FOI, Swedish Defence Research Agency, is a mainly assignment-funded agency under the Ministry of Defence. The core activities are research, method and technology development, as well as studies conducted in the interests of Swedish defence and the safety and security of society. The organisation employs approximately 1250 personnel of whom about 900 are scientists. This makes FOI Sweden's largest research institute. FOI gives its customers access to leading-edge expertise in a large number of fields such as security policy studies, defence and security related analyses, the assessment of various types of threat, systems for control and management of crises, protection against and management of hazardous substances, IT security and the potential offered by new sensors.

FOI
Defence Research Agency
Defence and Security Systems and Technology
SE-164 90 Stockholm

Phone: +46 8 555 030 00
Fax: +46 8 555 031 00

www.foi.se

Survey of patrolling algorithms for surveillance UGVs

DAVID A. ANISI, JOHAN THUNBERG

FOI-R--2266--SE Scientific report
ISSN 1650-1942 April 2007

Defence and Security, Systems and Technology

David A. Anisi, Johan Thunberg

Survey of patrolling algorithms for surveillance UGVs

FOI-R--2266--SE
ISSN-1650-1942
Scientific report
April 2007
Defence and Security, Systems and Technology

Issuing organisation	Report number, ISRN	Report type
FOI – Swedish Defence Research Agency Defence and Security, Systems and Technology SE-164 90 STOCKHOLM	FOI-R--2266--SE	Scientific report
	Research area code Strike and Protection	
	Month year April 2007	**Project no.** E28004
	Sub area code Ground Vehicles	
	Sub area code 2	
Author/s (editor/s) David A. Anisi, Johan Thunberg	**Project manager** Petter Ögren	
	Approved by Helena Bergman	
	Sponsoring agency FMV, 297316-LB704859	
	Scientifically and technically responsible Xiaoming Hu	

Report title
Survey of patrolling algorithms for surveillance UGVs

Abstract

This survey centers around the Cooperative Task Assignment and Path Planning problem, denoted C–TAPP. A vast amount of applications from the surveillance and security field can be formulated as C–TAPP problems. The purpose of this survey is to give a comprehensive description of current research relevant to the C–TAPP problem, as well as forming a basis for future research.

Initially, to set the stage for the subsequent discussions, two motivating examples taken from the domain of surveillance and security is presented and a more formal definition of the C–TAPP problem is given. Following that, a broad sampling of the research that is currently ongoing in this field is provided. Key features that are present in the overwhelming majority of the papers encountered in this survey are listed. Also, some important research aspects that have been largely untreated in the literature, are mentioned. A representative group of existing commercial systems are presented and their capabilities and shortcomings are discussed. In the last part of this survey, reviews of some individual papers can be found.

Keywords
Autonomy, Cooperative, Surveillance, Security, Path Planning, Task Assignment, UGV

Further bibliographic information	**Language** English
ISSN ISSN-1650-1942	**Pages** 58 p.
	Price According to price list

Utgivare
FOI – Totalförsvarets forskningsinstitut
Försvars- och säkerhetssystem
164 90 STOCKHOLM

Rapportnummer, ISRN
FOI-R--2266--SE

Klassificering
Vetenskaplig rapport

Forskningsområde
Bekämpning och skydd

Månad år
April 2007

Projektnummer
E28004

Delområde
Fordonsteknik

Delområde 2

Författare/redaktör
David A. Anisi, Johan Thunberg

Projektledare
Petter Ögren

Godkänd av
Helena Bergman

Uppdragsgivare/kundbeteckning
FMV, 297316-LB704859

Tekniskt och/eller vetenskapligt ansvarig
Xiaoming Hu

Rapportens titel
Litteraturöversikt rörande autonoma algoritmer för patrullerande spaningsfordon

Sammanfattning
Denna rapport är en litteraturstudie över kooperativ uppgiftstilldelning och banplanering. Säkerhet- och övervakningsdomänen är ett av detta problems främsta applikationsområden. Målet med litteraturstudien är att ge en översikt över befintliga forskningsresultat, samt att forma en bas inför framtida forskning.

Nyckelord
autonomi, samverkan, övervakning, säkerhet, banplanering, uppgiftstilldelning, UGV

Övriga bibliografiska uppgifter

Språk Engelska

ISSN ISSN-1650-1942

Antal sidor: 58 s.

Distribution **enligt missiv**

Pris: **Enligt prislista**

Contents

1 Acronyms

AP	Assignment Problem
C–TAPP	Cooperative Task Assignment and Path Planning
GA	Genetic Algorithms
GAP	General Assignment Problem
IP	Integer program
LIP	Linear Integer Program
LP	Linear Program
MILP	Mixed Integer Linear Program
MRTA	Multi-Robot Task Allocation
$\mathcal{NP}$	Non-deterministic Polynomial time
OAP	Optimal Assignment Problem
TSP	Traveling Salesman Problem
UAV	Unmanned Air Vehicle
UGV	Unmanned Ground Vehicle
(M)VRP	(Multiple) Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows
VRPTWSD	Vehicle Routing Problem with Time Windows and Split deliveries

2 Introduction and Definition of Scope

In both civil and military applications, surveillance is performed in order to assist in the prevention, detection and monitoring of intrusion, theft or other safety-related incidents. Application areas and facilities that require such supervision are innumerous and include airport facilities, military installations, border-lines, storage buildings, harbors, power plants, banks, factories and offices.

Todays surveillance and security solutions are based on a combination of

- human guards (manned gates, airport screeners, store detectives),
- electronic systems (cameras, intrusion alarms, fire detection),
- physical security (fences, gates),
- software (reporting, verification, logging).

In the ideal case, surveillance should be performed in a continuous manner and cover the entire facility, although in practice, financial and head-count constraints limit it to only encompass the most important and critical areas. Recent scientific and technological developments is however taking us towards more autonomous and mobile solutions. The market for semi-autonomous sentry vehicles is in fact already established and growing. As for today, there are a few tailor-made safety and security vehicles on the market, but so far, they possess a quite limited functionality and capabilities (see Section 3). From a performance standpoint, the potential benefits with adopting a security or surveillance UGV are numerous and well documented:

1. cost savings,
2. humans are removed from direct exposure to potentially harmful situations,
3. autonomous systems can perform many security and surveillance routines more effectively than humans since the don't get bored and thereby inattentive during long working hours,
4. autonomous systems don't participate in ”inside jobs”.

Here-below, in order to concretize a few problem instances, two fictitious motivating examples are described and discussed in more detail. This is the scope of Sections 2.1 and 2.2. Following that, we start Section 2.3 by arguing that these two considered examples are in fact instances of the same problem class, namely *cooperative task assignment and path planning for multi vehicle systems*. The remaining part of Section 2.3 is therefore devoted to formally defining this base-line problem.

2.1 Motivating Example: Constrained rounds of patrol

To assist in the prevention and detection of safety-related incidents, it is customary to perform patrolling rounds. Traditionally, patrolling rounds are performed by humans. To obtain maximal security, these rounds should be performed in a continuous manner. However, having a full-time guard 24 hours a day, 7 days a week will cost more than 3 full-time employees.

This fact has prevented EsCoTer AB, a medium size company in Stockholm, from obtaining this high level of security. EsCoTer is an importer and distributer of Asian scooters, ATVs and dirt-bikes into the Swedish market. As such, it has a warehouse that has repeatedly been an object of interest for intruders and burgles. Faced with this problem, the owner of the company has therefore looked for alternative solutions to complement the traditional way of patrolling. The most flexible and cost effective offer so far has been delivered by a security company called Sentry Inc. and involves using a small group of semi-autonomous vehicles for performing these patrolling rounds.

The most basic solution Sentry Inc. could provide was to engage a group of sentry UGVs that cooperatively visit a set of known and predefined sites in a regular and repetitive manner. On their way between the sites, each UGV is to survey its surrounding by using its on-board sensors. Possible onboard sensors include laser scanners, IR-cameras and chemical sensors with which one can detect *e.g.* intruders, fire, gas leaks or even abnormal radioactivity. As a more refined solution for more challenging scenarios, Sentry Inc. provides a solution that can handle patrol rounds which are *constrained* to fulfill certain conditions. According to the specifications, possible constraints are:

Temporal and/or spatial visiting constraints: It might for instance be desirable to assure that sensitive sites of high priority are visited at least once during given time intervals. This imposes a *temporal* constraint on the solution. Sensors with limited field of view provide a prototype example of *spatial* constraints.

Line of sight constraints: In addition to visiting the sites, the threat situation may call for monitoring of the UGVs themselves. It is therefore of interest to have the capability to perform the patrolling rounds while mutually keeping the line of sight between given UGVs clear.

Non-predictability constraints: Performing the rounds in a regular manner, makes it easy for potentially hostile forces to plan their actions and circumvent this line of defense. Therefore, it is of interest to introduce some degree of non-predictability in the patrolling rounds.

Verifiability constraints: These are introduced as to attain a quality certification, *i.e.* assurance of certain levels on key features. This might for instance involve guarantees that the non-deterministic rounds will not neglect any site completely.

The owner of EsCoTer realizes that the second alternative suits the needs of his company better and therefore signs a two year contract with Sentry Inc. within a matter of weeks.

2.2 Motivating Example: Rapid search and localization of possible intruders.

At 3 o'clock in the morning, the intrusion alarm suddenly goes off. Due to budget and personnel constraints, there is only one guard in duty at this time of the night. Previous experience has shown that in these situations, one must not leave the main security console unattended. Therefore the guard in duty delegates a mission to his team of semi-autonomous sentry UGVs.

More precisely, they are required to join forces with those UGVs that are already performing continuous patrolling rounds (see Section 2.1) and first of all, determine if the alarm is false, and if not, search for and localize the possible intruder(s). Upon detection, some of the UGV team members are expected to cooperatively surround and monitor the intruder. The remaining vehicles are assigned to determine the extend of the threat by monitoring sensitive points of interest and searching for cronies to the intruder. During the entire mission, the UGVs provide the guard on duty with relevant decision data. With this information at hand, he can make decisions regarding suitable and graded actions, *e.g.* requiring backup.

2.3 Problem Definition: Cooperative task assignment and path planning for surveillance and security applications

In this section, we start by recognizing the inter-connection between the two examples of Sections 2.1 and 2.2. More precisely, we argue that they are two instances of the same problem class, namely the problem of Cooperative Task Assignment and Path Planning (C–TAPP). To see this, notice that in the first example, visiting any of the sites can be seen as a task and the goal is to plan the paths for all the UGVs such that these tasks are performed in an optimal manner while fulfilling the constraints. The second example (Section 2.2) consists of a number of subsequent tasks (joining forces with the other UGVs, determining if the alarm is false, searching the environment for possible intruders *etc.*) that require the planning of optimal paths. Also in this example, it is possible to impose various constraints on the problem formulation.

Next, to set a common ground for the subsequent sections, a more formal definition of the C–TAPP problem is given.

Problem 1 (Cooperative Task Assignment and Path Planning) *Given N UGVs and M tasks that they have to perform, assign the tasks and find paths for all the vehicles in a cooperative manner. The task assignment and the generated paths are to fulfill all the constraints imposed on them while minimizing a given cost function.*

Notice that this problem formulation allows some of the N vehicles to remain in their initial position and do nothing. This could be of great strategical interest since the inactive vehicles can be used for performing other missions in parallel.

Remark 2.1 *Obviously one of the keywords in Problem 1 is* cooperation*. Surprisingly, concrete definitions of the meaning of this term within the multi-vehicle field are sparse in the literature. For now, we define* cooperation *from an optimization perspective: "Cooperation emerges from the objective of minimizing the given cost function"*[1]*. In Section 3 however, we follow [9] and provide a list of some alternative definitions of cooperation.*

Before presenting various choices of objective function and constraints in C–TAPP, we make a small digression to put the C–TAPP problem into perspective by presenting our view on how C–TAPP enters the overall system

[1]By this we are also able to distinguish *cooperation* from *coordination*, which can be thought of as an implication of the constraints of an optimization problem.

architecture. In this survey, an overall *modular* design of the system architecture is assumed. Figure 2.1 depicts a rather classical way of decomposing the overall problem of designing a multi-UGV system, where only parts of the interactions between the modules are counted for. The essence of Problem 1

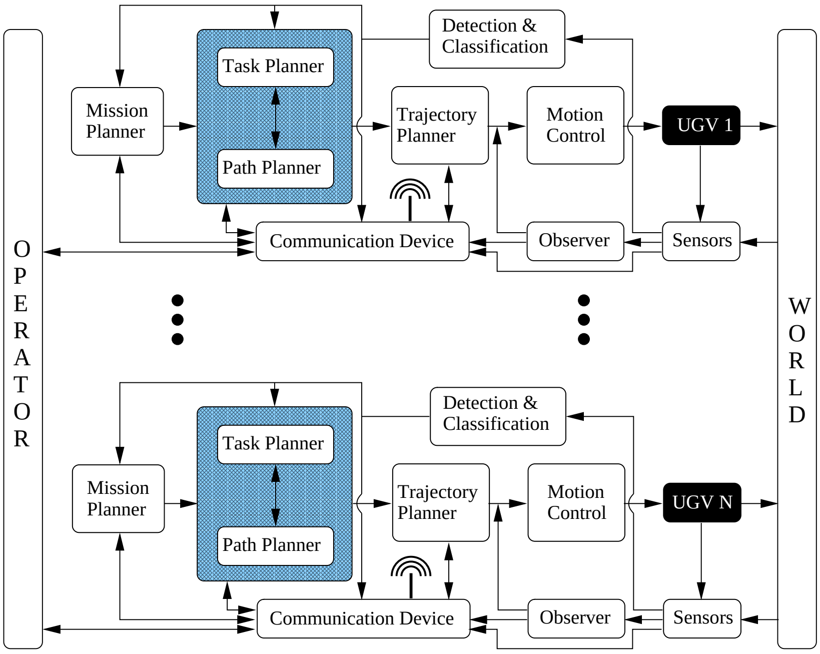


Figure 2.1: Possible modular structure of a multi-UGV system.

is that the principal interest of this survey lies on the task assignment and path planning module of Figure 2.1. In particular, methods for *cooperative* task- and path planning methods will be in focus. Hence, several other crucial subproblems, such as, trajectory planning, actuator control, observer design, sensor fusion, communication technology issues, sensor detection, *etc.* will not be discussed. We assume these modules are available to us, so that for instance the vehicles are assumed to know their positions, either through direct measurements from on-board sensors, or through a suitably deigned observer.

Having put the C–TAPP problem into perspective, we proceed by listing the most relevant choices of objective function and constraints in C–TAPP. This list particularly emphasizes C–TAPP problems for surveillance and security applications.

1. **Possible objective functions:**

   a) Minimize the total time for completing the tasks[2].

   b) Minimize the distance traveled while performing the tasks.

   c) Minimize the maximal or accumulative threat encountered during the mission.

   d) Minimize a combination of the previous objectives.

   e) Minimize a combination of task completion time and number of vehicles used.

   f) Provide soft ordering by associating revenues to all tasks and maximizing the total revenue (*cf.* [25, 40, 7]).

[2] Assumption on constant vehicle speed gives an equal work-load formulation.

g) Maximize total utility as defined by the difference between total cost and total revenue [12, 25].

Remark 2.2 *Problem 1 is a generalization of the Traveling Salesmen Problem (TSP) and is therefore also $\mathcal{NP}$-hard. Consequently, we can not expect to solve all problem instances to optimality within a reasonable amount of time. In practice, some heuristic algorithm may be used for solving Problem 1. A possible constraint on the solution would then be:*

- *to obtain solutions whose cost are within a certain factor from the optimal one,* i.e. *are ε-optimal.*

2. **Possible task assignment constraints:**

   a) No tasks are to be neglected.

   b) Every task should be assigned to one and only one UGV.

   c) Number of tasks assigned to each vehicle is upper bounded.

   d) A task, m, must have at least $n_m > 1$ UGVs assigned to it.

   e) Different UGVs have different capabilities so that not all vehicles can perform all tasks.

   f) Temporal constraints such as:
      - time-windows for the completion of certain tasks,
      - ordering, *e.g.* that task X has to be performed before task Y.

   g) Non-predictability.

3. **Possible paths planning constraints:**

   a) Spatial constraints such as:
      - upper bound on the total path length for some of the vehicles (fuel constraint).
      - given/free initial and/or final positions for some of the vehicles.
      - collision free paths.

      In addition to these, one might consider other spatial constraints imposed by such tasks which cannot be solved if:
      - the distance to it is larger than a given threshold,
      - the task is approached from certain directions.

      Both these examples are highly relevant for camera surveillance scenarios.

   b) Line of sight/communication maintenance constraints.

   c) Dynamically feasible (*i.e.* the needs of the trajectory planner is addressed, *cf.* Figure 2.1).

A challenging and highly relevant extension to the C–TAPP problem is to explicitly recognize the presence of *uncertainty*. In the face of measurement noise, parametric uncertainty, modeling errors and other disturbances, the deterministic nature of Problem 1 falls short. One approach in the literature to handle this issue is to pose the C–TAPP problem within a stochastic or robust optimization framework. To reasonably limit its scope, the main focus of this survey will however be on a more implicit approach to handle the uncertainty issue, namely: by requiring the solution method to be of relatively low computational complexity, we will be able to solve the problem repeatedly. Hence, as new information about the environment or mission objectives is gathered while the mission unfolds, online re-planning can be performed. This way, feedback is incorporated and a certain degree of robustness is obtained.

3 Survey of Current Research and Existing Systems

A vast amount of research and a huge number of publications have been devoted to problem formulations more or less related to the C–TAPP problem, as defined in Section 2.3. In this section a broad overview of the research that is currently ongoing in this field is provided. The exposition is neither complete nor self-contained, hence appropriate external references are provided in order to allow the interested reader to probe more deeply into the subject.[1] At the end of this section, we also provide a short list of some commercial UGV systems tailored for surveillance and security applications.

Current Research

Initially, a concise description of the state of current C–TAPP related research is provided. Then, some important issues that deserve particular treatment follows.

In essence, C–TAPP related research has so far been quite informal, concept-oriented and primarily focused on

1. specification of particular problem instances. Often, this is done with some real-world application in mind.

2. presentation of some heuristic, empirical or *ad hoc* solution method, *e.g.* a proper coordination and cooperation architectures and different problem decomposition techniques.

3. validation of the proposed solution method through simulations or experiments in a proof-of-concept fashion.

These three steps are the foundation of the overwhelming majority of the papers in this survey. The first step involves specification of the objective function to be minimized and relevant constraints imposed on the task- and path planner. As previously mentioned, this step normally is inspired from a particular application-domain. As an illustrative example, reference [33] considers a scenario where a group of N vehicles are required to visit M known target locations within a hostile environment with P static threats. The objective function consists of a combination of risk minimization, balancing the workloads between the vehicles and minimizing the mission completion time. As for the task- and path-wise constraints, the authors require all M targets to be visited, avoiding collisions and flying within predefined length limits (fuel constraint). To make the problem more realistic, the authors may further impose timing and ordering constraints on the tasks as well as an upper limit on the number of targets that can be assigned to each vehicle.

[1] Reviews of some of the individual papers can be found in Chapter 4.

As for the second step, the literature includes a wide variety of techniques and ideas. This is also usually where the main research focus lies. A classical approach to solve challenging combinatorial optimization problems such as C–TAPP is based on clustering [37, 12]. The two main ideas here are to either cluster-first-route-second or the other way around. In order to improve solution quality, the clustering and routing phases can be repeated; at the expense of computational load. As an example, reference [33] approximates an exact MILP formulation of C–TAPP in four different ways using the clustering ideas presented here-above. These approximations have lower computational complexity and are therefore better suited for online purposes. The approach taken by Maddula *et al.* [39] illustrate another distinguished way of tackling the C–TAPP problem. In a first phase, an initial assignment is constructed. In a second phase, this initial assignment is refined using four target exchange operators that are defined in the paper. The same idea is elaborated upon in several other papers encountered in this survey. Heuristic ways of improving an initially feasible solution include

- Tabu search [40, 2] (which are known to perform well on various routing problems [36, 10, 37]),
- stochastic hill-climbing [26],
- ant colony optimization [22],
- genetic algorithms [30].

It is important to mention that there is a natural way of decomposing the C–TAPP problem into two subproblems:

1. the optimal task assignment problem,
2. the optimal path planning problem.

Unless the objective function in the task assignment problem is *path independent*, this modular scheme is bound to produce sub-optimal solutions. Having a path independent objective function is hardly the case in most realistic surveillance and security applications. Consequently, the ideal case from this survey's point of view, would be to solve these two subproblems *concurrently*. This is however beyond current reach for all but few problem instances. The following fact serves as a partial explanation to this: As indicated in Remark 2.2, the C–TAPP problem can be viewed as a generalization of the well-known Traveling Salesmen Problem (TSP). To see this, consider the obstacle free, single vehicle case ($N = 1$). Let further the M tasks coincide with the "cities" that the "salesmen" has to visit before returning to its initial position. The objective will be to minimize the total tour length. The obstacle free environment implies that an ordering of the cities also serve as a feasible path for the vehicle. Since TSP is one of Karp's 21 original $\mathcal{NP}$-complete problems [32], one cannot solve all C–TAPP problem instances to optimality within a reasonable amount of time. It is therefore customary to solve the two subproblems of task assignment and path planning in an iterative manner. This approach was depicted in Figure 2.1.

Since the field of cooperative multi-vehicle systems is a relatively young research domain, some important aspects of C–TAPP have been largely untreated in the literature. In particular, the following two aspects deserve much more attention from the research community:

1. more theoretical aspects and frameworks for formal analysis [31, 25],
2. evaluative and comparative studies [34, 36].

As indicated in Remark 2.1, concrete definitions of the meaning of the key term "coordination" are sparse in the multi-vehicle literature. Next, we follow [9] and provide a list of some alternative definitions of this term. Explicit definitions of cooperation include:

1. "joint collaborative behavior that is directed toward some goal in which there is a common interest or reward"
2. "a form of interaction, usually based on communication"
3. "[joining] together for doing something that creates a progressive result such as increasing performance or saving time"

This last definition is probably the one closest to the definition provided in Remark 2.1: "Cooperation emerges from the objective of minimizing the given cost index". This definition originates from an optimization perspective. Also, as mentioned earlier, this point of view allows us to distinguish *cooperation* from *coordination*, which can be thought of as something emerging from the constraints of an optimization problem.

In the literature, there exists a body of work that aims at providing a suitable classification scheme and taxonomy for the field of cooperative multi-robotics (see *e.g.* [9, 14, 15, 25, 31]). These papers also provide excellent surveys of the literature at different times. Next, a handful of selected topics from these important papers will be discussed.

In [14] the authors present a taxonomy that classifies cooperative teams. Seven important aspects are mentioned and include collective size, the systems communication and computational capabilities. A summary of the proposed taxonomic axis can be found in Table 3.1. It can be noted that the communication issue constitutes a relatively large fraction of the classification dimensions. In addition, a rather comprehensive survey of existing work as it appeared in

Taxonomic Axis	Description
Collective Size	The number of robots in the group
Coll. Reconfigurability	Rate for spatial re-organization
Coll. Composition	Group being homogeneous or heterogeneous
Comm. Range	Upper limit on the inter-robot distance such that communication is still possible
Comm. Topology	Describes possible inter-robot communication
Comm. Bandwidth	Amount of information that can be transmitted
Processing Ability	Each units model of computation

Table 3.1: Summary of the taxonomic axis as they appear in [14].

the mid 90's is provided in [14]. In order to illustrate the usefulness of the suggested taxonomy, [14] sorts the surveyed papers according to their position in the taxonomy.

Another important work that provides natural dimensions along which multi-robot systems can be separated is [9]. In this paper, the authors identify five important "research axis" or taxonomic axis that can be used when comparing different system designs:

Group Architecture: this axis can be described as the "infrastructure upon which collective behaviors are implemented". Concepts such as group differentiation (homogeneity/ heterogeneity)[2], control type (centralization/ decentralization) and communication structure fall into this category.

[2]Which corresponds to the "collective composition" axis of [14].

Resource Conflict: strategy for resolving possible group conflicts, *e.g.* the collision avoidance problem in mobile robotics or the multi-access problem in computer networks.

Origin of Cooperation: (biologically/socially inspired) mechanisms that motivate and achieve cooperation in systems where this has not been "explicitly engineered" into the system.

Learning: strategies for finding correct values for design parameters, *e.g.* reinforcement learning, genetic algorithms or neural networks.

Geometric Problems: issues tied to the embedding of the system in a two- or three-dimensional world. Examples include multi-robot path planing and moving to formation.

Also in [9], the authors provide a survey of existing work, and further discuss some open research problems, technological constraints and the influence of other academic disciplines that have shaped the field of cooperative robotics. The reader is urged to consult [9] for a fuller discussion. It must be emphasized however that the task assignment problem is largely overlooked in [9]. From the C–TAPP's point of view, the task decomposition and allocation method certainly requires an axis on its own.

A possible classification of different coordination schemes is that of explicit *vs.* implicit coordination [16, 31][3]. A multi-vehicle team may coordinate *explicitly* using communication or negotiations. An example of one such mechanism is market-based coordination [12], where individual vehicles competitively bid for the tasks to be performed. This auction-based approach is based on some given bidding rule [46]. However, multi-vehicle teams may also cooperate *implicitly*. In this case, communication is mediated through inter-vehicle and vehicle-world interactions. This type of communication is called *stigmergic* in the biological literature [4]. As an example, a box-pushing application is considered in [13] that achieves "cooperation without communication". This is possible since the object being manipulated also functions as a "communication channel" shared by all the robots. The relative merit of these two coordination schemes remains an open question. According to [31] however, it is in general easier to perform a formal analysis on explicit approaches. They are also considered to produce more accurate and near-optimal solutions. On the downside, explicit coordination schemes are not as flexible, robust and – due to the inherent computation and communication complexity – scalable as the implicit approach.

Another feature than can be used for classifying different architectures is whether the system is centralized or not [9]. In centralized systems, the decisions regarding cooperation and coordination are made at one single central control unit. Decentralized systems on the other hand, are characterized by the lack of such a unit. Instead, robots rely solely on locally available and processed knowledge. As far as pros and cons are considered, decentralized systems are generally considered to be inherently more reliable, robust and scalable [12, 31]. In reality however, there is a continuum of possible system designs that span the spectrum between the two extreme cases. Market-based approaches serve as a typical example that resides in the middle of the spectrum.

In the literature, there are few papers that explicitly recognize the presence of *uncertainty*. To reasonably limit its scope, the main focus of this survey is on a more implicit approach to handle the uncertainty issue, namely requiring fast solutions to the C–TAPP problem. The low computational time allows

[3]This can also be referred to as intentional *vs.* emergent coordination [24, 15].

us to perform re-planning online as new information about the environment or mission objectives is gathered. This information can then be processed and fed back regularly to the C–TAPP planner. The reader should however be aware of the important work and significant progresses that have been made in explicitly incorporating uncertainty in the problem formulation. Stochastic or robust versions of problems related to C–TAPP have been considered in *e.g.* [1, 7, 19, 20, 26], all of which have been reviewed in Section 4.

Existing Commercial Systems

As for today, there are some tailor-made safety and security vehicles on the market. Below, a handful of such enterprise-ready systems, together with their key features are presented. In essence, todays vehicles can be described as well-equipped sensor platforms capable of performing a number of low-level, single-step tasks: *i.e.* if X occurs, do Y. This includes recording video and audio, taking digital photography, sounding an alarm or even releasing a dense smokescreen to frighten off an intruder. Hence, so far the existing systems possess quite limited functionality and capabilities. What is really needed to take this to the next level is the challenging task of generalizing the X and Y to more complicated and advanced high-level missions. It should also be emphasized that, except for Rotundus' product, the listed vehicles are mainly intended for interior applications. Further expansion to outdoor environments is however foreseeable in the future.

PatrolBot^{TM} from MobileRobots Inc.

MobileRobots Inc.[4] 30,000$ Security PatrolBot has been called the "first fully autonomous robotic surveillance and monitoring system available off-the shelf"[5]. Some of the key capabilities of the PatrolBot are

Figure 3.1: Security PatrolBot. Image courtesy of MobileRobots Inc.

- open door detection (with your choice of alert responses),
- detection of motion and smoke,

[4]Formerly ActivMedia Robotics, `http://www.mobilerobots.com/PatrolBot.html`
[5]See the entertaining PatrolBot blog at `http://patrolbot.blogspot.com/`

- able to follow response, so you can lead vehicle to its work site,
- route scheduler, including day/night, holiday, weekday and weekend schedules with randomizing capabilities,
- remote video capability from low-light, pan-tilt-zoom and/or omnidirectional-camera,
- 2-way audio communication to relay live spoken warnings from a security guard in the main security console, or simply play pre-recorded audio messages,
- instant camera zoom to a point you click,
- snapshots on demand, sent to your storage system,
- automated response sends the vehicle immediately to a point of intrusion for verification from your integrated alarm system,
- (optional) door and elevator operation with Wi-Fi controls,
- (optional) card-readers and/or iris scanners to ask selected by-passers to identify themselves,
- (optional) hazard sensing and other custom sensors.

Despite the 10% maintenance fee per year, there is a rather quick return of investment associated with PatrolBot: typically 3-6 months on a 24/7 position and 12 months on a part-time basis. PatrolBot is equipped with a 24V lead-acid battery with a run time of 3.5hrs. Recharge time (with a high-capacity charger) is 4hrs.

CyberGuard® from Cybermotion Inc.

Starting out in mid 80's, Cybermotion Inc.[6] produced the only commercially available security vehicle in the world at that time. The first generation CyberGuard called SR2 was introduced in 1990, based on the companies three-wheel synchro-drive K2A base, which had been commercially available for research applications since 1984. The introduction of the upgraded six-wheel-drive K3A platform vehicle resulted in the CyberGuard SR3 platform in 1996. This is the platform shown in Figure 3.2, which is also equipped with the Enhanced Sensor Package (ESP). The ESP sensor suite provides:

- high speed pan and tilt camera able to localize, track, record and report suspicious events such as flame or intruders. Note that the video transmitter and the associated voice channel are optional add ons,
- ultrasonic intrusion detector,
- microwave intrusion radar (K-band >25GHz),
- optical flame detector to detect fires at an early stage,
- standard ionization sensors for smoke detection,
- several gas sensors able to monitor CO (carbon monoxide), CH4 (methane), C2H6O (ethanol), C3H8 (propane), and several other substances,
- temperature and humidity sensors,

[6] `http://www.cybermotion.com`

- (optional) non-contact puddle detector (low power laser tuned to the absorption wavelength of water) to reliably detect puddles of water or even damp carpets,
- (optional) optical pyrodetector measures the temperature of objects at a safe distance and may detect hot doors and electrical panel boxes (an indication of possible fire behind them) or hot motors (abnormal operation),
- (optional) auxiliary gas sensors such as for oxygen or even nerve gases,
- (optional) infrared illuminator.

Figure 3.2: The CyberGuard SR3 based on the K3A platform. Image courtesy of Cybermotion Inc.

From this comprehensive list of sensor capabilities, it is clear the the CyberGuard SR3/ESP platform represents one of the most diverse commercial product of today. Note however that a superior sensor capability not necessarily is a testimony of a vehicle's level of autonomy or efficiency.

When contacting the company regarding price information and image courtesy, we learned that the CyberGuard line is currently out of production and there is no replacement.

Rotundus AB

Uppsala-based Rotundus AB, was formed in 2004 as a spin off from the idea of utilizing spherical robots for planetary exploration. The ball shaped robot is extremely rugged and durable. It is large enough to handle rough outdoor terrain such as snow, mud, sand or water. Figure 3.3 is a testimony of this. Referring to Figure 3.4, the robot is propelled by means of a pendulum placed inside the shell. By moving the pendulum in either direction, the center of mass

Figure 3.3: Rotundus' spherical robot handles rough outdoor environments elegantly. Image courtesy of Rotundus AB.

gets shifted and the sphere starts rolling in that particular direction. Using this technology, the robot is able to travel with speeds up to 10 km/h.

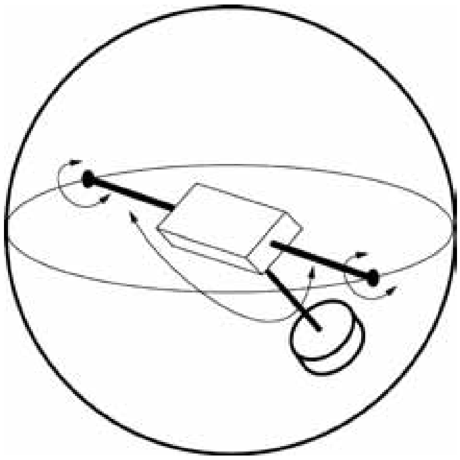


Figure 3.4: Rotundus generates movement by shifting the center of mass of the pendulum inside the spherical shell. Image courtesy of Rotundus AB.

As far as the sensor placement is considered, the developers have found a natural place, namely where the main horizontal axis meets the shell (see Figure 3.5). This way, all vital components are enclosed within the robot shell. Two VGA cameras with 10x optical zoom constitutes the robot's main sensor capacity and provide it with a 360° field of view.

(a) (b)

Figure 3.5: Image courtesy of Rotundus AB.

RoboSentry™ from CCS Robotics

Based on the PatrolBot of MobileRobots Inc., the robotics division of Cypress Computer Systems Inc. (CCSRobotics)[7] offers derivative products specifically designed for security applications. Hence, CCSRobotics is a value-added reseller and integrator of MobileRobots' PatrolBot product line. For instance, CCSRobotics have created the 50,000$ RoboSentry Defender with capabilities such as guarding tours, mustering, remote surveillance, and fire prevention.

Figure 3.6: RoboSentry Defender is a derivative product built upon the PatrolBot platform, Image courtesy of CCS Robotics.

Robot X from Secom Co. Ltd.

Secom Co., Ltd.[8], a Tokyo-based company better known for supplying human security guards, has developed a six-wheeled surveillance robot called Robot X. It can either patrol on a pre-define route (as indicated by a magnetic guide line)

[7] `http://www.ccsrobotics.com`
[8] `http://www.secom.co.jp/english/`

or be remotely controlled over a Wi-Fi link. The Robot X, which can be seen in Figure 3.7 has the following features:

- chase intruders (at up to 10 km/h),
- take high definition video pictures by means of the omnidirectional-camera mounted on top of the vehicle,
- issue loud warnings: either live spoken warnings from a security guard in a remote command post, or simply play pre-recorded messages,
- detect suspicious fires and start an optional automatic fire extinguisher to stop fires at an early stage,
- release a dense, billowing cloud of smoke from an optional smoke emitter.

The idea with the last mentioned capability is that even if the smokescreen does not frighten off the intruder, it will at least confuse them long enough for a human guard to get to the scene.

(a) Since 2005, Secom's six-wheeled sentry vehicle, Robot X, can be rented for a price of 2600$ per month.

(b) The smokescreen of Robot X is meant to frighten off, or at least confuse, the intruder.

Figure 3.7: Image courtesy of PC Watch Japan.

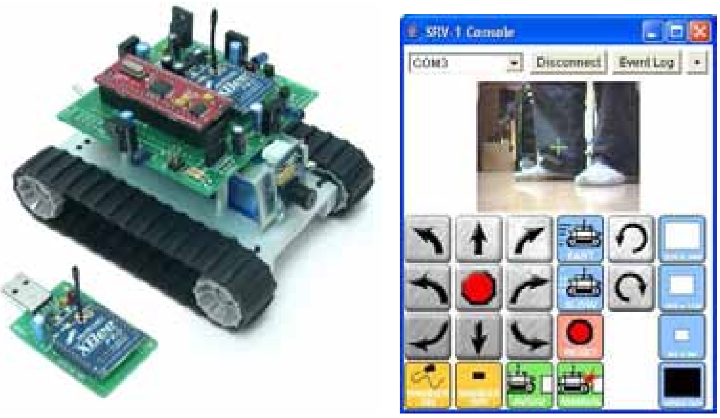
Unlike the aforementioned sentry vehicles, Robot X will however not be for sale. Secom plans to rent out the robots at ¥300,000 a month (2600$) which is half the cost of hiring a human security guard to do the same job. However, an extra ¥4.5 million (38 000$) is also required to build a patrol route. This high cost is partly explained by the fact that the route must be provided on a pavement with a magnetic guide line. Robot X then follows this magnetic line using a build-in magnetic guide sensor positioned at the bottom of the vehicle. The battery can run Robot X for about two hours when the robot continuously travels at 3-4 km/h. In practice, it is therefore necessary to locate battery chargers along the patrol route. To secure the communication needs, multiple WLAN base stations are also scattered along the patrol route.

SRV-1 from ThinkGeek Inc.

Think Geek Inc.[9] are selling a budget surveillance vehicle, namely the SRV-1 which costs 375$. The included software is completely open source on both the host computer end and the vehicle firmware. According to their homepage, the SRV-1 product features include:

[9] http://www.thinkgeek.com

- drive vehicle manually via web browser with live video feed (see Figure 3.8(b)),
- built in proximity sensors can be toggled on or off to assist when driving the vehicle manually,
- archive video on demand or via schedule,
- control access to vehicle and video feed via multiple user accounts,
- roving mode allows autonomous exploration with basic vision detection,
- wireless control up to 300 feet from host computer,
- fully open source and programmable.



(a) The SRV-1 vehicle provided by ThinkGeek Inc.

(b) Screen-shot of the SRV-1 open source software.

Figure 3.8: Image courtesy of ThinkGeek Inc.

Hence, SRV-1 is more appropriately classified as a remote controlled vehicle which nevertheless could be of interest as a base for developing more sophisticated and autonomous capabilities.

SeQ-1 from ITRI/SKS

SeQ-1 is the most recent member of the family of the surveillance UGVs. It is developed by the Taiwan government-sponsored Industrial Technology Research Institute[10] (ITRI) through cooperation with Taiwan Shin Kong Security Co., Ltd. (SKS). SeQ-1 is tailored for security monitoring purposes. It is not available for the commercial market yet but will debut at the 2007 SecuTech Expo that takes place in Taipei during April 16-18, 2007. Hence, little information about it can be found at the time of writing. According to an article in Digtime[11], SKS will trial SeQ-1 in office and commercial buildings and will then be responsible for marketing the vehicle. In addition, the following is stated:

"Through 360 panoramic surveillance, SeQ-1 is able to detect any emergencies, such as invaders, smoke emissions and fire outbreaks, and then immediately trace such targets, issue an alert and send on-the-spot video to monitoring centers, ITRI pointed out. The intelligence of SeQ-1 enables it to establish

[10] http://www.itri.org.tw
[11] http://www.digitimes.com/news/a20070413PD203.html

electronic maps for planning automatic patrols and avoiding collisions, ITRI indicated."

Having seen a number of commercial surveillance vehicles, we now turn to the reveiews of the individual papers.

4 Short Reviews of Individual Papers

In this section, reviews of a number of individual papers is presented.

4.1 A formal analysis and taxonomy of task allocation in multi-robot systems, by G.P. Gerkey and M.J. Mataric

This paper can be found in reference [25].

Problem Formulation

The aim of the article is to provide a domain independent taxonomy of Multi-Robot Task Allocation (MRTA) problems. The authors restrict their attention to the class of problems where the tasks are independent of each other, *i.e.* no ordering of tasks occurs.

Relation to the C–TAPP Problem

A taxonomy of MRTA problems is of course relevant for C–TAPP.

Proposed Solution Method and Mathematical Tools Used

The authors admit that task independence is a strong assumption that clearly limits the scope of the study. Problems where the ordering of the tasks affect the objective function will not be dealt with.

The authors first give an introduction to *utility* and *combinatorial optimization*, tools that are used in MRTA. Utility is described as the difference between two measures. The cost measure of performing a task is subtracted from the quality measure of performing the task. For example if the cost is traveled distance, a vehicle with a high quality measure of a target can be assigned to the target even though there are vehicles closer to the target.

The field of *Combinatorial Optimization* provides a set-theoretic framework, based on subset systems, for describing a wide variety of optimization problems [42].

Definition 4.1 *A subset system (E, F) is a finite set of objects, E, and a nonempty collection of subsets of E, F, that are closed under inclusion. That is, if $X \in F$ and $Y \subseteq X$ then $Y \in F$. The elements of F are called independent sets.*

If each element e_i in E is alloted a utility u_i, *subset maximization* refers to the combinatorial problem associated with the subset system (E, F) and is performed by choosing the independent set $X \in F$ that has the highest possible total utility among the elements in F. The authors proceed by presenting the greedy algorithm [42]:

Algorithm (The Greedy Algorithm).

1. Reorder the elements of $E = \{e_1, e_2, ..., e_n\}$ such that $u(e_1) \geq u(e_2) \geq u(e_n)$.

2. set$X := \emptyset$

3. For $j = 1$ to n:
   if $X \cup \{e_j\} \in F$ then $X = X \cup \{e_j\}$

A subsystem is a matroid if and only if it the Greedy algorithm solves the associated combinatorial utility maximization problem.

Now the authors introduce the taxonomy of MRTA problems (MRTA problems with no ordering constraints of tasks, the tasks can not be coupled). The problems are divided into different problem classes depending on the characteristics of targets and vehicles and how assignments are performed in time. Three axes of separation are presented for describing MRTA problems:

- **single-task vehicles (ST) *vs.* multi-task vehicles (MT)** *i.e.* vehicles that are only able to perform one task *vs.* vehicles that are able to perform multiple tasks.

- **single-robot tasks (SR) *vs.* multi-robot tasks (MR)**, *i.e.* tasks that can be serviced by one vehicle *vs.* tasks that must be serviced by many vehicles.

- **instantaneous assignment (IA) *vs.* time-extended assignment (TA)**. IA means that the available information concerning the vehicles, the tasks, and environment permits only an instantaneous allocation of tasks to robots, with no planning for future allocations. TA means that more information is available, such as the set of all tasks that will need to be assigned or a model of how tasks are expected to arrive over time.

ST-SR-IA This corresponds to an optimal assignment problem (OAP) which can be solved by a centralized linear programming approach in $O(mn^2)$ time (*e.g.* Kuhn's Hungarian method). Various other methods are available that solves the problem in polynomial time.

ST-MR-IA This corresponds to the Set Partitioning Problem (SPP) which is $\mathcal{NP}$-hard. This problem has been studied extensively in the literature and many heuristics exists.

MT-SR-IA The authors argue that these problems are uncommon, and also it is the same problem class as *ST-MR-IA*.

MT-SM-IA This corresponds to the Set Covering Problem (SCP), and is $\mathcal{NP}$-hard. Various heuristics exists.

Time extended assignments These problems are scheduling problems that are $\mathcal{NP}$-hard. Methods are described in the paper for ST-SR-TA, ST-MR-TA, MT-SR-TA and MT-SM-TA.

Mathematical tools used: Combinatorial optimization, linear programming, Computational complexity and heuristic algorithms.

Personal Comments, Pros and Cons, assessment of paper quality

For the somewhat limited class of problems where the ordering of targets does not affect the objective function of the problem that shall be solved, this paper constitute a very good taxonomy. The paper is well written.

4.2 Principled approaches to the design of multi-robot systems, by C. Jones, D. Shell and M.J. Mataric B.P. Gerkey

This paper can be found in reference [31].

Problem Formulation

An overview of three different principled methodologies of distributed Multi-Robot Systems (MRS) is presented. Those are

1. formal analysis of multi-robot task allocation (MRTA),
2. formal MRS controller design methodology,
3. formal approach to large-scale MRS.

Another overview paper of multi-robot systems can be found in [9].

Relation to the C–TAPP Problem

The first part, formal analysis of multi-robot task allocation, is interesting for C–TAPP. Controller design and large-scale systems are not as relevant.

Proposed Solution Method and Mathematical Tools Used

The authors address the concept of explicit and implicit coordination. Explicit coordination occurs when there are explicit task-directed communications or negotiations about global resource usage or task assignments in order to achieve coordinated behavior. Implicit coordination occurs when only robot-robot communication or robot-world interactions are allowed, without an explicit notion of task directed communication or negotiation. In implicit coordination a robot might register where its nearest neighbors are heading, but it cannot receive no more information.

The three methodologies presented can be placed on different positions on an axis between explicit coordination and implicit coordination, see Figure 4.2.

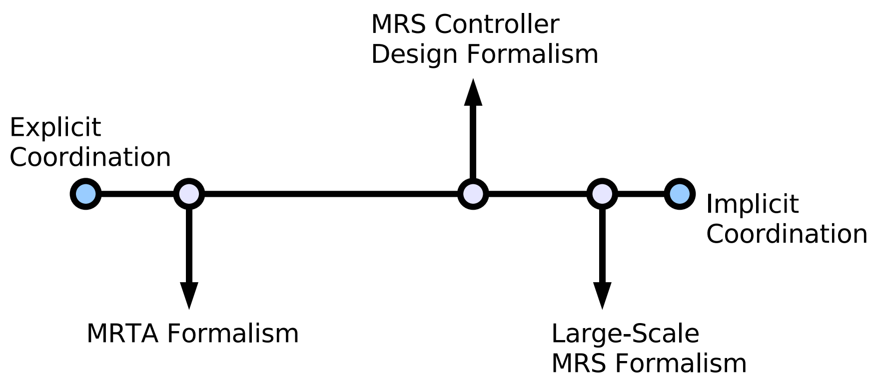


Figure 4.1: Placement of the 3 design methodologies along the explicit *vs.* implicit coordination spectrum, according the type of MRS on which they focus, source [31].

Many of the concepts introduced in MRTA can also be found in [25].

Personal Comments, Pros and Cons, assessment of paper quality

A good survey but not as relevant for C–TAPP as [25].

4.3 Coordinated Target Assignment And Intercept for Unmanned Vehicles, by R.W. Beard and T.W. McLain, M.A. Goodrich and E.P. Anderson

This paper can be found in reference [3].

Problem Formulation

The considered problem is to assign N vehicles to M targets in a hostile environment. Each vehicle can only be assigned to one target, but each target can be assigned to multiple vehicles. This is a set partitioning problem which is $\mathcal{NP}$-hard (see also Section 4.1 or reference [25]). The problem addressed is to do

1. cooperative target assignment,
2. coordinated vehicle intercept,
3. path planning,
4. feasible trajectory generation,
5. asymptotic trajectory following.

Relation to the C–TAPP Problem

Subproblems 1) and 3) are very relevant, while subproblem 2) can be of interest in particular applications.

Proposed Solution Method and Mathematical Tools Used

A system architecture for a single vehicle is provided. The communication manager makes it possible to communicate between vehicles and thereby achieve cooperation. The Task Manager, Target Manager and Intercept Manager are communicating with each other, trying to find a good solution. Since subproblems 4) and 5) are not very relevant to the C–TAPP problem, they will not be accounted for in this review.

Target Manager

The amount of possible assignments are M^N. The authors introduce four objectives in the the assignment:

a) minimize the group path length to the target,

b) minimize the group threat exposure,

c) maximize the number of vehicles prosecuting each target (to maximize survivability),

d) maximize the number of targets visited.

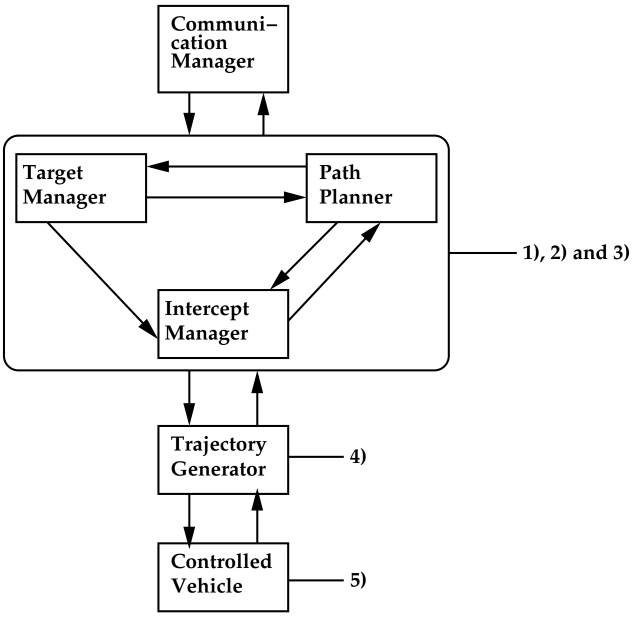


Figure 4.2: System architecture for a single UAV, source [3].

These objectives are referred to as *ShortPath*, *AvoidThreats*, *MaxForce* and *MaxSpread* respectively. The first two objectives are myopic and the two latter ones are team objectives.

Satisficing decision theory is used to address the *ShortPath* and *AvoidThreats* objectives. First a median length cost, $\overline{J}_{length,i}(V_i, T_j)$, and a median threat cost $\overline{J}_{threat,i}(V_i, T_j)$ is computed for each vehicle V_i and target T_j, *i.e.* the k costs are reduced to a median cost for each vehicle-target pair. A normalized acceptability function $\mu_{A_i}(T_j) \in [0, 1]$ and rejectability function $\mu_{R_i}(T_j) \in [0, 1]$ are defined for vehicle V_i in such a way that $\mu_A = 1$ for the closest target and $\mu_A = 0$ for the most distant target. Similarly, the rejectability function is normalized so that $\mu_R = 1$ for the target with the highest risk and $\mu_R = 0$ for the target with the lowest risk. A set of acceptable targets are then created for vehicle V_i as

$$S_{V_i} = \{T_j : \mu_{A_i}(T_j) \geq b_i \mu_{R_i}(T_j)\},$$

where b_i is a selectivity index used to ensure that S_{V_i} has appropriate cardinality.

The two team objectives *MaxForce* and *MaxSpread* are competing objectives. A monotonic function that encodes the value of *MaxForce* is introduced. It strongly rewards teams that are larger than a certain size. The total group value of *MaxForce* is the product of all individual *MaxForce*-values of the targets that have assigned vehicles. This would perhaps lead to that all vehicles are assigned to the same target. To balance this with *MaxSpread* the cardinality of the set of targets with assigned vehicles is multiplied with this group value and we get a function V. The goal is to maximize V, and thereby get a good compromise between *MaxForce* and *MaxSpread*.

Intercept Manager

The Intercept Manager "talks" to the Path Planner and delivers paths and constant velocities for each vehicle, that synchronizes the Time over Target (TOT) times for each vehicle.

Path Planner

A Voronoi diagram is constructed. Each convex polygon cell is either a threat, a target or a start position of a vehicle. From each target position and each start position of the vehicles, straight lines are drawn to the corners of the polygon cell. For each vehicle, the k-best paths are computed to each target, using Eppstein's k-best paths algorithm. The cost of traveling along an edge is a compromise between the length of the edge and the risk of traveling there, *i.e.* for edge i the cost is

$$J_i = \kappa J_{length,i} + (1-\kappa) J_{threat,i}.$$

Mathematical tools used Eppstein's k-best paths algorithms, Voronoi diagram, Satisficing Decision Theory.

Personal Comments, Pros and Cons, assessment of paper quality

A very good paper. It delivers an end-to-end solution, from target assignment via path planning to the controlled vehicle.

4.4 Mission planning for synthetic aperture radar surveillance, by D. Panton and A. Elbers

This paper can be found in reference [41].

Problem Formulation

The authors have published a series of papers on variations of the same theme [41, 29, 30]. In this first paper, the authors consider the problem of optimal (or at least automatic) mission planning for a *single* aerial vehicle performing Synthetic Aperture Radar (SAR) surveillance. The mission objective is to start off from a given base-node, scan a set of given land stripes, or *swaths* (typically no more than 20), and end at a possibly different base-node. Each swath can be scanned from each of its four sides, so that the output of the mission planner should describe the best sequence of swaths, as well as the associated side to be scanned. In addition to this, the solution method should be able to incorporate no-fly zones and mandatory screening of given swath sides.

The considered problem is clearly similar to the well-known Traveling Salesman Problem (TSP), but different because of the way the swaths are scanned. In fact, the above described problem is equivalent to a TSP where we are constrained to visit one city in every given group of four cities. Consequently, since TSP is known to be in the class of $\mathcal{NP}$-complete problems [32, 42], achieving solutions in real operational time may involve adoption of heuristic methods and acception of *satisficing* mission plans, *i.e.* “good enough solutions” that are not necessarily optimal.

Relation to the C–TAPP Problem

This paper solely considers the task assignment part of the C–TAPP problem for *single* vehicle case. Hence, two important aspects of the the C–TAPP problem, namely cooperation and task planning are neglected. Further, the considered problem is to be seen as a subproblem in the regional surveillance problem [30]. By assuming that the swaths are given, this paper does not address how to obtain a suitable/optimal regional cover.

Proposed Solution Method and Mathematical Tools Used

The authors devise a method using integer programming (IP). As in the case with TSP, the by far most distinguished difficulty with such an approach is to ensure that the solution does not contain any sub-tours.

As for handling no-fly zones while flying between different swaths, an approach based on *a priori* pruning of the feasible set and an heuristic post-processing is advocated. This approach is illustrated in Figure 4.3. Upon detection of a path segment that violates the no-fly zone constraint, a local visibility graph is constructed where the two swath sides of interest serve as the source and destination nodes respectively. The intermediate nodes are taken as the corners of the no-fly zones violated. A new path that circumvents the no-fly zones (but may be sub-optimal to the overall problem) is obtained by solving a shortest path problem in the constructed visibility graph.

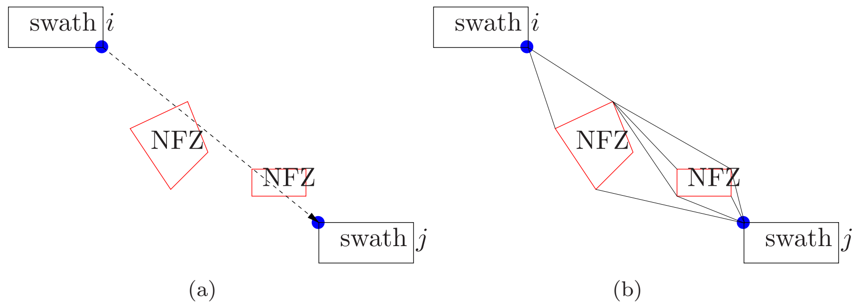


Figure 4.3: Through post-processing, solutions are obtained that do not pass through the non-fly zones.

Mandatory screening of given swath sides are handled by imposing simple equality constraints on the IP problem.

Mathematical tools used: Integer programming (IP), sub-tour elimination methods, visibility graph, shortest path problem.

Personal Comments, Pros and Cons, assessment of paper quality

The mission planner was developed on behalf of the Defence Science and Technology Organization (DSTO), Australia, and showed an average decrease in tour lengths of 15% over those produced intuitively and manually.

4.5 Mission planning for regional surveillance, by M. John, D. Panton and K. White

This paper can be found in reference [30].

Problem Formulation

The authors have published a series of papers on variations of the same theme, [41, 29, 30]. In this paper the authors extend earlier work [41] where the problem of optimal mission planning for a *single* aerial vehicle performing Synthetic Aperture Radar (SAR) surveillance was considered (see Section 4.4). The regional surveillance problem considered in the current paper [30] incorporates an additional spatial characteristic: there are no predefined swaths to be scanned. Hence, the question of how to find an optimal cover of swaths for the region has been added. This introduces a new level of decisions (namely the optimal region cover) and therefore makes this problem an order of magnitude harder to solve. As in their previous paper, the possibility of incorporating no-fly zones should be provided.

Relation to the C–TAPP Problem

The regional surveillance problem considered in this problem is an interesting one for security applications. However, since the *single* vehicle case is the only one accounted for, the important cooperative aspect of the C–TAPP problem is overlooked.

Proposed Solution Method and Mathematical Tools Used

In this paper, two different approaches for the solution of the region surveillance problem, namely an integer programming (IP) model and a genetic algorithm (GA), is considered,. In addition to formulating the problem instances, computational experiment has been carried out on a diverse range of experimental regions in order to investigate and compare the efficiency of both methods.

As far as the IP model is considered, one may notice that the optimal cover may include swaths of arbitrary shape and size. In order to reduce the number of potential swaths, the authors only consider rectilinear swaths within a finite number of discretized sizes. Upon this approximation, the derivation of the IP problem is quite straightforward and can be found in [29, 30]. In essence, the IP model is a TSP-like problem and as such the sub-tour elimination constraints serve as the main obstacle for successful implementation of larger problem instances. The principal reason for considering the IP model has in fact been as a base-line problem to compare the GA solutions with.

GA apply the mechanics of "natural selection" to a population of candidate solutions (or chromosomes), over time. This with the objective of producing increasingly fit individuals. The application of the GA in this paper is based on a permutation of discretized swaths covering the region.

Regarding the computational comparison between the IP and GA formulations, the efficiency is measured as a combination of the total distance traveled and the CPU time required to produce the solution. As might be expected, none of the executions of the GA identified the optimal mission, however satisfying or "good enough" suboptimal missions were produced within a reasonable amount of time. Therefore, as the authors rightly conclude, the GA formulation is *operationally* superior to the IP model.

Mathematical tools used: Integer programming (IP), genetic algorithm (GA).

Personal Comments, Pros and Cons, assessment of paper quality

An intuitive but sub-optimal approach to solve the region surveillance problem would be to adopt a two-stage approach. In the first step, one may find an appropriate set of swaths that cover the entire region. Based on this partition and in a second step, a feasible mission tour can be obtained in accordance with the results of [41]. This procedure can also be performed iteratively.

4.6 Multi-target assignment and path planning for groups of UAVs, by Maddula *et al.*

This paper can be found in reference [39].

Problem Formulation

In this work, the authors consider an environment with N vehicles, M targets and P threats. All the targets and threats are assumed to be stationary and known *a priori*. Additionally, the authors assume that all the vehicle are identical and that there is only one type of task to be performed at each target site.

The goal is then to assign all the targets to the vehicles as to minimize a combination of maximum path length[1], divide work equitably among the fleet of vehicles, and limit the threat faced by each vehicle.

Relation to the C–TAPP Problem

Despite the title of this paper, the problem formulation solely results in *ordered assignment of the targets*, *i.e.* the output of the algorithm is an ordered target set for each vehicle. In particular, no explicit path is provided for the leg between two subsequent targets. Notice that using straight line segments to connect the targets is not a viable option in the case when the vehicles have a limited turning radius or the environment includes obstacles or threats. In general, calculating these sub-paths is an integral part of the C–TAPP problem.

Proposed Solution Method and Mathematical Tools Used

The proposed algorithm has two subsequent phases associated with it.

1. The goal of the first stage is to obtain a *feasible path graph* (FPG), which is passed on to the second phase. The FPG is constructed as follows:

   a) Build a Voronoi tessellation of the environment based on the given threats. This results in the so called *waypoint graph*, $\bar{G}_W$.

   b) From each target, add an edge to the m nearest nodes in $\bar{G}_W$. By also introducing the targets as new nodes, the so called *augmented waypoint graph*, G_W is obtained.

   c) Calculate the risk associated with traversing each edge in G_W by approximating the integral along the edge with a finite sum based on the distance of various points on the edge to nearby threats.

   d) Prune the graph, G_W, by deleting all edges that have a risk higher than the given threshold, thus satisfying one of the objectives by construction. This pruned graph is called the *reduced edge graph* (REG).

   e) Obtain the K shortest paths from each vehicle to all the targets and between all target–pairs within a chosen threshold distance.

   f) Finally, the FPG is obtained by only keeping the vehicle and target nodes in REG. In particular, the so called *waypoint nodes* obtained from the Voronoi tessellation of the first step, which could be utilized for more explicit path planning along the sub-tours, are deleted. As for the edges in FPG, if there is a path between two nodes in REG that is shorter than a chosen threshold, an edge is added to FPG.

   It should be emphasized that the choice of various thresholds along the way must be done in an heuristic and iterative manner in order to obtain a "reasonably dense" FPG. In practice, this means that one wants to make FPG as sparse as possible but without excluding good sub-paths or disconnecting it.

[1]Which naturally coincides with the minimum time formulation because of the constant speed assumption made in the paper.

2. During the second phase, an ordered target set is dedicated to each vehicle. This phase is based on the information provided by FPG and includes two subsequent steps. In a first step, one of the three proposed (semi) greedy algorithms for constructing initial task assignment runs. In the second step, this initial assignment is refined using spatially constrained exchange of one or several targets between the vehicles. This exchange procedure is based on random selection between four exchange operators that are defined in the paper.

Mathematical tools used: Voronoi tessellation, graph pruning, greedy- and heuristic combinatorial optimization.

Personal Comments, Pros and Cons, assessment of paper quality

The idea of graph pruning using given thresholds and its connection with the satisficing framework is appealing. Interesting enough, the same idea underlies one of the most successful methods for meta heuristic solution of the VRP, namely the Granular Tabu Search method (see Section 4.9 or reference [36]).

4.7 Minimum time multi-UGV surveillance, by D.A. Anisi and P. Ögren

This paper can be found in reference [2].

Problem Formulation

The problem of covering a user defined area with obstacles with surveillance UGVs in minimum time, is addressed. The UGVs sensor range is large, and the only limitations of the range are occluding obstacles. The authors name this problem *Minimum Time UGV surveillance problem (MTUSP).*

Relation to the C–TAPP Problem

It is very relevant, especially for surveillance applications of the C–TAPP problem.

Proposed Solution Method and Mathematical Tools Used

The area that should be covered, A, is assumed to be an orthogonal polygon with orthogonal polygon holes. For a definition of polygons with holes read Section II in reference [2]. A convex cover of the area is generated by the union of all rectangles with maximized area in A. The UGVs should start at certain depots in A and return to certain depots in A in minimum time. Assigning these rectangles to the UGVs in an optimal way, considering the objective function, is $\mathcal{NP}$-hard. Hence a heuristics is applied.

The solution method consists of an algorithm decomposed into three subproblems. Firstly the convex cover of the area is generated, secondly a Tabu search assignment/ordering algorithm runs assigning rectangles to UGVs. In each Tabu step the third subroutine is called which calculates the optimal path for each vehicle given the rectangles it should visit. The third subroutine uses Dijkstra's graph search method on a specially designed *Route Graph*.

The objective function is also possible to change to become a compromise between minimizing travel time or minimizing total traveled distance for all UGVs. Simulation examples are shown of both cases.

Mathematical tools used: Combinatorial optimization, set covering, heuristic methods, Tabu search, Dijkstra's algorithm.

Personal Comments, Pros and Cons, assessment of paper quality

This is a good method for surveillance purposes. Well written. No tight bounds are presented though.

4.8 Real-time optimal mission scheduling and flight path selection, by K. Yoonsoo and D-W. Gu and I. Postlethwaite

This paper can be found in reference [33].

Problem Formulation

In a hostile environment with P static threats of different magnitude, N UAVs visit M targets. Each UAV can at most visit q targets and $M \leq Nq$. The UAVs' starting points are not necessarily the same, and do not necessarily coincide with their terminal points. The objective is to minimize the individual traveling cost to balance the workload. The problem should be handled in real-time and its solution has to be within a bound from optimum.

Relation to the C–TAPP Problem

The problems stated are related to the Multi Vehicle Routing Problem (MVRP), where task assignment and path planing depend on each other. This paper is interesting for the C–TAPP problem.

Proposed Solution Method and Mathematical Tools Used

An environment is generated with threats and targets. Threat contours are introduced, making the travel cost higher for traveling near a threat. The authors consider a cover set of rectangles, covering the threats. A vertex set consisting of the corner points of the rectangles and the targets is constructed. Dijkstra's graph search algorithm is applied to get the shortest paths between the points in the set.

Now the objective is to calculate T^*, where

$$T^* \equiv \min_{j} \max_{A(j)_i} T(j)_i^*,$$

is the least maximum traveling cost among all UAVs finishing all its sub-assignments. Here, $A(j) \in A$ is one of the feasible assignments, $A(j)_i$ is the the sub-assignment given to the $i^{textrmth}$ UAV, and $T(j)_i^*$ is the traveling cost for the $i^{textrmth}$ UAV finishing all its sub-assignments. Two problems are considered, one in which the UAVs need to return to their starting positions and one in which they do not.

Two exact solution methods E, E_{ret} and four non-exact algorithms H_1, H_2, H_3 and H_4 are proposed for the two problems. E solves the no-return problem and E_{ret} solves the return problem. All methods use Mixed Integer Linear Programming (MILP).

Now the four suboptimal algorithms H_1 to H_4 are all divided into two phases. In *phase one* there is a suboptimal partitioning of tasks among the UAVs, and in *phase two* E or E_{ret} is used on each UAV separately. The cost $T \geq T^*$ is obtained for each algorithm.

In H_1 the cost of traveling from the start position to the most expensive target for the $i^{textrmth}$ UAV is minimized. Traveling costs between targets are ignored. This gives an upper bound $\frac{T}{T^*} \leq 2q-1$. This bound can be tightened by further choosing as low value on q as possible (if it is possible to change q).

The number of binary variables needed and the number of constraints are both of order MN. In H_2 the costs of traveling between targets are also accounted for, this algorithm has an upper bound of $\frac{T}{T^*} \leq q$, and the order of the number of binary variables and constrains is MN^2.

In H_3 and H_4 the objective is to minimize the traveling cost from the $i^{textrmth}$ UAVs start position to the second target visited via the first target visited, and also from any starting position or target to the next two target positions respectively. The orders of the number of variables and constrains needed in the two algorithms are MN^2 for H_3 and MN^2 respective MN^6 for H_4. The upper bounds are $\frac{T}{T^*} \leq \frac{2(q-1)}{3} + 1$ if $q = 3k + 1 (k = 0, 1, 2, 3, ...)$ otherwise $\frac{T}{T^*} \leq 2\lceil \frac{q}{3} \rceil$ for H_3, and $\frac{T}{T^*} \leq \lceil \frac{q+1}{2} \rceil$ for H_4.

Mathematical tools used: Dijkstra's algorithm, MILP.

Personal Comments, Pros and Cons, assessment of paper quality

A comprehensive and detailed paper. Well posed solutions, the amount of binary variables seems a bit large though, even in the approximate solutions.

4.9 Classical and Modern Heuristics for the Vehicle Routing Problem , by G. Laporte and M. Gendreau and J.Y. Potvin

This paper can be found in reference [36].

Problem Formulation

The article is a survey of heuristic algorithms for the Vehicle Routing Problem (VRP). Following the paper, the VRP is described as follows. Consider a graph $G = (V, E)$ where $V = \{0, ..., M\}$ is a vertex set. In this article and in general, attention is only restricted to the undirected case, *i.e.*, $E = \{(i, j) : i, j \in V, i \leq j\}$ represents an edge set. Vertex 0 is a depot while the remaining vertices are customers. With each vertex $V \setminus \{0\}$ is associated a non-negative cost of length c_{ij}, (since G is undirected c_{ij} and c_{ji} are used interchangeably). The VRP consists of designing N vehicle routes of least total cost, each starting and ending at the depot, such that each customer is visited exactly once, the total demand of any route does not exceed the vehicle capacity Q, the length of any route does not exceed a preset maximal route length L. In some versions of the problem N is fixed *a priori*. In others it is a decision variable. The problem is $\mathcal{NP}$-hard. For other surveys of the VRP, see [10, 37].

The paper is divided into two parts, Classical Heuristics and Modern Heuristics (Tabu Search Heuristics).

Relation to the C–TAPP Problem

A good survey of different heuristic approaches for solving the VRP, which is relevant for C–TAPP.

Proposed Solution Method and Mathematical Tools Used

In the first part of the article classic algorithms such as the savings algorithm, sweep algorithm and petal algorithms are described. In the second part modern heuristics are presented, *i.e.* Tabu search methods.

The savings algorithm It applies to the VRP where the number of vehicles is a decision variable. The algorithm is a two step procedure, where the second step runs either parallel or sequential. First the saved cost s_{ij} is computed for traveling from customer i to customer j, *i.e.* (i, j) instead of traveling $(0, i)$ and $(j, 0)$, $i, j \in \{0, ..., M\}$ $i < j$. In other words s_{ij} represents the saved cost of merging a route starting with $(0, i)$ with a route ending with $(j, 0)$. The savings are stored in a list of nondecreasing order. M routes are created as $(0, j, 0)$, where $j \in \{0, ..., M\}$. Now in the second step routes are merged with each other either parallel or sequential until it is not possible to save more route costs.

The sweep algorithm This algorithm works by clustering customers. The clustering is performed by rotating a ray centered at the depot, and divide the vehicles into N different clusters depending on what their angles are. Each vehicle solves a Traveling Salesman Problem (TSP) in the cluster it is assigned to.

The petal algorithm Several routes are constructed, called petals. This could be done *e.g.* with the savings algorithm. Now a set partitioning problem is solved, *i.e.* all customers should be visited only once, and all customer should be visited. The union of the petals used is the empty set and the cost should be minimized.

Cluster-first, route-second In this algorithm the customers are gathered into clusters. To each cluster belongs a seed point. Through a General Assignment Problem (GAP), all seed points are allocated to vehicles. A TSP is finally solved within the clusters for each vehicle.

Several Improvement heuristics are used, where exchange of customers between vehicles are possible according to different exchange moves, *e.g.* see Section 4.10. A comparison between the classical heuristics are presented in terms of achieved solution in relation to optimum for a varying number of targets. The computation time is also presented for some of the methods. From the results it is obvious that the petal algorithms outperform the other algorithm.

Tabu search heuristics In the second part some different Tabu search algorithms are investigated, which are meta-heuristics. Starting with a solution generated from one of the classical algorithms or a greedy algorithm such as the Nearest Neighbor algorithm, exchange of customers are made possible between different vehicles with different moves. Moves done recently are saved in a Tabu list, such moves are not allowed to be used for p future iterations, where p is a positive number. In other words a move is contained in the Tabu list for p moves. The Tabu list is introduced to prevent the solution from getting stuck in a local minima. The algorithm is terminated when a stopping criteria is met, such as no improvement in a number of subsequent iterations.

Different Tabu search heuristics are described and compared to the classical algorithms, and from the comparisons it is obvious that the Granular Tabu Search (GTS) outperforms the other algorithms. It is almost as close to optimum as the other Tabu search algorithms but is often more than 10 times faster, almost like the classical algorithms. This can be partly explained because GTS only considers paths between targets that are shorter than a chosen threshold value.

Personal Comments, Pros and Cons, assessment of paper quality

A good survey. Even though it is rich in words, the context is sometimes a bit unclear, sometimes the words should have been replaced by mathematical expressions.

4.10 A Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows and Split Deliveries, by S.C. Ho and D. Haugland

This paper can be found in reference [28].

Problem Formulation

The Vehicle Routing Problem with Time Windows (VRPTW) is an extension of the Vehicle Routing Problem (VRP) (see Section 4.9), where each customer has to be served within a certain time interval. Also each customer i is assumed to have a demand d_i that must be fulfilled. In the problem addressed by the authors, an extension to VRPTW is considered with split deliveries, VRPTWSD. Vehicles can deliver fractions of the demand d_i to customer i, where the fractions sum up to d_i. For other papers about the VRPTW see [38, 11, 5, 22].

Relation to the C–TAPP Problem

In all types of Vehicle Routing Problems (VRP), the path planning and the task assignments are coupled. VRP problems are relevant for C–TAPP.

Proposed Solution Method and Mathematical Tools Used

Firstly the problem is constructed as a Mixed Integer Linear Program (MILP). Secondly a Tabu search heuristic is introduced. A Tabu search heuristic is a meta heuristic, that uses a Tabu lists with forbidden moves. This prevents the solution to get stuck in local minima.

In the Tabu search heuristic constructed by the authors, an initial solution is first constructed, consisting of a set of routes $\sigma = \{R_1, ..., R_v\}$. This solution is constructed by a greedy algorithm. Each customer has a time window when it can be served, *i.e.* a starting time a_k and a stopping time b_k $(k = 1, ..., n)$. When a route for a vehicle is constructed, a new customer j is added after the last customer i on the tour if the following holds

$$j = \arg\min\{t_{ik} + \max\{a_k - \theta_i - t_{ik}, 0\} | k \in C\}.$$

The parameter θ_i is the time that the vehicle starts to service customer i and C is the entire set of customers and the depot. If the vehicle cannot deliver the entire demand to the last customer in the route, the vehicle will deliver a fraction and return to the depot. More routes than available vehicles might occur. If the succeeding algorithms cannot reduce the number of routes, the solution will be declared infeasible.

Now the main Tabu search algorithm is applied on the initial solution. Four different move operators are introduced, *relocate*, *relocate split*, *exchange*, and *2-opt**. The operator *relocate* takes a customer i from route R_k and puts it somewhere in route R_l. If a customer is visited by both R_k and R_l, *relocate split* can be used to remove the customer from R_k and instead force R_k to visit one of the customers only visited by R_l. The split has been moved to a new customer. The *exchange*-operator works by exchanging a customer in R_k for a customer in R_l, that may be inserted anywhere in the routes. *2-opt** works by replacing all customers visited after customer i in R_k by all customers visited after customer j in R_l respectively.

To these four moves belong four neighborhoods with four Tabu-lists. Moves made in each neighborhood less than p iterations ago are contained in respective list. A move contained in a Tabu list can not be used unless the move will generate the best solution seen so far. Every q iteration a route saving phase is performed, trying to eliminate routes with maximum 3 customers. The algorithm will run until the solution is not improved for y consecutive iterations. Finally a post optimization phase will run, where each route will be optimized locally.

The algorithm runs both on VRPTWSD and VRPTW in a modified form, and the authors claim that it improved 5 of the 56 best published solutions to the Solomons benchmarks.

Mathematical tools used: Tabu search heuristics.

Personal Comments, Pros and Cons, assessment of paper quality

Well written. The MILP formulation was well stated.

4.11 MACS–VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows, by L.M. Gambardella, E. Taillard and G. Agazzi

This paper can be found in reference [22].

Problem Formulation

The problem of solving the Vehicle Routing Problem with Time Windows (VRPTW) is addressed, see Section 4.10.

Relation to the C–TAPP Problem

Path planning an task assignment are strongly coupled when solving vehicle routing related problems, hence the paper is relevant for C–TAPP. The method used in this paper, Ant Colony Optimization (ACO), is one of many heuristic approaches.

Proposed Solution Method and Mathematical Tools Used

A virtual ant colony is assumed where ants are walking around leaving pheromone trails. Pheromone trails are used for exploration and exploitation. In exploration a higher probability is given to elements with a strong pheromone trail, and in exploitation higher probability is give to an element that maximizes a blend of pheromone trail values and heuristic evaluations.

Exploration Chose with probability $(1 - q_0)$: $\frac{\tau_{ij} \cdot [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} \tau_{il} \cdot [\eta_{il}]^\beta}$ if $j \in N_i^k$ else 0.

Exploitation Chose with probability q_0: $\tau_{ij} \cdot [\eta_{ij}]^\beta$ $j \in N_i^k$.

Here, η_{ij} is a static value of the closeness of the nodes i and j, and τ_{ij} is a dynamic value of the amount of pheromone on the trail between node i and node j. Also, β and q_0 are parameters and N_i^j are the nodes yet not visited. The pheromone trails are updated locally and globally. Locally a visited trail get smaller amounts of pheromone. Globally trails get higher amounts of pheromone if they participate in paths of shorter length that minimizes the objective function.

The authors now introduce *MACS-VRPTW*. They use two coupled ant colonies. The first is minimizing the number of routes used, *ACS-VEI*, and the second is minimizing the total travel time *ACS-TIME*. Both subroutines use independent pheromone trails but they share the same optimal path. The algorithm runs until a stopping criteria is fulfilled. The authors claim that *MACS-VRPTW* improves some of the best solutions known for a number of problem instances in the literature.

Mathematical tools used: Ant colony optimization, nearest neighbor.

Personal Comments, Pros and Cons, assessment of paper quality

Interesting and good formulations, however even though the method improves several benchmark problem, computation times are sparsely accounted for.

4.12 Formulation and solution of the UAV paparazzi problem, by P. Ögren *et al.*

This paper can be found in reference [40].

Problem Formulation

The objective function of the Traveling Salesman Problem (TSP), the Vehicle Routing Problem (VRP) and the Assignment Problem (AP) are modified to account for targets that are capable of hiding, *i.e.* not possible to detect. The problem is similar to the one that Paparazzi photographers are facing, hence the so called Paparazzi utility function is formulated for the targets.

Relation to the C–TAPP Problem

It is very relevant.

Proposed Solution Method and Mathematical Tools Used

The Paparazzi utility function for target j, chased by a single UAV i, is expressed by

$$c_{ij}(\pi) = P_j \, \mathrm{e}^{-k_j(\frac{d_\pi(i,j)}{\nu}+T_j)},$$

where P_j is the revenue of capturing the target, $k_j \geq 0$ is a measure of how fast the probability of finding the target decays, ν the constant speed of the UAV, and T_j the time elapsed since the target was last seen. $d_\pi(i,j)$ denotes the total length that the UAV must travel in order to to reach the target if its route is defined by the permutation π. By permutation is meant, some assignment of targets to UAVs.

The authors use the Paparazzi utility function in the TSP, VRP and AP. TSP is the non-capacitated single vehicle version of VRP, see Section 4.9. For AP, or OAP as it is referred to in Section 4.1, good algorithms for finding the global optimum exists. For the other problems a Tabu search heuristic is used.

Mathematical tools used: Combinatorial optimization, heuristics, Tabu search.

Personal Comments, Pros and Cons, assessment of paper quality

A good presentation of the Tabu search method.

4.13 Decentralized Algorithms for Vehicle Routing in a Stochastic Time-Varying Environment, by E. Frazzoli and F. Bullo

This paper can be found in reference [8].

Problem Formulation

The N-vehicle Dynamic Traveling Repairperson Problem (N-DTRP) is considered. There are N vehicles inside a convex environment $\mathbf{Q}$. In $\mathbf{Q}$ there are also stochastically generated targets, according to a homogeneous spatio-temporal Poisson process. The expected waiting time to service shall be minimized for the targets.

Relation to the C–TAPP Problem

This is relevant for cooperative task assignment and path planning in stochastic and time-varying environments.

Proposed Solution Method and Mathematical Tools Used

An environment $\mathbf{Q} \subset \mathbf{R}^2$ is considered. First two problems are explained, *The Continuous Multi-Median Problem* and *The Traveling Salesman Problem* (TSP). In the multimedian problem, a couple of vehicles shall be placed in a way that minimizes the expected service time for a stochastically generated target, *i.e.* creating the optimal Voronoi cells. The TSP considered is the TSP restricted to $\mathbf{R}^2$ and the Euclidean TSP. Interesting aspects of the Euclidean TSP are presented. Let TSP(D) denote the minimum length of a tour through all points in the environment D, then the following holds

$$\lim_{M \to +\infty} \frac{TSP(D)}{\sqrt{M}} = \beta_{TSP,2},$$

where M is the number of targets and $\beta_{TSP,2}$, is a constant.

The authors briefly present a couple of algorithms for solving the TSP, among them *concord*, and *Linkern*. Both are available in ANSI C code at `http://www.math.princeton.edu/tsp/concorde.html`

The authors now present methods that already exists for the N-DTRP, and then a new method. The new method is constructed in two steps. First a method is developed for the single-vehicle DTRP, then this policy is used in the Multi Vehicle DTRP.

1. **Single-Vehicle Receding Horizon Median/TSP (sRH) policy**– While there are no targets, move at unit speed to the optimal position of the median problem, otherwise stop. While there are targets, do the following ; (i) for a given $\eta \in (0, 1]$, find a path that maximizes the number of targets reached within $\tau = \max\{\text{diam}(\mathbf{Q}), \eta TSP(D)\}$ time units; (ii) service from the current location this optimal fragment. Repeat.

2. **Multi-Vehicle Receding Horizon Median/TSP (mRH) policy**– N vehicles are considered. For all $i \in \{1, ..., N\}$, the i-th vehicle computes its Voronoi cell V_i and executes sRH(V_i), with the single following modification. While vehicles are servicing targets in their Voronoi cells they will shortcut all targets already visited by other targets.

Mathematical tools used: Voronoi-cells, Poisson distribution.

Personal Comments, Pros and Cons, assessment of paper quality

A well written paper and interesting aspects are presented of the TSP and the median problem. Bounds are presented for the methods and also numerical results.

4.14 Decentralized Task Assignments for Unmanned Aerial Vehicles, by M. Alighanbari and J.P. How

This paper can be found in reference [19].

Problem Formulation

M tasks shall be assigned to N UAVs. A distinction from the paper in Section 4.15, presented by the same research group, is that multiple tasks can be assigned to an agent. The score of achieving a task is decaying in time. The aim is to do task assignment decentralized, without conflicts in the assignments of tasks between different UAVs. Conflicts occur when more than one UAV is assigned to the same target. This problem could be seen as a decentralized robust version of the Vehicle Routing Problem (VRP).

Relation to the C–TAPP Problem

It is relevant when errors occur in sensor data.

Proposed Solution Method and Mathematical Tools Used

Heterogeneous vehicles are assumed, *i.e.* vehicles with different capabilities, sensor ranges *etc.*. A binary matrix K is given, where $K_{vw} = 1$ if vehicle v can be assigned to target w. Nothing is mentioned about the construction of K so we assume it to be given *a priori*. Each UAV is assumed to communicate to at least one other UAV. A connected communication graph is assumed, *i.e.* all UAVs can communicate with all other UAVs in an ad hoc network. A discrete time information consensus protocol is presented.

$$\dot{I}_i(t+1) = I_i(t) + \sum_{j+1}^{N_V} \sigma_{ij} G_{ij} (I_j(t) - I_i(t)),$$

where G_{ij} is a positive constant representing the relative confidence of UAV_i to UAV_j about their information. Here, $\sigma_{ij} \in \{0, 1\}$ equals 1 if a communication link exists between UAV_i and UAV_j, else 0.

The shortest paths are constructed of all possible permutations of the task assignment combinations for each UAV, using graph search algorithms. These paths are referred to as petals. Now an assignment algorithm is constructed in form of a MILP, where the best petal is chosen for each UAV, maximizing the global score. Only one vehicle might visit a target at the most.

Now some different phases are introduced to reach consensus in task assignments between vehicles. In the first phase, the vehicles try to reach consensus in the situational awareness, *i.e.* consensus in information about the environment. Each UAV is assumed to have an individual situational awareness perhaps different from the other UAVs, this is due to measurement errors. If no consensus is reached, conflicting solutions might occur where more than one UAV is assigned to the same target. If full consensus is reached, all the vehicles run the

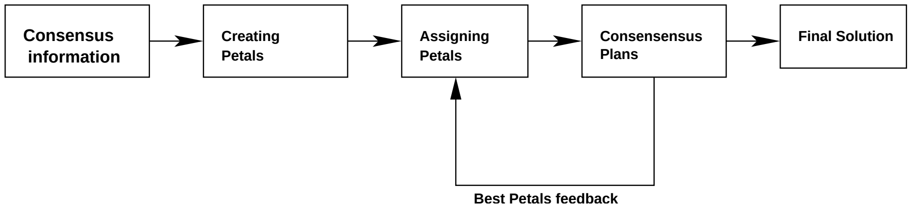


Figure 4.4: The different phases during task assignment.

same algorithm which is called to run the algorithm implicit, in that case no conflicts occur.

If no consensus in the situational awareness is reached, each UAV runs the assignment algorithm individually and then chooses a set of its "best" petals. These petals are then distributed to the other UAVs via the consensus protocol. This could be seen as a feedback of plans. The best petals are chosen by running the algorithm as usual, receiving the best petal which is removed from the set of petals. The algorithm then runs again and a new best petal is received and removed from the set of petals. This is done until the k best petals are received for UAV_i. Finally there will be a reassignment using the distributed $N \cdot k$ petals. The authors argue that making the set of "best" petals larger, considerably decreases the risk of conflicts.

Mathematical tools used: MILP, graph search algorithms, consensus control.

Personal Comments, Pros and Cons, assessment of paper quality

This paper is well written and feedback of plans is an interesting idea.

4.15 Robust Planning For Coupled Cooperative UAV Missions, by L.F. Bertucelli, M. Alighanbari and J.P. How

This paper can be found in reference [7].

Problem Formulation

The assignment problem with disturbed target positions is introduced. N UAVs are being assigned to $M \geq N$ targets. Each UAV must be assigned to one target only. An extension of this problem also dealt with, is the assignment of a fleet of heterogeneous vehicles. Couplings between reconnaissance tasks and strike tasks are investigated.

Relation to the C–TAPP Problem

Relevant for stochastic environments and interesting coupling of tasks, the assignment problem treated is quite trivial though.

Proposed Solution Method and Mathematical Tools Used

An environment is considered where the i^{th} target score is assumed to be Gaussian with expected score $\overline{c}_{k,i}$, and standard deviation $\sigma_{k,i}$ at time k. A stochastic programming approach to the assignment problem is the following MILP formulation

$$\begin{aligned} \max J_k &= \sum_{i=1}^{N_T} \overline{c}_{k,i} x_{k,i} \\ \text{subject to :} & \quad \sum_{i=1}^{N_T} x_{k,i} = N, \quad x_i \in \{0,1\}, \end{aligned}$$

where $x_{i,k}$ is a binary variable that equals one if target i is taken by some UAV at time k. All targets must be taken at all times k. It is assumed that all targets are situated at the same start position, and that the disturbances are the same for all UAVs. Solving this problem at time k is easier than the Optimal Assignment Problem OAP, and only involves sorting a vector of target costs $\overline{c}_{k,i}$.

This formulation only considers first moment information, $\overline{c}_{k,i}$, and ignores second moment information, $\sigma_{k,i}$. The authors argue that a better model is instead to maximize the "best" worst case score, making the system more robust. They argue in favor for the Soyster formulation with a little twist called μ. The MILP formulation of the assignment problem becomes

$$\begin{aligned} \max_x J_k &= \sum_{i=1}^{N_T} (\overline{c}_{k,i} - \mu_i \sigma_{k,i}) x_{k,i} \\ \text{subject to :} & \quad \sum_{i=1}^{N_T} x_{k,i} = N_V, \quad x_i \in \{0,1\}. \end{aligned}$$

When $\mu_i = 1$ is this formulation equal to the Soyster formulation. Introducing μ makes it possible to tune the level of robustness in the solution. The simulations presented show that the expected outcome of a robust solution is slightly lower than the expected outcome of a stochastic solution. The standard deviation is decreased significantly though, which was expected.

In the second part of the paper, missions with heterogeneous vehicles are considered. First an estimator model is presented of how the predicted standard deviation $\sigma_{k+1|k}$ at time $k+1$, is estimated from the standard deviation σ_k at time k. Then $\sigma_{k+1|k}$ is used in the MILP formulation. Different MILP approaches with heterogeneous vehicles are presented in coupled and uncoupled missions.

Two types of vehicles are assumed, reconnaissance vehicles and strike vehicles. In the uncoupled approach the strike and reconnaissance vehicles are sent out in parallel without communication between them. In the coupled approach the strike vehicles know what targets the reconnaissance vehicles will visit, and use this information in the optimization. The coupling between the two vehicles yield a nonlinear objective function, which is made linear by introducing a new binary variable and some linear constraints on it. The simulations presented shows that the coupled approach yield a higher value in the objective function.

Mathematical tools used: MILP, Gauss distributions, Estimator/Predictor models, Stochastic Optimization and Robust Optimization.

Personal Comments, Pros and Cons, assessment of paper quality

Perhaps hard to implement in the VRP and computationally time consuming. However coupling task execution between heterogeneous vehicles is interesting.

4.16 Filter-Embedded UAV Task Assignment algorithms for Dynamic Environments, by M. Alighanbari, L.F. Bertuccelli and J.P. How

This paper can be found in reference [1].

Problem Formulation

The authors have published a series of papers on variations of the same theme, see Sections 4.15 and 4.14. In this paper, the problem of assigning N UAVs to $M \geq N$ targets is considered when disturbances occur in target scores. Each UAV must be assigned to one target only. In each time step the UAVs are reassigned to the targets, assuming that the vehicles receive new information of target scores in each time step. The authors present a method to filter the measurements, *i.e.* reducing the disturbances, and by doing so increase task assignment performance.

Relation to the C–TAPP Problem

The problem is posed as an assignment problem only, and is solved to optimum by a central linear program in NM^2 time (*e.g.* Kuhn's Hungarian method). The problem is relevant when noise occur in target measurements.

Proposed Solution Method and Mathematical Tools Used

Measurements of targets are assumed to be disturbed. As new sensor information about the targets reach the UAVs, the assignment algorithm runs again. The assignment problem is formulated as a Linear Integer Program (LIP). The program is centralized.

A problem that might occur when dealing with disturbed measurements is "churning", which means that a UAV decides to go one target in a time step k but decides to go to another one in time step $k + 1$. The UAV then may alternate between these two targets, and as k increases the UAV might get caught in the middle.

To avoid this, a binary filter is introduced to cut of high frequency shifting in the assignments between time steps. The output of the filter is then used in the assignment. The filter is first described using logical formulations and then translated into the language of LIP. As input the filter gets target measurements. In a first approach, the filter feeds back its output into itself, and in a second approach the filter also gets the unfiltered signal as an input.

The authors show in some examples that their filters work. By cutting of high frequencies, the disturbances will be ignored.

Mathematical tools used: LIP, binary filters, Fourier transform.

Personal Comments, Pros and Cons, assessment of paper quality

Well written paper and well stated problems for task assignment with measurement errors. The "churning" problem seems quite unlikely though, which is also mentioned by the authors.

4.17 Market-Based Multirobot Coordination: A Survey and Analysis, by M.B. Dias, R. Zlot, N. Kalra and A. Stentz

This paper can be found in reference [12].

Problem Formulation

The paper constitutes a survey of market-based approaches to multi-robot coordination. Papers that offer market based approaches to multi robot coordination can be found in [46, 47, 35, 6, 43].

Relation to the C–TAPP Problem

Market-based approaches is one of the possible *explicit* coordination schemes that can be adopted within a multi-vehicle team.

Proposed Solution Method and Mathematical Tools Used

The authors define a set of underlying elements shared by market-based multirobot approaches. 1) The team of vehicles are given an objective that can be decomposed into subcomponents achievable by individuals or subteams. The team has access to a limited set of resources with which to meet this objective; 2) A goal objective function quantifies the system designer's preferences over all possible solutions; 3) An individual utility function (or cost function) specified for each vehicle quantifies that vehicle's preferences for its individual resource usage, and contributions towards the team objective given its current state. Evaluation of this function does not require global knowledge; 4) A mapping between the team objective function and the individual and subteam utilities; 5) Resources and individual or subteam objectives can be redistributed using a mechanism such as an auction.

In an auction, a set of items are offered by an auctioneer in an announcement phase to the team of vehicles. In a single-item auction only a single item is offered at each phase, compared to combinatorial auctions where multiple items are offered and any vehicle or subteam of vehicles can bid on any combination of subsets of the items. In between, in multi-item auctions, multiple objects are offered but the participants can at most win one item apiece.

The authors argue that market based approaches fall into a hybrid category in the middle of the spectrum between centralized and fully distributed approaches to multi-robot coordination. They also discuss costs and utilities and refer to [25].

The authors give the following formal definition to the Multirobot Task Allocation Problem.

The Multirobot Task Allocation Problem: Given a set of tasks T, a set of robots R, and a cost function for each subset of robots $r \in R$ specifying the cost of completing each subset of tasks, $c_r : 2^T \to \mathbb{R}^+ \cup \{\infty\}$, find the allocation $A^* \in R^T$ that minimizes a global objective function $C : R^T \to \mathbb{R}^+ \cup \{\infty\}$, where R^T is the set of all possible allocations of the tasks T to the team of robots R.

Task allocation gets complicated when the tasks are not independent or constrained. Sometimes tasks can be roles, *e.g.* the different positions in a football team.

How to combine task decomposition with task allocation is also an important issue. There are two common approaches to this planning problem, *decompose-then-allocate vs. allocate-then-decompose*. In the former tasks are decomposed by a single agent recursively into simple subtasks and then allocated to the team. In the latter more complex tasks are allocated to the team members. Decoupling task allocation and task decomposition might result in highly suboptimal solutions. During task execution the authors refer to *loosely coordinated teams* and *tightly coordinated teams*, where teams in the former group only coordinate during task allocation and decomposition but not during execution, and teams in the latter group communicate also during execution. Tight coordination is challenging.

Other aspects of market based approaches that are investigated are *Quality of Solution*, *Scalability* and *Dynamic Events and Environments*. Theoretical guarantees and experimental results are delivered for four different auctioning approaches; 1) Combinatorial auctions; 2) Central Single task iterated auctions; 3) Central instantaneous assignment; 4) Peer-to-peer trading; 5) Central multi-task auctions followed by peer-to-peer trading. Also computational complexity is accounted for. Robustness and fluidity must be accounted for when dealing with dynamic events and environments, the robots must also be able to handle online tasks and uncertainties. The authors refer to work related to these issues, and also work dealing with heterogeneous teams.

Mathematical tools used: Combinatorial Optimization,mMarket based multi-robot coordination, utility functions, set theory.

Personal Comments, Pros and Cons, assessment of paper quality

An interesting paper. The paper does not only constitute a survey of market based approaches to multi-robot coordination, but also constitutes a survey of problems in task allocation, task decomposition and task execution.

4.18 The Generation of Bidding Rules for Auction-Based Robot Coordination, by C. Tovey and M.G. Lagoudakis

This paper can be found in reference [46].

Problem Formulation

The authors address the problem of how to derive good bidding rules for given team objectives, *i.e.* how to construct a good auction based vehicle coordination system. The three different team objectives dealt with are *MiniSum*, *MiniMax* or *MiniAve*, *i.e.* minimizing for a group of vehicles the total traveled distance, the mission completion time or the average waiting time for a target to be served.

Relation to the C–TAPP Problem

Market based approaches are examples of target assignment through explicit communication between vehicles. The tasks are not independent. This paper is interesting for the C–TAPP problem.

Proposed Solution Method and Mathematical Tools Used

The mission is to allocate a group of targets to a group of vehicles. For that purpose bidding rules are introduced. Assume there are N vehicles $r_1, ..., r_N$ and M currently unallocated targets $t_1, ..., t_M$. Assume further that a set of targets T_i are assigned to each vehicle r_i, *i.e.* there is a set $\{T_i, ..., T_N\}$. Let $\mathrm{PC}(r_i,T_i)$ denote the minimum path cost of vehicle r_i and $\mathrm{STC}(r_i,T_i)$ denote the minimum sum of per target cost over all targets in T_i from its current location. The values of $\mathrm{PC}(r_i,T_i)$ and $\mathrm{STC}(r_i,T_i)$ are calculated locally using a meta-heuristic procedure. The three team objectives are

- **MiniSum** $\min_T \sum_j \mathrm{PC}(r_i,T_i)$
- **MiniMax** $\min_T \max_j \mathrm{PC}(r_i,T_i)$, and

- **MiniAve** $\min_T \frac{1}{m} \sum_j \text{STC}(r_i,T_i)$.

Each unallocated target is bidden upon by vehicles in a bidding round according to a bidding rule. The vehicle with the lowest bid will win, and the target will be allocated to that vehicle.

Bidding Rule Robot r bids on target t the difference in performance for the given team objective between the current allocation of targets to vehicles and the allocation that results from the current one if robot r is allocated target t. (Unallocated targets are ignored.)

The authors show that each vehicle's bid for one of the team objectives is decoupled from from the information of the other vehicles, making the system more decentralized and distributed. The authors finally present results matching the optimal ones computed with MILP in small environments.

Mathematical tools used: Meta-heuristics, vector summation.

Personal Comments, Pros and Cons, assessment of paper quality

Single item auctions are easy to implement, less communication is needed than in combinatorial auctions. The local meta heuristic methods are interesting.

4.19 Towards Collaborative Robots for Infrastructure Security Applications, by Guo, Y. and Parker, L.E. and Madhavan, R.

This paper can be found in reference [27].

Problem Formulation

The motivating application for this paper more or less coincides with the one considered in this survey, namely collaborative vehicle infrastructure security. In accordance with Sections 2.1 an 2.2, the two base-line scenarios are that of patrolling rounds and intruder/threat response. These scenarios are addressed by both considering the problem of distributed sensing for localization and mapping of the environment, as well as the multi vehicle motion planning problem. The authors also discuss the system integration of these two sensing an planning problems.

Relation to the C–TAPP problem

This paper is among the best matches found for the security applications considered in this survey. The overlapping between the problem considered in this paper and Problem 1 is evident.

Proposed Solution Method and Mathematical Tools Used

The multi vehicle localization problem is addressed by employing a distributed version of the Extended Kalman Filter (EKF), while the 3D terrain mapping algorithm is taken from [21].

As for the multi vehicle motion planning problem, which is the most interesting part of this work from the current survey's point of view, the following subsequent steps are taken:

1. The environment is partitioned into N number of disjoint regions. Essentially, the Voronoi regions, V_i, are assigned to each of the N sentry vehicles.

2. Each vehicle continuously patrols its Voronoi region, V_i. This sub-task is addressed by completely covering V_i with disks with radius as large as the sensor range, R_c. Minimum number of disks are used by adopting a regular and pre-defined cover pattern. The question of how to visit the covering disks is somewhat neglected in this paper, although the most straightforward solution is mentioned: static and pre-defined template paths to follow.

3. If an intruder is detected, a subset of the vehicles enter the "threat respond mode" and move from their current position to the threat position. This boils down to finding a feasible trajectory between two given points and is a well-studied problem. The remaining vehicles should re-calculate the steps 1) - 2) to provide continuous and complete patrolling.

Finally, the integration of the sensing and planning capabilities is achieved in accordance with Figure 4.5.

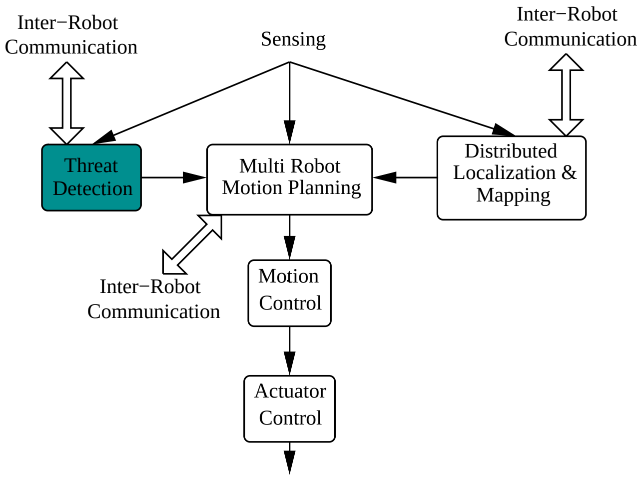


Figure 4.5: Block diagram of the integration of sensing and planning. Notice that the important threat detection issues (marked with a shadowed box) are not addressed in this paper.

Mathematical tools used: Distributed Extended Kalman Filter (EKF), Voronoi diagrams, set coverage.

Personal Comments, Pros and Cons, assessment of paper quality

The considered problem is of highest interest. Once the intrusion alarm goes off, it is not clear what sub-set of the vehicle that continue the patrolling task and what sub-set that enters the "threat respond mode".

4.20 Decentralized Control of Cooperative Robotic Vehicles: Theory and Application, by J.T. Feddema and C. Lewis and D.A. Schoenwald

This paper can be found in reference [18].

Problem Formulation

The authors describe decentralized control for a group of vehicles. Aspects such as input/output reachability, structural observability, controllability and connective stability of the system are investigated. The methods are tested in an experimental test platform for formation control, perimeter surveillance and surround tasks.

Relation to the C–TAPP Problem

The paper mostly deals with motion control and trajectory design, which are problems overlooked in this thesis.

Proposed Solution Method and Mathematical Tools Used

The authors define a system with N vehicles. The overall system is described by

$$\dot{x} = f(t, x, u)$$
$$y = h(t, x).$$

x is the state of the system, which consists of all N different substates. The variable u is the input to the system, and y is the output. The system can also be described as

$$\mathbf{S} : \dot{x}_i = f_i(t, x_i, u_i) + \tilde{f}_i(t, x, u), i \in \{1, ..., N\}$$
$$y_i = h_i(t, x_i) + \tilde{h}_i(t, x).$$

Now the authors assume a system of decoupled vehicles, *i.e.* $\tilde{f}_i(t, x_i, u_i) = 0$. A simple example is considered, where some vehicles should form a line formation. The vehicles are assumed to be simple integrators with a proportional control. A output feedback from the different vehicles output to each vehicles input is added. Each vehicle "talks" to two other vehicles and is also being "talked" to by two vehicles. Controllability, reachability and stability is investigated for the system, both in continuous and discrete time, in terms of parameters such as sampling time and interaction gain between the vehicles.

In the experimental setup there is, except for the vehicles, a base station. Two communication protocols are assumed, the star network and the token ring. In the star network all vehicles communicate with the base station, in opposite to the token ring, where each vehicle communicate only with its neighbors. The latter is more robust, and does not contain a single point of failure.

The vehicles spread out uniformly along the border of the perimeter which they should guard, according to the control model described in the article (token ring). Intrusion sensors are also spread out along the border. If an intruder cross the border, the vehicle closest to the intrusion sensor setting off the alarm approaches the sensor. The rest of the vehicles are once again spreading out uniformly along the perimeter.

A potential field approach for the surround task is also considered in the article.

Personal Comments, Pros and Cons, assessment of paper quality

A good description of decentralized control.

4.21 Control of multiple Robotic Sentry Vehicles, by J. Feddema, C. Lewis and P. Klarer

This paper can be found in reference [17].

Problem Formulation

A group of sentry vehicles shall guard border of an area. They shall also perform surround and division tasks.

Relation to the C–TAPP Problem

The problem is related, however only the somewhat static problem of guarding a perimeter is addressed. There is no algorithm for patrolling in the paper.

Proposed Solution Method and Mathematical Tools Used

Experiments are performed in an outdoor environment. A testbed is used including vehicles called "Roving All Terrain Lunar Explorer Rover" (RATLER), and a bas station (Laptop). Each vehicle is assigned a ID-number, making communication easier. The authors choose a token ring communication network, instead of a star formed network where each vehicle has to communicate via the base station. In the token ring each vehicle can only transmit information to one vehicle, and all vehicles are connected with the base station via other vehicles in a ring. In this system there is no single point of failure if the ring is reconfigured when a communication link is down.

A perimeter is controlled by equidistant vehicles positioned at it. The token ring communication system is implemented. At the perimeter, Miniature Intrusion Detection Sensors (MIDS) are placed. As soon as an alarm goes off, the nearest vehicle approaches the MIDS. The remaining vehicles adjust their positions to stand equidistant on the perimeter. There are descriptions of the state space of the single vehicles, and the state space of the interconnected system of vehicles.

Other issues accounted for in the article are formation control and path planning. Formation control, *e.g* arrow formation, is done by choosing a leader in a group, and then the other vehicles are placed graphically relative to the leader. A potential field approach to path planning is assumed. A primitive obstacle avoidance scheme is assumed. Robots that encounter a target back away and change direction.

Personal Comments, Pros and Cons, assessment of paper quality

Tests are performed in a real outdoor environment which is good. In the surveillance consensus protocol's similar to the ones in [44] are used. For C–TAPP the methods used in the article are a bit primitive.

4.22 Motion Planning with wireless Network Constraints, by D.P. Spanos and R.M. Murray

This paper can be found in reference [45].

Problem Formulation

The authors address feasibility aspects of motion planning for groups of vehicles connected by a range constrained wireless network.

Relation to the C–TAPP Problem

The considered problem is relevant when running decentralized algorithms and vehicles must stay connected with each other.

Proposed Solution Method and Mathematical Tools Used

A set V of N vehicles is assumed. The vehicles are approximated by simple integrators. This approximation amounts to a local controllability assumption. The authors argue that many applications involving robots instead of autonomous vehicles will truly be kinematic.

Each agent has a position $q_i \in \mathbf{R^2}$, and a fixed broadcast range r_i. The set of all vehicle positions is denoted by $\mathbf{Q}$. A graph $C = (V, E_C)$ called the *communication network* is created,

$$(i, j) \in E_C \quad \Leftrightarrow \quad \min\{r_i, r_j\} - d_{ij} \geq 0.$$

The set of vehicles that agent i is connected to is denoted by $N_C(i)$. The authors also introduce an $information\ flow$ graph I. This graph represents an abstract design requirement for the network. $N_I(i)$ denotes the set of vehicles connected to agent i in I. C is I-connected iff two vehicles connected by an edge in I are also connected by a path of at most two edges in C. Choosing two hops restrict the vehicles to communicate directly only with their nearest neighbors, making the system distributed.

The *geometric connectivity robustness* is introduced as

$$R_I(i) \quad \doteq \quad \min_{k \in N_I(i) \cup N_C(i)} \left[\frac{1}{2} \max_{j \in N_C(i)} P(i, j, k) \right],$$

where

$$P(i, j, k) \quad \doteq \min\{\min\{r_i, r_k\} - d_i k, \min\{r_j, r_k\} - d_j k\}.$$

When $R_I(i) \geq 0 \quad \forall i \in V$, C is I-connected.

The authors prove that it is possible to move from any I-connected formation to any other I-connected formation, by using the star convexity property of the set of I-connected configurations. One problem is that it is only possible to move between I-connected formations, and not all connected formations. The authors now relax I and make it possible to move between configurations belonging to different *information flows*, *i.e.* different Is. This is done using the constraints

$$\begin{aligned} R_I(i, \mathbf{Q}) &\geq 0 \\ I &= f(\mathbf{Q}) \end{aligned}$$

The authors now announce the main result of the paper. Let $\mathbf{Q}$ and $\tilde{\mathbf{Q}}$ be two configurations and suppose C($\mathbf{Q}$) and C($\tilde{\mathbf{Q}}$) are both connected (*i.e.* the connectivity graphs induced by $\mathbf{Q}$ and $\tilde{\mathbf{Q}}$). Let R_m be the minimum robustness of these two configurations. Then there exists a motion $\gamma(t)$ from $\mathbf{Q}$ to $\tilde{\mathbf{Q}}$ satisfying

$$R_{I_s}(i, \gamma(t)) \geq R_m \quad \forall i \in V, \quad t \in [0, 1].$$

This guarantees a certain robustness.

The authors briefly address the subject of difficulties with obstacles. There is an asymmetry in reachability here. A connected configuration $\mathbf{Q}$ is called unobstructed if there exists a contractive motion respecting the constraints of the obstacles taking the configuration to its center point. Any other connected configuration is called obstructed. It is possible to go to any connected configuration from an unobstructed configuration, but going from an obstructed configuration to other configurations is not possible with the methods presented.

Mathematical tools used: Graph connectivity, star-convexity.

Personal Comments, Pros and Cons, assessment of paper quality

A good paper for formation control and connectivity. The geometric connectivity measure seems like a good measure of robustness. However The necessity of introducing the information flow graph I might be questioned. It reduces the amount of possible formations unless it is relaxed.

4.23 Parallel Stochastic Hill-climbing with small Teams, by B.P. Gerkey and S. Thun

This paper can be found in reference [26].

Problem Formulation

The authors address the problem of coordinating N vehicles working toward a common goal. Problems like these are in general $\mathcal{NP}$-hard and require a computation load that is exponential in n. They introduce a heuristic called *parallel stochastic hill-climbing with small teams (parish)*, and apply it on a concrete problem: multi-vehicle pursuit evasion.

Relation to the C–TAPP Problem

The problem is relevant, elements of both path planning and task assignment occur.

Proposed Solution Method and Mathematical Tools Used

The authors first introduce *parish* in a abstract sense. In this formulation N vehicles face the multi-vehicle problem M. The maximum allowed team size is $t \leq N$ and q is a plan involving a team of vehicles. $P(q)$ is a probability function of choosing the plan q. They also introduce a value heuristic $v(q)$ consisting of the two parts $b(q)$ and $c(q)$, where $b(q)$ is the benefit of performing the plan towards the goal, and $c(q)$ is the cost of that plan, $v(q) = b(q) - c(q)$. $P(q_j) \geq P(q_i) \Leftrightarrow v(q_j) \geq v(q_i)$.

The algorithm $parish$**:**

While M not done
  do parallel for each vehicle s
    do for $1 \leftarrow l$ **to** t
      do $Q_l \leftarrow$ {q: q is a feasible l-searcher plan involving s } $\cup \{\emptyset\}$
        sample $\hat{q}_l$ from Q_l according to $P(q)$
        if $\hat{q}_l \neq \emptyset$
          then Execute $\hat{q}_l$
            break

This is the core formulation of the algorithm, which can be described as stochastic rather than greedy. The authors argue that letting the vehicles sometimes choose worse plans than they could, makes the total system avoid going into local optimum.

The information, plans *etc.* are distributed among the vehicles so that everybody has an equal copy, and the algorithm runs in parallel on every vehicle, thereby avoiding a single point of failure. The algorithm can run either online or offline. The value $b(q)$ is a combination of the benefit of a plan, and the (possible negative) value of disbanding groups of vehicles containing vehicles that should participate in q.

The authors now apply the algorithm for the multi-vehicle pursuit evasion game. Differential vehicles with 180° field of view is assumed. An area that will be searched is divided into convex regions, in this case rectangles, that do not exceed 8 m which is assumed to be a vehicles sensor range. An undirected graph $G(V, E)$ is constructed, where V, the vertices, are the regions and E, the edges, are the borders between the regions.

When skipping the details, the modified *parish* is:

```
(*Create a list of teams, T and a list of plans, A*)
  (*Start with singleton teams and no plans ∀ vehicles s_i (i = 1, ..., N)*)
    (*Each team decides what to do in parallel*)
      (*If no plan, this team has only one member, call it s*)
        (*Consider teams of increasing size up to t*)
          (*Make some plans with l vehicles, but also consider the null plan*)
            (*Choose one of the plans according to P( · )*)
            (*If choosing the null plan, keep looking*)
            (*Else, assemble the team, maybe disbanding other teams*)
              (*Store the chosen plan, start executing it*)
              (*We have a satisfactory plan, stop looking*)
      (*Else, we already have a plan, keep executing it*)
```

A simple example is presented with an indoor environment with connecting corridors. In this type of setting a singleton plan is to move vehicle to an adjacent region. A plan involving 2 vehicles is to move one vehicle to the current position of the other, and then move that vehicle to an adjacent region. A probability distribution where the best plan will be chosen with probability 0.9 and the other plans are chosen uniformly with the same probability. Good results are chosen for the example, but no hard bounds for the algorithm are presented.

Personal Comments, Pros and Cons, assessment of paper quality

A selection rule avoiding local optimum. The algorithm is also directed towards moving targets, but is applicable on static environments, *i.e.* pure surveillance missions.

4.24 Complete Multi-Robot Coverage of Unknown Environment with Minimum Repeated Coverage, by S.S. Ge and C-H. Fua

This paper can be found in reference [23].

Problem Formulation

Complete coverage of an area with obstacles in a finite time, while minimizing repeated coverage for N mobile ground vehicles. Low range sensors of vehicles are assumed. A priori knowledge about the environment is not necessary.

Relation to the C–TAPP Problem

The covering problem is more relevant for de-mining, vacuum cleaning *etc.*, than for surveillance applications of the C–TAPP problem.

Proposed Solution Method and Mathematical Tools Used

The vehicles are distributed over the searching area, E. Each vehicle r_i can cover a disc area with diameter d_r around its current position. The authors introduce so called spurious obstacles, which consists of areas already covered by vehicles. The vehicles should avoid the border of E, obstacles, spurious obstacles and other vehicles, each of them should not be approached closer than a certain distance. The space inside E that can be reached by vehicles is E_{ex}.

An unnamed algorithm is proposed, which runs distributed. The algorithm consists of two subalgorithms, Algorithm 1 (Normal Mode) and Algorithm 2 (Wrap Mode). In *Normal mode* the vehicle travels around always connected to its current spurious obstacle (avoiding all other spurious obstacles). The next position of the vehicle is the position that is adjacent to as much of its spurious obstacle as possible, not further away than $0.5d_r$ from its current position.

If no feasible new position is possible, the vehicle will change to *Wrap Mode*. In this mode, the vehicle will be able to travel along the border of its spurious obstacle until it finds an uncovered area, it will also be able to cover concave areas surrounded by obstacles.

The authors prove that the only areas that can suffer from repeated coverage are the areas between obstacles that are less than $2d_r$, this space is $|E_{nrw}|$. They also prove that the complete covering time of the area is bounded by:

$$\frac{4|E_{ex}|}{\pi N d_r^2} \leq T \leq \frac{4(|E_{ex}| + |E_{nrw}|)}{\pi d_r^2}$$

Personal Comments, Pros and Cons, assessment of paper quality

This paper is rich in symbols, and the notation is a bit heavy. However a solid method is proposed with a bounded time for completely covering the area.

5 Conclusions

A vast amount of research and huge numbers of publications have been devoted to consider problem formulations more or less related with the *cooperative task assignment and path planning* (C–TAPP) problem. In this survey, a broad sampling of the research that is currently ongoing in this field was provided.

In essence, it was concluded that an empirical, application-oriented perspective has underlied the overwhelming majority of the papers encountered. In addition, some important aspects that have been largely untreated in the literature was recognized. In particular, 1) the more theoretical aspects of the C–TAPP problem and frameworks for formal analysis and 2) evaluative and comparative studies, deserve much more attention from the research community.

To get an insight into where the front line lies as far as existing products are considered, a representative group of surveillance and sentry vehicles were presented. To keep this list tractable, only commercial or enterprise-ready products were considered. Their key features, capabilities and shortcomings were examined and listed. In summary, sentry vehicles of today can be described as well-equipped sensor platforms capable of performing a number of low-level, single-step tasks: *i.e.* if X occurs, do Y. This includes recording video and audio, taking digital photography, sounding an alarm or even releasing a dense smokescreen to frighten off the intruder. What is really needed to take this to the next level is then the challenging task of generalizing the X and Y to more complicated and advanced high-level missions.

Finally, in the last part of this survey, review of some of the individual papers encountered was provided.

Bibliography

[1] M. Alighanbari, L.F. Bertuccelli, and J.P. How. Filter-Embedded UAV Task Assignment Algorithms for Dynamic Environments. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pages 1–15, 2004.

[2] David A. Anisi and Petter Ögren. Minimum time multi-ugv surveillance. Submitted to the 46$^{\text{th}}$ IEEE Conference on Decision and Control, New Orleans, LA, Dec. 2007.

[3] R.W. Beard, T.W. McLain, MA Goodrich, and EP Anderson. Coordinated target assignment and intercept for unmanned air vehicles. *IEEE Transactions on Robotics and Automation*, 18(6):911–922, 2002.

[4] R. Beckers, O.E. Holland, and J.L. Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. *Artificial Life IV*, 181:189, 1994.

[5] R. Bent and P. Van Hentenryck. A Two-Stage Hybrid Local Search for the Vehicle Routing Problem with Time Windows. *Transportation Science*, 38(4):515–530, 2004.

[6] M. Berhault, H. Huang, P. Keskinocak, S. Koenig, W. Elmaghraby, P. Griffin, and A. Kleywegt. Robot exploration with combinatorial auctions. *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, 2, 2003.

[7] L.F. Bertuccelli, M. Alighanbari, and J.P. How. Robust planning for coupled cooperative UAV missions. In *Proc. of the* 43rd *IEEE Conference on Decision and Control, 2004. CDC.*, volume 3, 2004.

[8] Luca F. Bertuccelli and Jonathan P. How. Robust UAV Search for Environments with Imprecise Probability Maps. In *IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05, Seville, Spain*, pages 5680–5685, 2005.

[9] Y.U. Cao, A.S. Fukunaga, and A. Kahng. Cooperative Mobile Robotics: Antecedents and Directions. *Autonomous Robots*, 4(1):7–27, 1997.

[10] J-F. Cordeau, M. Gendeau, G. Laporte, J-Y. Potvin, and F. Semet. A guide to vehicle routing heuristics. *Journal of Operationsl Research Society*, 53(5):512–522, May 2002.

[11] J.F. Cordeau and Québec Groupe d'études et de recherche en analyse des décisions (Montréal. *The VRP with Time Windows*. Groupe d'études et de recherche en analyse des décisions, 2000.

[12] MB Dias, R. Zlot, N. Kalra, and A. Stentz. Market-based multirobot coordination: a survey and analysis. *Proceedings of the IEEE*, 94(7):1257–1270, 2006.

[13] BR Donald, J. Jennings, and D. Rus. Analyzing teams of cooperating mobile robots. *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 1896–1903, 1994.

[14] G. Dudek, M.R.M. Jenkin, E. Milios, and D. Wilkes. A taxonomy for multi-agent robotics. *Autonomous Robots*, 3(4):375–397, 1996.

[15] I. Dutta, AD Bogobowicz, and JJ Gu. Collective robotics-a survey of control and communication techniques. *Intelligent Mechatronics and Automation, 2004. Proceedings. 2004 International Conference on*, pages 505–510, 2004.

[16] A. Espinosa, J. Lerch, and R. Kraut. Explicit vs. Implicit Coordination Mechanisms and Task Dependencies: One Size Does Not Fit All. *Team Cognition: Understanding the factors that drive processes and performance. American Psychological Association*, 2004.

[17] J.T. Feddema, C. Lewis, and P. Klarer. Control of multiple robotic sentry vehicles. *Proceedings of SPIE Unmanned Ground Vehicle Technology, SPIE*, 3693:212–223, 1999.

[18] JT Feddema, C. Lewis, and DA Schoenwald. Decentralized control of cooperative robotic vehicles: theory and application. *Robotics and Automation, IEEE Transactions on*, 18(5):852–864, 2002.

[19] Decentralized Task Assignment for Unmanned Aerial Vehicles. Alighanbari, m. and how, j.p. In *Proc. of the* 44^{th} *IEEE Conference on Decision and Control and 2005 European Control Conference, CDC-ECC'05*, pages 5668–5673, 2005.

[20] E. Frazzoli and F. Bullo. Decentralized algorithms for vehicle routing in a stochastic time-varying environment. In *Proc. of the* 43^{rd} *IEEE Conference on Decision and Control, CDC*, 2004.

[21] K. Fregene, R. Madhavan, and LE Parker. Incremental multi-agent robotic mapping of outdoor terrains. *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, 2, 2002.

[22] L.M. Gambardella, É. Taillard, and G. Agazzi. MACS-VRPTW: a multiple ant colony system for vehicle routing problems with time windows. *Mcgraw-Hill'S Advanced Topics In Computer Science Series*, pages 63–76, 1999.

[23] Shuzhi Sam Ge and Cheng-heng Fua. Complete Multi-Robot Coverage of Unknown Environments with Minimum Repeated Coverage. In *IEEE International Conference on Robotics and Automation, Barcelona, Spain*, pages 727–732, April 2005.

[24] BP Gerkey and MJ Mataric. Sold!: auction methods for multirobot coordination. *Robotics and Automation, IEEE Transactions on*, 18(5):758–768, 2002.

[25] B.P. Gerkey and M.J. Mataric. A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems. *The International Journal of Robotics Research*, 23(9):939, 2004.

[26] B.P. Gerkey, S. Thrun, and G. Gordon. Parallel Stochastic Hill-Climbing with Small Teams. *Multi-Robot Systems: From Swarms to Intelligent Automata*, 3:65–77, 2005.

[27] Y. Guo, L.E. Parker, and R. Madhavan. Towards Collaborative Robots for Infrastructure Security Applications. *Proceedings of The 2004 International Symposium on Collaborative Technologies and Systems*, pages 235–240, 2004.

[28] S.C. Ho and D. Haugland. *A tabu search heuristic for the vehicle routing problem with time windows and split deliveries*. Department of Informatics, University of Bergen, 2002.

[29] Maria John, David Panton, and Kevin White. Genetic algorithm for regional surveillance. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 2, pages 1573–1579, Orlando, Florida, USA, 13-17 July 1999.

[30] Maria John, David Panton, and Kevin White. Mission planning for regional surveillance. *Annals of Operations Research*, 108:157–173, Nov. 2001.

[31] C. Jones, D. Shell, M.J. Mataric, and B. Gerkey. Principled Approaches to the Design of Multi-Robot Systems. *Proc. of the Workshop on Networked Robotics, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, 2004.

[32] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of computer computations (Proc. Sympos., IBM Thomas J. Watson Res. Center, Yorktown Heights, N.Y., 1972)*, pages 85–103. Plenum, New York, 1972.

[33] Yoonsoo Kim, Da-Wei Gu, and Ian Postlethwaite. Real-time optimal mission scheduling and flight path selection. Submitted to IEEE Transactions on Automatic Control, accepted Jan. 12, 2007.

[34] Yoonsoo Kim, Da-Wei Gu, and Ian Postlethwaite. A comprehensive study on flight path selection algorithms. In *the IEE Seminar on Target Tracking: Algorithms and Applications*, pages 77–90, 2006.

[35] M.G. Lagoudakis, E. Markakis, D. Kempe, P. Keskinocak, A. Kleywegt, S. Koenig, C. Tovey, A. Meyerson, and S. Jain. Auction-based multi-robot routing. *Robotics: Science and Systems*, 2005.

[36] G. Laporte et al. *Classical and Modern Heuristics for the Vehicle Routing Problem*. Blackwell Synergy, 1999.

[37] Gilbert Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3):345–358, Jun. 1992.

[38] A. Le Bouthillier and T.G. Crainic. A Cooperative Parallel Meta-Heuristic for the Vehicle Routing Problem with Time Windows. *Computers & Operations Research*, 32(7):1685–1708, 2005.

[39] T. Maddula, A.A. Minai, and M.M. Polycarpou. Multi-target assignment and path planning for groups of uavs. In S. Butenko, R. Murphey, and P. Pardalos, editors, *Recent Developments in Cooperative Control and Optimization*. Kluwer Academic Publishers, 2003.

[40] Petter Ögren, Sven-Lennart Wirkander, Anna Stefansson, and Johan Pelo. Formulation and Solution of the UAV Paparazzi Problem. In *AIAA Guidance, Navigation, and Control Conference and Exhibit; Keystone, CO, USA*, 21–24 August 2006.

[41] D. M. Panton and A. W. Elbers. Mission planning for synthetic aperture radar surveillance. *Interfaces*, 29(2):73–88, 1999.

[42] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity*. Dover Publications Inc., Mineola, NY, 1998. Corrected reprint of the 1982 original.

[43] A. Pikovsky, P. Shabalin, and M. Bichler. Iterative Combinatorial Auctions with Linear Prices: Results of Numerical Experiments. *E-Commerce Technology, 2006. The 8th IEEE International Conference on and Enterprise Computing, E-Commerce, and E-Services, The 3rd IEEE International Conference on*, pages 39–39, 2006.

[44] W. Ren, R.W. Beard, and E.M. Atkins. A survey of consensus problems in multi-agent coordination. *American Control Conference, 2005. Proceedings of the 2005*, pages 1859–1864, 2005.

[45] DP Spanos and RM Murray. Motion planning with wireless network constraints. *American Control Conference, 2005. Proceedings of the 2005*, pages 87–92, 2005.

[46] C. Tovey, M.G. Lagoudakis, S. Jain, and S. Koenig. The generation of bidding rules for auction-based robot coordination. *Proceedings of the 3rd International Multi-Robot Systems Workshop*, pages 14–16, 2005.

[47] R. Zlot and A. Stentz. Market-based Multirobot Coordination for Complex Tasks. *The International Journal of Robotics Research*, 25(1):73, 2006.