



UNIVERSITÀ DI PISA

Facoltà di Ingegneria

Tesi di laurea specialistica in Ingegneria dell'automazione
Relazione sintetica

Cooperative control of 3D mobile agents with limited sensing capabilities

RELATORI:

Prof. Ing. Andrea CAITI

Dott. Ing. Emanuele CRISOSTOMI

CONTRORELATORE:

Prof. Ing. Alberto LANDI

SUPERVISORE ESTERO:

Dr Guy-Bart STAN



CANDIDATO:

Stefano FALASCA

Anno accademico 2008 - 2009

Sommario

Lo studio dei sistemi cooperanti richiede l'uso di tecniche derivate da diverse branche dell'ingegneria, delle scienze informatiche e della matematica. Questo lavoro propone una formulazione matematica da usare per la definizione e l'analisi di una particolare classe di gruppi di veicoli. Si prendono in considerazione flotte di agenti nello spazio 3d; vengono analizzate le possibilità di cooperazione quando la comunicazione è limitata all'acquisizione attraverso sensori, con campo visivo limitato, della posizione di altri veicoli. Allo scopo di studiare gli effetti dei limiti di comunicazione viene formulato il problema del *rendezvous*, per il quale è fornita una soluzione. I risultati ottenuti sono in linea con quelli tradizionalmente ottenuti nell'ambito del controllo cooperativo.

Abstract

The field of cooperative control for groups of vehicles lies within several different engineering, computer sciences and mathematical branches. This work proposes a mathematical framework to be used in defining and analysing a particular class of vehicles' groups. Fleets of agents moving in the 3d space are considered; their ability to cooperate when no communication but position sensing—using a limited field of view—is analysed. The statement of a slightly modified version of the *rendezvous* problem for the considered system and an algorithm to solve it are used as a mean to explore the effects of the communication limits. The obtained results are on the same level with traditional results of cooperative control.

The synthesis of cooperating behaviours for groups of mobile robots has been capturing the interest of several researchers in the last years. The field of cooperative control is considered to be a relatively new one, even if it has been pioneered in the late 1980s by, among few others, P.K.C.Wang. Thanks to the big advances in computer technology and microelectronics the realisation of low cost vehicles is now possible and the fields of application of cooperating groups of vehicles has grown. For this reason there has been increased research interest in systems composed of multiple autonomous mobile robots exhibiting cooperative behaviour. The theoretical framework supporting such studies is still in its formative stages, although some tools (e.g. digraphs, invariance principles, etc.) are almost directly borrowed from other disciplines and used to tackle a certain set of problems that arise.

In the preliminary stages of the research regarding such field, a great source of ideas came from biological systems. Interesting cooperative behaviours arise in biological networks; this is true regardless of the level of resolution the observation of biological systems has: cooperative behaviours can be found all the way from interactions among molecules and cells to the behavioural ecology of animal groups. Influences from the biological world can be easily recognised in the definition of goals too. While first works (e.g. *ibid.*, Maeda) explored cooperation issues in order to tackle practical problems such as robot-aided manufacturing or space exploration; the focus eventually moved towards the study of which class of problems can be dealt with via cooperative control techniques. Of course, several practical scenarios have been proposed in the related literature spanning from the surveillance of geographical areas to the accomplishment of tasks such as fire extinguishing. Practical applications of cooperating systems usually relies on the composition of several base behaviours previously treated in literature.

Cooperative control is aimed at finding control laws for autonomous vehicles; the term *autonomous*, here, indicates their ability to take decisions to some degree. Autonomous vehicles are not subject to direct human control given neither by on-board operators—if any—nor by remote ones. This is the main reason why autonomous vehicles are useful for accomplishing hazardous tasks (e.g. exploring contaminated areas) or in hostile environments (space or submarine exploration).

The field of cooperative control lies within several different engineering, computer sciences and mathematical branches. Control theory, optimization, networking and parallel and distributed computation are usually involved to some extent in every work on cooperative control. Moreover every cooperative system is a complex engineering system during the realisation of which several problems have to be solved (e.g. designing an appropriate computation platform, real time implementation of the algorithms, design of

vehicles and their actuation, design of low level control laws for vehicles, appropriate sensorization, etc.). For this reason it is clear that a certain level of abstraction is needed in treating cooperative control, especially from a theoretical point of view.

The theory of cooperative control mainly involves two aspects, namely: the definition of *models* and the accomplishment of *tasks*; the explicit link between the two is to be found in *algorithms*. The modelling portion is aimed at mathematically defining the system and the means of interactions; it usually involves a geometrical definition of the vehicle and of its sensing capabilities (sensors are almost always supposed to be the *payload* of vehicles), the topology and mechanisms of communication, the means of controlling vehicles and some computational structure aimed at permitting cooperation. As for the tasks there is a given set of goals that become an accepted *basis* of behaviours; rendezvous, flocking, deployment, consensus and target tracing are some examples. An algorithm is a mean of exploiting the modelled system—and particularly the given computational structure—in order to achieve a given task.

It is easy to see that what makes the field of cooperative control different from that of distributed computation is the assumption that the processing units are vehicles; thus it is not surprising that geometrical aspects play an important role in defining both the model and the tasks. Moreover as vehicles move—or, what is the same, while the network evolves—interconnection topology changes; for this reason models are specifically thought to account for spatial-related issues. Geometrical and communication issues are now briefly covered.

Vehicles are generally thought to move on a 2d space¹. More precisely: models are often stated, in their early stages, as being relative to the 3d space whilst it is rare to find trace of 3d motions when algorithm design is concerned. In Bullo/Cortés/Martínez—that is the most comprehensive resource currently available on this topic as far as the author knows—3d models are formulated while only some results are effectively solved in the 3d space; the most important ones—as, for instance, deployment, to which an entire chapter is devoted—are only stated for the 2d case. The author’s belief is that the reason for this is to be found—apart from the increased complexity deriving from a 3d formulation—in the possibility of casting several cooperative control-related problems into path planning tasks: obstacle avoidance, in fact, is often considered within this field and traditional path planning algorithms are intended to work within a 2d space. Of course in-

¹for the sake of precision it should be said that “vehicles are thought to lie on a plane”, since vehicles’ orientation is often taken into account when non-holonomic constraints are assumed to be in place.

creased complexity and simulation issues plays an important role as causes for such trend².

As we pointed out before it is almost mandatory to consider changes in interconnections' topology when cooperative control is concerned. For this reason constraints imposed by limits in communication capabilities of vehicles are traditionally explored. While some research works assume all to all communication among vehicles it is more frequent to find studies where whether two vehicles can communicate or not depends on their relative distance being smaller than a prescribed value. Communication constraints are often tightened when particular emphasis is put to the environmental characteristics: a non convex—possibly filled with obstacles—environment is often supposed to interact with communication capabilities in that vehicles not being too distant from each other can communicate only if their line of sight is not obstructed. The constraints described hitherto are completely dictated by geometrical considerations. On the other hand lies a different approach to the problem; it is based on considering the network topology as being dictated by an external super-entity; conditions on links' presence and persistence are usually made (e.g. “if the network topology is such that a spanning tree for the network always exists. . .”). Again P.K.C.Wang depicted a very detailed yet complex scenario by introducing a limited field of view in the 3d space.

However, as far as the author knows, limited field of view problems are not properly taken into account in the cooperative control related literature up to now. This is mainly because completely characterising implications of having a limited field of view is far from being a complete task. Salaris et al., for example, is a very recent paper taking into account such complex topic; it characterises optimal trajectories for the parking problem; yet a deeper understanding of the implications may be thought to be needed in order to allow the use of such constraints for a task as complex as cooperative control. why another work on cooperative control

In addition the author's belief is that the 3d space should be given a bigger attention; some preliminary works are necessary in order to lead the 3d case towards a maturity comparable to that of the 2d case. It is possible to foresee that several purely geometrical-based studies have to be carried out before being able to take into account complex topics such as dynamics and non-holonomic constraints. In this regard the robotic modelling tools may play an important role.

Two main threads trigger the interest of the author as far as the on-going debate on cooperative control is concerned: the exploration of what it is possible to achieve by means of groups of cooperating vehicles and the contributions

²it is probably needless to say that prototyping a land vehicle is far simpler than trying aerial, underwater or space vehicles.

definition—if not of a proper, complete mathematical framework—of what mathematical models are required to describe and how they are supposed to do so.

As for the exploration of abilities of cooperative control systems, a particular framework is considered. Unidirectional, content-limited communication is supposed to be in place; this means that not only the network topology is geometrically dictated (in a very sharp way, as we will see), but also the content of communication is supposed to be limited to partial information regarding the vehicles' state (communication will be, in fact, limited to have the position of vehicles as object: no algorithmic information will be transmitted at any time). It will be seen that such limited conditions allows, however, the vehicles to cooperate to some extent.

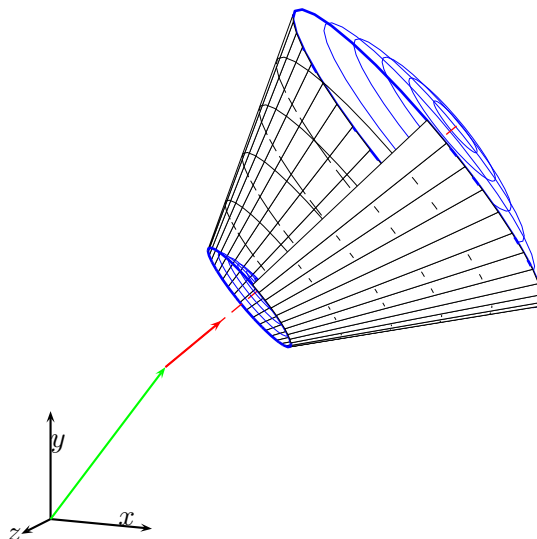
As it has been pointed out above, different mathematical models are used for different portions of the overall system; the author's belief is that the widely accepted dichotomy between what is network-related (and usually represented by means of graphs) and algorithms has to be overcome. More precisely network models should be used whenever problems as channel failures, propagation delays, transmission errors and similar problems have to be taken into account—which is usually not the case; on the contrary, topological issues should be directly considered in defining, designing and analysing both tasks and algorithms. A frequent approach, on the contrary, consists of checking the proposed algorithm for desirable properties when topological variations occur; even when this is not the case models do present a division between the network aspects and the algorithmic ones.

The contributions of this work are threefold: first of all it considers 3d agents having limited field of view and with communication capabilities limited to position sensing in order to explore their ability to cooperate; secondly, when it comes to define the problem and analyse it, a novel approach is used—of course, several assessed techniques are largely used throughout the work; it finally solves the *rendezvous* problem within the proposed framework as a mean to check the proposed framework for coherency.

We now give an overview of the problem of interest. There is a set of dimensionless vehicles moving on the 3d space. The environment within which agents move contains nothing but the agents themselves. Vehicles can change their position and attitude in an arbitrary way with discrete timing; a first order, fully actuated model with no constraints on inputs will represent them. Vehicles choose their new position according to some algorithm based on some sensed information. Two kinds of sensing are possible: each vehicle can sense its actual position and attitude at every time as well as the position of a certain set of other vehicles; such set is determined by the field of view of the sensing vehicle and by its current position and attitude. This field of view is the portion of space delimited by

assumptions and definitions

Figure 1: A sensing agent with its sensing cone.



a cone and within some given minimum and maximum distance. Perfectly precise information are assumed to be available at any time. Since the algorithms agents are intended to use in this work are not too complex, it is assumed that every agent has enough computational capabilities to execute them in order to convert the information sensed at time k into the new position that will be occupied at time $k + 1$.

Vehicle representation in space is accomplished through the definition of the notion of *agent*; such mathematical structure is aimed at knowing about both the position and attitude of a vehicle. 5 parameters are needed in order to completely represent the set of possible configurations for vehicles since only position and heading are considered to be of interest. The reason why we need to distinguish among agents having different headings is that we want the sensing field to depend on attitude; a *sensing axis* is defined for an agent and it represents the symmetry axis of the vehicle's field of view. A *sensing agent* is defined as the combination of an agent and a set of parameters that define the visual *cone* of vehicle's sensors; such parameters define the minimum and maximum distances m and M that are allowed between a *sensing agent* and a sensed vehicle as well as an angle α that defines the aperture of the field of view; figure 1 shows a *sensing agent*. Given a *sensing agent* the *set of sensed vehicles* is the set of all vehicles that lie within the sensing cone of the agent; this is easily recognised to be a set such that whether a vehicle belongs to it or not does not depend on its attitude.

While graphs are traditionally used as a mean of describing network connection within the cooperative control literature, we propose a different way of using them. In the context of parallel computing *Directed Acyclic Graphs* (DAG) are used in order to represent the direct dependence of units of computational code from partial results produced by different routines; similarly they can be used to represent how computation carried by a given agent at a given time relies on sensed information.

In this context a node of a graph represents the joint information “vehicle-time”, whilst the edges that incides on a node show the dependence of the computation represented by the node on (the sensed portion of) the consequences of some other computation. For instance if a node 1 represents agent a at time 1 and a node 2 represents agent b at time 2, an edge pointing from 1 towards 2 means that agent b needs to know the results of what agent a ecided at time 1 in order to take its decision at time 2; more precisely it needs to know the position of agent a at time 1, since this is the only piece of information it can sense.

Since vehicles move, the topology of visual connections varies in time. Whether a visual link is in place between a *sensing agent* and a vehicle or not depends both on the *sensing agent*’s decisions and on other agents’ choices. The DAG representing a group of vehicles takes partial account of such changes.

So far we defined the scenario we work within and the main mathe- analysis
 matical entities we are going to work with. The next step to be taken is analysing the cooperative control problem in general; we will subsequently move our attention towards the definition of a task for the group of vehicles to accomplish.

As being in visual contact is the only way in which a cooperative behaviour can be established, keeping (part of) the visual links that are in place—if any—is the only mean to maintain cooperation. The first concern when designing cooperative algorithms is to devise some connection maintaining strategy.

The main difficulty in trying to find such a strategy is due to the visual relation being asymmetrical; that is to say: if agent a senses agent b and tries to keep connectivity with it, since b —in general—does not sense a , it is difficult to guarantee for b not to act in an *opposite* way. Nonetheless it is possible to find such strategies and, moreover, the one we present is such that it can be recursively, indefinitely applied. What we mean by that is that, as for the example above, b does not prevent a from following it and, furthermore, a —while following b —does not prevent other vehicles from following it, and so on.

In particular our strategy is defined in terms of a set of possible movements for vehicles to allow other vehicles to follow them and a second set of possible movements for a vehicle to follow another vehicle. The first set does not depend on any information, the second one depends on relative position of the two agents and—what allows the recursion described above—is a subset of the former.

Since there exists a whole set of points belonging to the admissible set for each vehicle at each time, and since the available characterization of such set turns out not to be an operative one, a convex subset of it is worked out. This convex approximation of the feasible set (shortly *the feasible set*) is built in this way: as for the position of the *sensing agent*, depending on the relations between sensing parameters and the relative position between the *sensing agent* and the vehicle it has to keep connectivity with, a feasible point such that every position within a certain distance from it is admissible is generated; based on the same information an interval for each of the two angles representing the *sensing agent's* orientation is worked out. The amplitude of the convex approximation of the feasible set can be expressed in terms of the maximum distance from the generated points and by the length of the angles' intervals; these values depend on both the sensing parameters and on relative position between the considered agents.

The described strategy allows each *sensing agent* to *follow* another vehicle. Since during its movements it is possible for an *agent* to sense more than one vehicle at a time, each *agent* may have to choose which one of the sensed agent it wants to keep connectivity with. This work does not consider the problem of maintaining joint connectivity when possible; it focus, instead, on the exploiting of a communication structure where each *agent*—except one—has exactly one *leader* at each time. The *agent's* leader may, of course, change during the evolution of the network.

It is important to notice that a connection maintaining strategy needs some connections to be in place. What we have done so far, then, is valid when the starting configuration of the group of vehicles is such that it is possible to assign a sensed leader to each vehicle and, moreover, by doing so the group results to be fully connected or—and what is the same—the connection maintaining strategy applies to every connected component of the group of agents as for their initial configuration.

When leader changes happen several problems may arise. Until now—assuming an appropriate starting configuration—we only had to consider an *agent* and its leader in order to keep the full connectivity. If an *agent* changes its leader, however, the overall connection is not guaranteed, even if the connection with the new leader is maintained. Conditions on the DAG are obtained in order to assess the acceptability of leader assignments.

Based on this conditions a distributed strategy for leader changing is worked out; the strategy requires an a priori ordering among vehicles to be in place.

The defined framework and the analysis carried out are used to solve the *rendezvous* problem; the definition of the task, in this context, is given as: every agent has to reach a final position such that it senses a given *target* point, this has to be accomplished while maintaining the full connectivity. goal

It is possible to express the described task in terms of the Gram matrix of the configuration of points that represents the *agents'* positions; in this way a set of conditions are devised that allow to decide whether a configuration of points can solve the problem or not. Such conditions can be verified on an iterative basis, i.e. they can be converted into a set of conditions regarding a portion of the Gram matrix—such sub-matrix takes into account one *agent* and its leader. However, the devised conditions need, to some extent, to take into account other vehicles' position too. More precisely the iterative checking of conditions can be thought as a mean of solving the following problem: for each *agent*, given a point that represents the position of its leader, we find a second point from which the *agent* can sense both its leader and the target; moreover—and this is where other vehicles are taken into account—the second point has to be such that the *agent* can act as a leader for a third *agent*, i.e. the problem has to have a solution if a solution of the problem is taken as an input (this is a *recursivity* condition). This is, as far as the problem statement is considered, analogous to what happened for the connection maintaining strategy; solving this *recursive* problem is, however, quite a different task.

An algorithm for each vehicle—apart from one—to execute in order to accomplish the described task is proposed here. It is based on the following idea: while maintaining visual contact with its leader, each agent follows its leader until this leader stops; when one agent stops moving every agent following it—if such agents exist—moves itself towards an *appropriate* distance from the leader while, again, maintaining the visual contact with it; when an agent reaches the *appropriate* distance from its leader it starts moving on an ellipse until it reaches a point such that the *pair visibility problem* is solved and it can act as a leader in the *pair visibility problem* for other agents. The ellipse is proven to be such that, by moving on it, it is possible to satisfy the conditions for *recursivity* described above. algorithm

The overall algorithm starts by executing the base algorithm *follow*. The agent's leader will be followed in its movements until it stops. In every moment, if the leader starts moving again, the base algorithm *follow* will be put in charge again³. When the leader stops moving the base algorithm *change distance* is executed. It is in charge of changing the distance between

³the same thing happens when the leader changes.

the agent and its leader. The distance needs to be changed when it is not possible, by following a circular path around the leader, to engage the desired ellipse; when the distance between the agent and its leader is such that this can be done, the base algorithm is changed. The new base algorithm that will be put in charge when an appropriate distance is reached will cause the agent to follow a circular orbit around its leader until it engages the ellipse. Then, the agent starts moving on the ellipse until it solves the partial problem with an appropriate distance from the origin. It then stops, thus holding its position.

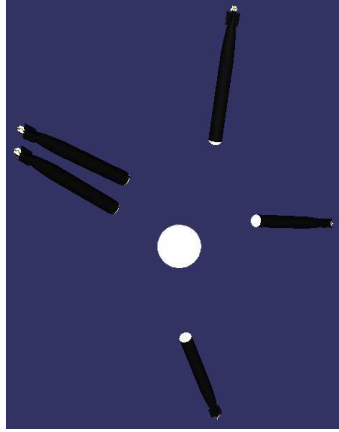
A C++ software has been developed in order to experimentally validate the described results. Although it has been designed as a proper piece of code and realised following established techniques in order to make it extensible, maintainable and, generally speaking, “well coded”, here it is simply regarded as a mean of obtaining 3d visualisation of agents’ group behaviours.

The realised simulator permitted to check the effective behaviour of several groups of vehicles. The very nature of the proposed algorithm for *rendezvous* is such that vehicles follow each other until some of them resolve a portion of the overall problem; for this reason it is possible to observe two distinct phases: during the first one agents move in a group towards the target following an overall leader that freely moves; in the second one agents moves all around the leader and the target in order to find a solution to portions of the overall tasks, this last behaviour is executed following a hierarchy among vehicles. As it has been discussed above the partial solutions are guaranteed to add up to a global solution. discussion

While following each other, vehicles exhibit a behaviour that should be familiar to who knows the *averaging control and communication law*; it is an algorithm for *rendezvous* that is intended to be used in a framework completely different from the one we are working with; nonetheless during its execution—as it happens here—sub-groups of agents tend to converge towards a common trajectory. Although a more powerful communication system (bidirectional) is used with *averaging control and communication law*, connection is not guaranteed to be maintained in that context as it is here. Finally it is author’s feeling that the degrees of freedom given by the convex feasible set discussed for connection maintaining—that plays a dominant role here—can be exploited in order to let the vehicles flock with a determined geometrical configuration during this phase.

As for the second phase of execution, simulation showed that every vehicle tends to stop at the same distance as each other from the common target; this can be seen in figure 2. This does not directly follow from the theoretical analysis of the algorithm. The author’s belief is that this is affected by two main aspects. First of all the algorithm implementation—used

Figure 2: The end configuration for a simulation



within simulations—requires to define stop conditions for the base algorithm that cause vehicles to orbit on an ellipse; these conditions can be defined in several ways and the end position assumed by the vehicle depends both on them and on the orbiting strategy. Moreover the set of points solving the portion of the overall problem considered by each agent depends on sensing parameters.

The present work established some conditions for vehicles moving on a 3d space to cooperate when sensing other vehicles within a limited field of view. Even if a non conventional framework of description and analysis has been proposed—and used—in order to conduct this study, obtaining the results of this work did not require the use of particular mathematical tools. Nonetheless the goals reached are on the same level with traditional results of cooperative control as far as both the task requirements and the obtained behaviours are considered. conclusions

Possible future developments of the proposed work are many-fold. The work can be extended by using second order models for the agents; author's belief is that Notarstefano et al., for instance, can be used as a basis in order to do so; several difficulties, however, would be tackled because of the limited field of view. Designing collision avoidance algorithms for the groups of agents of interest can be an interesting problem; limits on communication, however, may bar chances of success in trying to guarantee collision-free movements. Moreover, the presence of measurement noise should be considered in future works. The use of non-holonomic vehicles in this framework is a very appealing topic, yet far from being approachable at the current stage. Finally the peculiar modelling of the algorithm is thought to be worth some further investigations.

Bibliography

- Bullo, F./Cortés, J./Martínez, S.:** Distributed Control of Robotic Networks. Princeton University Press, 2009, Applied Mathematics Series, Electronically available at <http://coordinationbook.info>, ISBN 978-0-691-14195-4
- Maeda, J.:** Development and Application of Cooperative of Cooperative Robots for Construction Work. In Proc. 16th International Symp. on Industrial Robots. 1986, 679–689
- Notarstefano, Giuseppe et al.:** Maintaining limited-range connectivity among second-order agents GG. 2006
- P.K.C.Wang:** Interaction dynamics of multiple mobile robots with simple navigation strategies. Journal of robotic systems, 6 1988, Nr. 1, 77–101
- Salaris, P. et al.:** Optimal paths in a constrained image plane for purely image-based parking. In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference. settembre 2008, 1673–1680