]. Additionally, we have focused on the development and integration of many high-level autonomous capabilities studied in the area of cognitive robotics such as task planners [15, 16], motion planners [37, 36, 38], execution monitors [18], and reasoning systems [19, 17, 31], in addition to novel middleware frameworks which support such integration [27, 29, 30]. Although research with individual high-level cognitive functionalities is quite advanced, robust integration of such capabilities in robotic systems which meet real-world constraints is less developed but essential to introduction of robotic systems into society in the future.

[1] `www.ida.liu.se/~patdo/aiicssite1/`
[2] `www.uastech.com`

Dagstuhl Seminar Proceedings 10081
Cognitive Robotics
http://drops.dagstuhl.de/opus/volltexte/2010/2630

Consequently, our research has focused, not only on such high-level cognitive functionalities, but also on integrative issues.

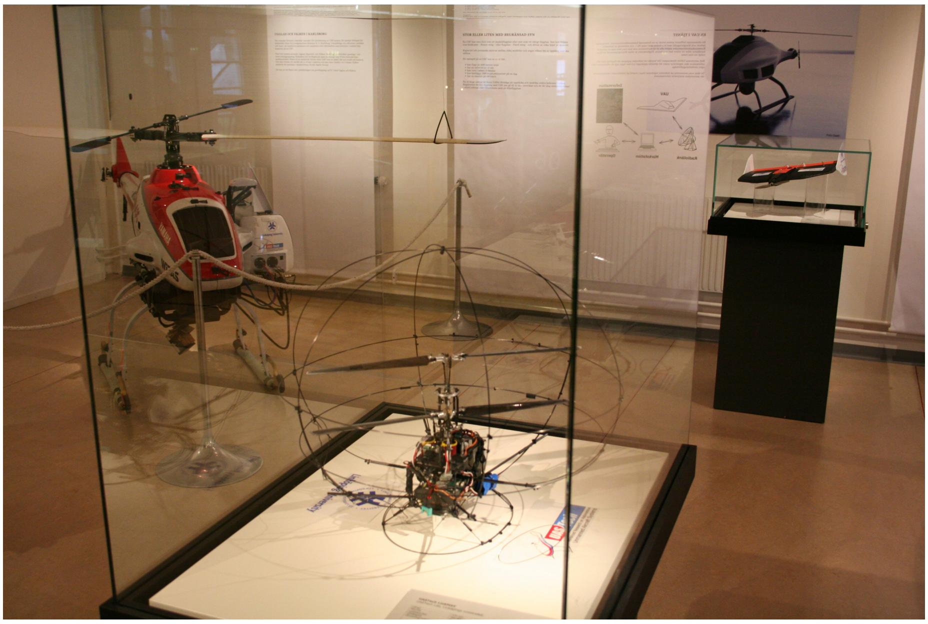

Figure 1: The UASTech RMAX (left), LinkMAV (center) and the PingWing (right)

More recently, our research efforts have begun to focus on applications where heterogeneous UASs are required to collaborate not only with each other but also with diverse human resources [20, 21, 28]. UASs are now becoming technologically mature enough to be integrated into civil society. Principled interaction between UASs and human resources is an essential component in the future uses of UASs in complex emergency services or bluelight scenarios. Some specific target UAS scenario examples are search and rescue missions for inhabitants lost in wilderness regions and assistance in guiding them to a safe destination; assistance in search at sea scenarios; assistance in more devastating scenarios such as earthquakes, flooding or forest fires; and environmental monitoring.

As UASs become more autonomous, mixed-initiative interaction between human operators and such systems will be central in mission planning and tasking. By mixed-initiative, we mean that interaction and negotiation between a UAS and a human will take advantage of each of their skills, capacities and knowledge in developing a mission plan, executing the plan and adapting to contingencies during the execution of the plan. In the near future, the practical use and acceptance of UASs will have to be based on a verifiable, principled and well-defined interaction foundation between one or more human operators and one or more autonomous systems. In developing a principled framework for such complex interaction between UASs and humans in complex scenarios, a great many interdependent conceptual and pragmatic issues arise and need clarification both theoretically, but also pragmatically in the form of demonstrators. Additionally, an iterative research methodology is essential which combines foundational theory, systems building and empirical testing in real-world applications from the start.

Figure 2: The UASTech LinkQuad Quadrotor Helicopter

2 A Conceptual Triad

In our current research, we have targeted a triad of fundamental, interdependent conceptual issues: delegation, mixed-initiative interaction and adjustable autonomy. The triad of concepts is being used as a basis for developing a principled and well-defined framework for interaction that can be used to clarify, validate and verify different types of interaction between human operators and UAS systems both theoretically and practically in UAS experimentation with our deployed platforms. The concept of delegation is particularly important and in some sense provides a bridge between mixed-initiative interaction and adjustable autonomy.

Delegation – In any mixed initiative interaction, humans request help from robotic systems and robotic systems may request help from humans. One can abstract and concisely model such requests as a form of delegation, $Delegate(A, B, task, constraints)$, where A is the delegating agent, B is the potential contractor, $task$ is the task being delegated and consists of a goal and possibly a plan to achieve the goal, and $constraints$ represents a context in which the request is made and the task should be carried out.

Adjustable Autonomy – In solving tasks in a mixed-initiative setting, the robotic system involved will have a potentially wide spectrum of autonomy, yet should only use as much autonomy as is required for a task and should not violate the degree of autonomy mandated by a human operator unless agreement is made. One can begin to develop a principled means of adjusting autonomy through the use of the *task* and *constraint* parameters in the $Delegate(A, B, task, constraints)$ function. A task delegated with only a goal and no plan, with few constraints, allows the robot to use much of its autonomy in solving the task, whereas a task specified as a sequence of actions and many constraints allows only limited autonomy.

Mixed-Initiative Interaction – Mixed-initiative interaction involves a very broad set of issues, both theoretical and pragmatic. One central part of such interaction is the ability of a a ground operator (GOP) to be able to delegate tasks to a UAS, $Delegate(GOP, UAS, task, constraints)$ and in a symmetric manner, the ability of a UAS to be able to delegate tasks to a GOP, $Delegate(UAS, GOP, task, constraints)$. Issues pertaining to safety, security, trust, etc., have to be dealt with in the interaction process and can be formalized as particular types of constraints associated with a delegated task. Additionally, the task representation must be highly flexible, distributed and dynamic. Tasks need to be delegated at varying levels of abstraction and also expanded and modified as parts of tasks are recursively delegated to different UAS agents. Consequently, the structure must also be distributable.

3 A First Iteration

3.1 The Architecture

Our RMAX helicopters use a CORBA-based distributed architecture [14]. For our experimentation with collaborative UASs, we view this as a legacy system and extend it with what is conceptually an additional outer layer in order to leverage the functionality of JADE [24]. "JADE (Java Agent Development Framework) is a software environment to build agent systems for the management of networked information resources in compliance with the FIPA specifications for interoperable multi-agent systems." [25]. The reason for this is pragmatic. Our formal characterization of the *Delegate()* operator is as a speech act. We also use speech acts as an agent communication language and JADE provides a straightforward means for integrating the FIPA ACL language which supports speech acts with our existing systems. The outer layer may be viewed as a collection of JADE agents that interface to the legacy system. We are currently using four agents in the outer layer:

1. **Interface agent** - This agent is the clearinghouse for communication. All requests for delegation and other types of communication pass through this agent. Externally, it provides the interface to a specific robotic system.
2. **Delegation agent**- The delegation agent coordinates delegation requests to and from other UAS systems, with the executor, Resource and Interface agents. It does this essentially by verifying that the pre-conditions to a $Delegate()$ request are satisfied.
3. **Executor agent** - After a task is contracted to a particular UAS, it must eventually execute that task relative to the constraints associated with it. The Executor agent coordinates this execution process.
4. **Resource agent** - The Resource agent determines whether the UAS of which it is part has the resources and ability to actually do a task as a potential contractor. Such a determination may include the invocation of schedulers, planners and constraint solvers in order to determine this.

3.2 Semantic Perspective: Delegation as a Speech Act

In [4, 26], Falcone & Castelfranchi provide an illuminating, but informal discussion about delegation as a concept from a social perspective. Their approach to delegation builds on a BDI model of agents, that is, agents having beliefs, goals, intentions, and plans [6], but the specification lacks a formal semantics for the operators used. Based on intuitions from their work, we provided a formal characterization of their concept of strong delegation using a communicative speech act with pre- and post-conditions which update the belief states associated with the delegator and contractor, respectively [21]. In order to formally characterize the operators used in the definition of the speech act, we used KARO [35] to provide a formal semantics. The KARO formalism is an amalgam of dynamic logic and epistemic / doxastic logic, augmented with several additional (modal) operators in order to deal with the motivational aspects of agents.

First, we define the notion of a task as a pair consisting of a goal and a plan for that goal, or rather, a plan and the goal associated with that plan. Paraphrasing Falcone & Castelfranchi into KARO terms, we consider a notion of strong/strict delegation represented by a speech act S-Delegate(A, B, τ) of A delegating a task $\tau = (\alpha, \phi)$ to B, where α is a possible plan and ϕ is a goal. It is specified as follows:

S-Delegate(A, B, τ), where $\tau = (\alpha, \phi)$

Preconditions:

(1) $Goal_A(\phi)$

(2) $Bel_A Can_B(\tau)$ (Note that this implies $Bel_A Bel_B(Can_B(\tau)))$

(3) $Bel_A(Dependent(A, B, \alpha))$

Postconditions:

(1) $Goal_B(\phi)$ and $Bel_B Goal_B(\phi)$

(2) $Committed_B(\alpha)$.

(3) $Bel_B Goal_A(\phi)$

(4) $Can_B(\tau)$ (and hence $Bel_B Can_B(\tau)$, and by (1) also $Intend_B(\tau)$)

(5) $MutualBel_{AB}$(“the statements above” $\wedge$ $SociallyCommitted(B, A, \tau)$)

Informally speaking this expresses the following: the preconditions of the S-delegation act of A delegating task τ to B are that (1) ϕ is a goal of delegator A (2) A believes that B can (is able to) perform the task τ (which implies that A believes that B himself believes that he can do the task!) (3) A believes that with respect to the task τ he is dependent on B.

The postconditions of the delegation act mean: (1) B has ϕ as his goal and is aware of this (2) he is committed to the task (3) B believes that A has the goal ϕ (4) B can do the task τ (and hence believes it can do it, and furthermore it holds that B intends to do the task, which was a separate condition in F&C’s set-up), and (5) there is a mutual belief between A and B that all preconditions and other postconditions mentioned hold, as well as that there is a contract between A and B, i.e. B is socially committed to A to achieve τ for A. In this situation we will call agent A the delegator and B the contractor.

Typically a social commitment (contract) between two agents induces obligations to the partners involved, depending on how the task is specified in the delegation action. This dimension has to be added in order to consider how the contract affects the autonomy of the agents, in particular the contractor’s autonomy. We consider a few relevant forms of delegation specification below.

3.2.1 Closed vs Open delegation

Falcone & Castelfranchi furthermore discuss the following variants of task specification:

- closed delegation: the task is completely specified: both goal and plan should be adhered to.
- open delegation: the task is not completely specified: either only the goal has to be adhered to while the plan may be chosen by the contractor, or the specified plan contains 'abstract' actions that need further elaboration (a 'sub-plan') to be dealt with by the contractor.

So in open delegation the contractor may have some freedom to perform the delegated task, and thus it provides a large degree of flexibility in multi-agent planning, and allows for truly distributed planning.

The specification of the delegation act in the previous subsection was in fact based on closed delegation. In case of open delegation α in the postconditions can be replaced by an α', and τ by $\tau' = (\alpha', \phi)$. Note that the fourth clause, viz. $Can_B(\tau')$, now implies that α' is indeed believed to be an alternative for achieving ϕ, since it implies that $Bel_B[\alpha']\phi$ (B believes that ϕ is true after α' is executed). Of course, in the delegation process, A must agree that α' is indeed viable. This would depend on what degree of autonomy is allowed.

This particular specification of delegation follows Falcone & Castelfranchi closely. One can easily foresee other constraints one might add or relax in respect to the basic specification resulting in other variants of delegation [5, 10].

3.2.2 Strong Delegation in Agent Programming

When devising a system like the one we have in mind for our scenario, we need programming concepts that support delegation and in particular the open variant of delegation. In the setting of an agent programming language such as 2APL [9], we may use plan generation rules to establish a contract between two agents. Very briefly, a 2APL agent has a belief base, a goal base, a plan base, a set of capabilities (basic actions it can perform), and sets of rules to change its bases: PG rules, PR rules and PC rules. PG-rules have the form $\gamma \leftarrow \beta \mid \pi$, meaning that if the agent has goal γ and belief β then it may generate plan π and put it in its plan base. PR rules can be used to repair plans if execution of the plan fails: they are of the form $\pi \leftarrow \beta \mid \pi'$, meaning that if π is the current plan (which is failing), and the agent believes β then it may revise π into π'. PC-rules are rules for defining macros and recursive computations. (We will not specify them here.)

The act of strong delegation can now be programmed in 2APL by providing the delegator with a rule

$$\phi \leftarrow Can_B(\tau) \wedge Dependent(A, B, \tau) \mid SDelegate(A, B, \tau)$$

(where $\tau = (\alpha, \phi)$), which means that the delegation act may be generated by delegator A exactly when the preconditions that we described earlier are met. The action $SDelegate(A, B, \tau)$ is a communication action requesting to adapt the goal and belief bases of B according to the KARO specification given earlier, and should thus, when successful (depending upon additional assumptions such as that there is some authority or trust relation between A and B), result in a state where contractor B has ϕ in its goal base, $Goal_A(\phi)$, $Can_B(\tau)$ and $MB(\text{`}contract')$ in its belief base, and plan α in its plan base. That is to say, in the case of a closed delegation specification. If the specification is an open delegation, it instead will have an alternative plan

α' in its plan base and a belief $Can_B(\alpha', \phi)$ in its belief base. It is very important to note that in the case of such a concrete setting of an agent programmed in a language such as 2APL, we may provide the Can-predicate with a more concrete interpretation: $Can_B(\alpha, \phi)$ is true if (either ϕ is in its goal base and α is in its plan base already, or) B has a PG-rule of the form $\phi \leftarrow \beta \mid \alpha'$ for some β that follows from B's belief base, and the agent has the resources available for executing plan α. This would be a concrete interpretation of the condition that the agent has the ability as well as the opportunity to execute the plan!

3.3 Pragmatic Perspective: Delegation as Contract Networks and Constraints

From a semantic perspective, delegation as a speech act provides us with insight and an abstract specification which can be used as a basis for a more pragmatic implementation on actual UAS platforms. There is a large gap between these perspectives though. We have chosen to also work from a bottom-up perspective and have developed a prototype software system that implements a delegation framework using the JADE architecture specified above. The system has been tested using a number of complex collaborative scenarios described later in the paper.

In the software architecture, we have focused on a number of issues central to making such a system work in practice:

- **Task Specification** – A specification for a task representation which we call *task specification trees*. This representation has to be implicitly sharable, dynamically extendable, and distributed in nature. Such a task structure is passed from one agent to another and possibly extended in more detail as delegation process is invoked recursively among agents and humans. If the delegation process is successful, the resulting shared structure is in fact executable in a distributed manner. The delegation agents associated with specific UASs are responsible for passing and extending such structures in order to meet the requirements of goal specifications or instantiations of abstract task specifications. The Executor agents associated with specific UASs have the capacity to execute specific nodes in a shared task specification tree that have been delegated to them.
- **Ability and Resource Allocation** – A central precondition to the Delegate speech act specified previously is whether a potential contracting agent *can* do a task. This involves a UAS determining whether it has the proper resources both statically (sensors) and dynamically (use of sensors, power, fuel), and whether it can schedule execution of the processes required to achieve a task at a particular time. In essence, a pragmatic grounding of the $Can()$ predicate in the architecture is required. We are approaching this problem through the use of a combination of distributed constraint solving and loosely coupled distributed task planning. The resource agents associated with specific UASs are responsible for reasoning about resources and solving constraint problems when queried by the associated UAS's delegation agent.
- **Collaborative Planning** – UASs which have accepted a delegated task are responsible for insuring that they can put together a team of UASs which can consistently contribute to the solution of the task. This involves recursive calls to the delegation process, local generation of sub-plans which achieve specific aspects of a task and the integration of these sub-plans into a coherent global plan which can be executed consistently and in a distributed manner to achieve the task requirements. For this we have been developing extensions to TALplanner which combine forward chaining with partial-order planning.

- **Delegation process** - The delegation process itself involves the use of speech acts and contract networks in combination. The process of delegation is quite complex in that a task specification tree has to be constructed dynamically in time and then executed in a distributed manner while meeting all the constraints specified in recursive delegation calls. This also has to be done in a tractable manner in order to ensure that temporal and spatial constraints are met.

These topics are work in progress and will be presented in more detail in future work. A prototype implementation does exist and is being used for experimentation with a number of complex collaborative UAS scenarios briefly described in the next section.

4 Collaborative Scenarios

We have chosen two relatively complex collaborative UAS scenarios in which to develop our mixed-initiative framework.

4.1 An Emergency Services Scenario with Logistics

On December 26, 2004, a devastating earthquake of high magnitude occurred off the west coast off Sumatra. This resulted in a tsunami which hit the coasts of India, Sri Lanka, Thailand, Indonesia and many other islands. Both the earthquake and the tsunami caused great devastation. During the initial stages of the catastrophe, there was a great deal of confusion and chaos in setting into motion rescue operations in such wide geographic areas. The problem was exacerbated by shortage of manpower, supplies and machinery. The highest priorities in the initial stages of the disaster were searching for survivors in many isolated areas where road systems had become inaccessible and providing relief in the form of delivery of food, water and medical supplies. Similar real-life scenarios have occurred more recently in China and Haiti where devastating earthquakes have caused tremendous material and human damage.

Let's assume for a particular geographic area, one had a shortage of trained helicopter and fixed-wing pilots and/or a shortage of helicopters and other aircraft. Let's also assume that one did have access to a fleet of autonomous unmanned helicopter systems with ground operation facilities. How could such a resource be used in the real-life scenario described?

Leg I In the first part of the scenario, it is essential that for specific geographic areas, the UAS platforms should cooperatively scan large regions in an attempt to identify injured persons. The result of such a cooperative scan would be a saliency map pinpointing potential victims, their geographical coordinates and sensory output such as high resolution photos and thermal images of potential victims. The resulting saliency map would be generated as the output of such a cooperative UAS mission and could be used directly by emergency services or passed on to other UASs as a basis for additional tasks.

Leg II In the second part of the scenario, the saliency map generated in Leg I would be used as a basis for generating a logistics plan for several of the UASs with the appropriate capabilities to deliver food, water and medical supplies to the injured identified in Leg I. This of course would also be done in a cooperative manner among the platforms.

Leg I of this mission has been flown using two RMAX helicopters flying autonomously and using the prototype software system described above. The output of the mission is a saliency map

Figure 3: Identified bodies from Leg I of the emergency services scenario

Figure 4: Emergency Supply Delivery

with geo-located injured humans and infrared and digital photos of the injured (Figure 3). Leg II of this mission has been tested in hardware in the loop simulation (Figure 4). Initial work with this scenario is reported in [21, 22, 33].

4.2 A UAS Communication Relay Scenario

A wide variety of applications of UASs include the need for surveillance of distant targets, including search and rescue operations, traffic surveillance and forest fire monitoring as well as law enforcement and military applications. In many cases, the information gathered by a surveillance UAS must be transmitted in real time to a base station where the current operation is being coordinated. This information often includes live video feeds, where it is essential to achieve uninterrupted communication with high bandwidth. UAS applications may therefore require line-of-sight communications to minimize quality degradation, which is problematic in urban or mountainous areas. Even given free line of sight, bandwidth requirements will also place strict restrictions on the maximum achievable communication range. These limitations are particularly important when smaller UASs are used, such as the 500-gram battery-powered LinkMAV (Figure 1), or the LinkQuad (Figure 2). In these situations, transmitting information directly to the base station can be difficult or impossible.

Both intervening obstacles and limited range can be handled using a chain of intermediate *relay UASs* passing on information from the surveillance UAV to the base station (Figure 5). A surveillance UAV can then be placed freely in a location that yields information of high quality. We are therefore interested in positioning relay UASs to maximize the quality of the resulting chain, given a known target position. However, we are also interested in minimizing resource

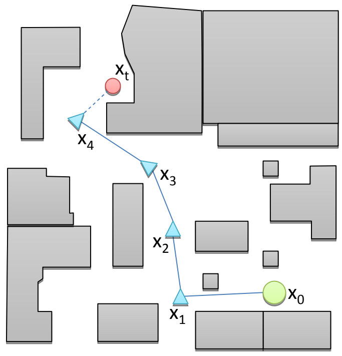

Figure 5: UAVs at x_1, x_2 and x_3 are acting as relays in a city, connecting the base station at x_0 with the surveillance UAV at x_4, surveilling the target at x_t.

usage in terms of the number of relay UAVs required. Given these two objectives, a variety of trade-offs are possible. For example, decreasing the distance between adjacent relay UASs may improve transmission quality but instead requires additional relays.

In [1, 3, 2], we have developed a number of graph search algorithms which are scalable and efficient approximations to continuous bi-objective optimization problems and applicable to relay positioning in discrete space. Using these algorithms as a basis, we are experimenting in simulation with multiple platforms which combine the use of such algorithms with our collaborative framework. A human operator or a contracted UAS will set up such a relay and delegate subparts of the relay mission to other UASs in the area. Currently, we are experimenting with hardware-in-the-loop simulations. An example of one the environments used in testing is an urban environment with semi-random placement of 100 tall buildings, as shown in Figure 6. To reduce clutter, the figure is based on a sparse discretization and shows only the "lowest" level of grid cells. Figure 7 shows a ground operation interface used to generate UAS communication relays.

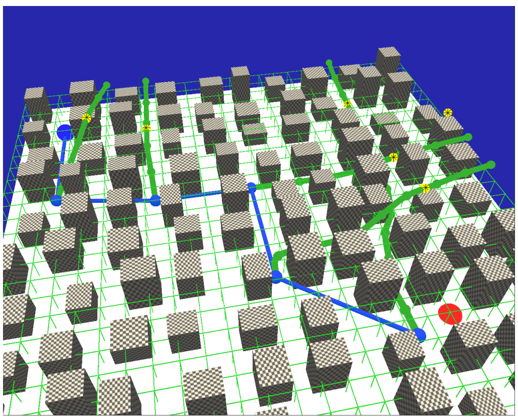

Figure 6: Randomized urban environment.

Figure 7: Ground operator interface for generating UAS communication relays

5 Conclusions

The gap between research done in cognitive robotics and pragmatic use of such results in real-life deployed systems embedded in highly dynamic outdoor environments is currently quite large. The research pursued and described in this paper is intended to take steps toward closing this gap by developing theories in a more traditional manner from the top-down as exemplified by the formal characterization of delegation as a speech act, but also by building demonstrators and prototype software systems which deal with all the complexities associated with systems and architectures constrained to operate in dynamic outdoor environments. This type of research demands an iterative and open-ended approach to the problem by combining theory, engineering and application in suitable doses, and continually trying to close the loop at early stages in the research. We will continue to pursue this approach and hope to report additional details in the near future.

Acknowledgements

This overview of research was written specifically for the 2010 Dagstuhl Workshop on Cognitive Robotics, Feb 22-26, 2010. It is intended to summarize work in progress which is jointly supported by grants from the Swedish Research Council (VR), the Swedish Foundation for Strategic Research (SSF) Strategic Research Center MOVIII, the Swedish Research Council (VR) Linnaeus Center CADICS, ELLIIT Excellence Center at Linköping-Lund for Information Technology and the Center for Industrial Information Technology (CENIIT). The work described in Section 3.2 was done in cooperation with Professor J-J. Ch. Meyer, Utrecht University.

References

[1] O. Burdakov, P. Doherty, K. Holmberg, J. Kvarnström, and P-M. Olsson. Positioning unmanned aerial vehicles as communication relays for surveillance tasks. In *Proceedings of the 5th Robotics: Science and Systems Conference (RSS)*, 2009.

[2] O. Burdakov, P. Doherty, K. Holmberg, J. Kvarnström, and P-M. Olsson. Relay positioning for unmanned aerial vehicle surveillance. *International Journal of Robotics Research*, 2010.

[3] O. Burdakov, P. Doherty, K. Holmberg, and P-M. Olsson. Optimal placement of UV-based communications relay nodes. *Journal of Global Optimization*, 2010.

[4] C. Castelfranchi and R. Falcone. Toward a theory of delegation for agent-based systems. In *Robotics and Autonomous Systems*, volume 24, pages 141–157, 1998.

[5] P. Cohen and H. Levesque. Teamwork. *Nous, Special Issue on Cognitive Science and AI*, 25(4):487–512, 1991.

[6] P.R. Cohen and H.J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(3):213–261, 1990.

[7] G. Conte and P. Doherty. Vision-based unmanned aerial vehicle navigation using geo-referenced information. *EURASIP Journal of Advances in Signal Processing*, 2009.

[8] G. Conte, M. Hempel, P. Rudol, D. Lundström, S. Duranti, M. Wzorek, and P. Doherty. High accuracy ground target geo-location using autonomous micro aerial vehicle platforms. In *Proceedings of the AIAA-08 Guidance,Navigation, and Control Conference*, 2008.

[9] M. Dastani and J.-J. Ch. Meyer. A practical agent programming language. In *Proceedings of AAMAS07 Workshop on Programming Multi-Agent Systems (ProMAS2007)*, pages 72–87, 2007.

[10] E. Davis and L. Morgenstern. A first-order theory of communication and multi-agent plans. *Journal Logic and Computation*, 15(5):701–749, 2005.

[11] P. Doherty. Advanced research with autonomous unmanned aerial vehicles. In *Proceedings on the 9th International Conference on Principles of Knowledge Representation and Reasoning*, 2004. Extended abstract for plenary talk.

[12] P. Doherty. Knowledge representation and unmanned aerial vehicles. In *Proceedings of the IEEE Conference on Intelligent Agent Technolology (IAT 2005)*, 2005.

[13] P. Doherty, G. Granlund, K. Kuchcinski, E. Sandewall, K. Nordberg, E. Skarman, and J. Wiklund. The WITAS unmanned aerial vehicle project. In *Proceedings of the 14th European Conference on Artificial Intelligence*, pages 747–755, 2000.

[14] P. Doherty, P. Haslum, F. Heintz, T. Merz, T. Persson, and B. Wingman. A distributed architecture for intelligent unmanned aerial vehicle experimentation. In *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems*, 2004.

[15] P. Doherty and J. Kvarnström. TALplanner: A temporal logic based forward chaining planner. *Annals of Mathematics and Artificial Intelligence*, 30:119–169, 2001.

[16] P. Doherty and J. Kvarnström. TALplanner: A temporal logic based planner. *Artificial Intelligence Magazine*, Fall Issue 2001.

[17] P. Doherty and J. Kvarnström. Temporal action logics. In V. Lifschitz, F. van Harmelen, and F. Porter, editors, *The Handbook of Knowledge Representation*, chapter 18, pages 709–757. Elsevier, 2008.

[18] P. Doherty, J. Kvarnström, and F. Heintz. A temporal logic-based planning and execution monitoring framework for unmanned aircraft systems. *Journal of Automated Agents and Multi-Agent Systems*, 19(3):332–377, 2009.

[19] P. Doherty, W. Łukaszewicz, and A. Szałas. Approximative query techniques for agents with heterogenous ontologies and perceptual capabilities. In *Proceedings on the 7th International Conference on Information Fusion*, 2004.

[20] P. Doherty, W. Łukaszewicz, and A. Szałas. Communication between agents with heterogeneous perceptual capabilities. *Journal of Information Fusion*, 8(1):56–69, January 2007.

[21] P. Doherty and J-J. Ch. Meyer. Towards a delegation framework for aerial robotic mission scenarios. In *Proceedings of the 11th International Workshop on Cooperative Information Agents*, 2007.

[22] P. Doherty and P. Rudol. A UAV search and rescue scenario with human body detection and geolocalization. In *20th Australian Joint Conference on Artificial Intelligence (AI07)*, 2007.

[23] S. Duranti, G. Conte, D. Lundström, P. Rudol, M. Wzorek, and P. Doherty. LinkMAV, a prototype rotary wing micro aerial vehicle. In *Proceedings of the 17th IFAC Symposium on Automatic Control in Aerospace*, 2007.

[24] G. Caire F. Bellifemine and D. Greenwood. *Developing Multi-Agent Systems with JADE.* John Wiley and Sons, Ltd, 2007.

[25] G. Caire F. Bellifemine, F. Bergenti and A. Poggi. JADE – a Java agent development framework. In J. Dix R. H. Bordini, M. Dastani and A. Seghrouchni, editors, *Multi-Agent Programming - Languages, Platforms and Applications.* Springer, 2005.

[26] R. Falcone and C. Castelfranchi. The human in the loop of a delegated agent: The theory of adjustable social autonomy. *IEEE Transactions on Systems, Man and Cybernetics–Part A: Systems and Humans*, 31(5):406–418, 2001.

[27] F. Heintz and P. Doherty. DyKnow: A knowledge processing middleware framework and its relation to the JDL fusion model. *Journal of Intelligent and Fuzzy Systems*, 17(4), 2006.

[28] F. Heintz and P. Doherty. DyKnow federations: Distributing and merging information among UAVs. In *Eleventh International Conference on Information Fusion (FUSION-08)*, 2008.

[29] F. Heintz, J. Kvarnström, and P. Doherty. A stream-based hierarchical anchoring framework. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2009.

[30] F. Heintz, J. Kvarnström, and P. Doherty. Bridging the sense-reasoning gap: DyKnow - stream-based middleware for knowledge processing. *Journal of Advanced Engineering Informatics*, 24(1):14–25, 2010.

[31] M. Magnusson, D. Landen, and P. Doherty. Planning, executing, and monitoring communication in a logic-based multi-agent system. In *18th European Conference on Artificial Intelligence (ECAI 2008)*, 2008.

[32] T. Merz, P. Rudol, and M. Wzorek. Control System Framework for Autonomous Robots Based on Extended State Machines. In *Proceedings of the International Conference on Autonomic and Autonomous Systems*, 2006.

[33] P. Rudol and P. Doherty. Human body detection and geolocalization for uav rescue missions using color and thermal imagery. In *IEEE Aerospace Conference*, 2008.

[34] P. Rudol, M. Wzorek, G. Conte, and P. Doherty. Micro unmanned aerial vehicle visual servoing for cooperative indoor exploration. In *Proceedings of the IEEE Aerospace Conference*, 2008.

[35] B. van Linder W. van der Hoek and J.-J. Ch. Meyer. An integrated modal approach to rational agents. In M. Wooldridge and A. Rao, editors, *Foundations of Foundations of Rational Agency*, volume 14 of *Applied Logic Series*. An Integrated Modal Approach to Rational Agents, 1998.

[36] M. Wzorek, G. Conte, P. Rudol, T. Merz, S. Duranti, and P. Doherty. From motion planning to control – a navigation framework for an unmanned aerial vehicle. In *Proceedings of the 21st Bristol International Conference on UAV Systems*, 2006.

[37] M. Wzorek and P. Doherty. Reconfigurable path planning for an autonomous unmanned aerial vehicle. In *Proceedings of the 16th International Conference on Automated Planning and Scheduling*, pages 438–441, 2006.

[38] M. Wzorek, J. Kvarnström, and P. Doherty. Choosing path replanning strategies for unmanned aircraft systems. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2010.

[39] M. Wzorek, D. Landen, and P. Doherty. GSM technology as a communication media for an autonomous unmanned aerial vehicle. In *Proceedings of the 21st Bristol International Conference on UAV Systems*, 2006.