Literature Review on Fault-Tolerant and Cooperative Control of Unmanned Aerial Vehicles

Ban Wang

Supervisor: Dr. Youmin Zhang

In the Department
of
Mechanical & Industrial Engineering

Presented in Partial Fulfilment of Requirements
For the Degree of
Doctor of Philosophy (Mechanical Engineering) at
Concordia University
Montreal, Quebec, Canada

September 2015

Abstract

Cooperative control of a team of small, inexpensive unmanned aerial vehicles (UAVs) is of great interest in civilian and military applications, including border patrol, search and rescue, surveillance, hostile territory monitoring, geographical mapping and communications relaying. It requires state-of-the-art technologies to realize the required levels of autonomy, efficiency, and reliability. This report summarizes a review of the literatures on different strategies for cooperative control of networked vehicles under normal and fault conditions. In total, more than 60 references in open literature are compiled to provide an overall picture of the historical, current, and future research and development in this active research and development area.

Contents

1 Introduction

Research on control of multi-vehicle systems performing cooperative tasks can date back to the late 1980s, initially beginning in the field of mobile robotics [1]. With the development of inexpensive and reliable wireless communication systems and advances in computing power, research in this field increased substantially in the 1990s. In the late 1990s and early 2000s, cooperative control of multiple aircrafts, especially unmanned aerial vehicles, became a highly active research area in the United States [2]. The motivation for this research may be traced to emergence of applications which are repetitive or dangerous. Groups of UAVs are of special interest for their abilities to coordinate simultaneous coverage of large areas, or cooperate to achieve goals such as border patrol missions, search and rescue operations, surveillance and reconnaissance actions, hostile territory monitoring, convoy protection, forest fire detection, power lines and pipelines inspection, large accident and disaster investigation and relief, cooperative environment monitoring, cargo transport, data and image acquisition of areas, crop spraying and communications relaying. On the other hand, observations made based on natural behavior of animals operating as a team have inspired scientists in different disciplines to investigate the possibilities of networking a group of systems to accomplish a given set of tasks without requiring an explicit supervisor [3]. Achieving cooperative control, however, is quite challenging. Many issues must be addressed in order to develop a working cooperative team, such as action selection, coherence, conflict resolution, and communication. Furthermore, these cooperative teams often work in dynamic and unpredictable environments, requiring the group members to respond robustly, reliably, and adaptively to unexpected environmental changes, failures in the inter-agent communication system, and modifications in the cooperative team that may occur due to mechanical failures, or the addition or removal of robots from the team by human intervention [4]. To overcome all these ever increasing requirements and challenges, advanced fault-tolerant control schemes constitute the key enabling technologies for ensuring reliable and efficient operation of cooperative systems.

Emerging results in cooperative vehicle control are presented via discussion of these topics, including particular requirements, challenges, and some promising strategies relating to each area ranging from pure simulation to control of multiple vehicles. However, given the significance of cooperative control algorithms and their integration with fault-tolerant control techniques, there is no well-organized, comprehensive survey publication about the state of current research in this new and active area along with opportunities for the future. This critical literature review reports an updated and more comprehensive review of the literature on different algorithms for cooperative control under normal and fault situations. The remainder of the report is organized as follows: The existing techniques to FTC and FDD are briefly introduced respectively in Section 2. In Section 3, cooperative control of multiple vehicles under normal and fault situations is reviewed. Finally, conclusions and future directions are drawn in Section 4.

2 Fault Diagnosis and Fault-Tolerant Control Systems

This section is mainly to give a brief overview of fault detection and diagnosis (FDD) and fault-tolerant control (FTC) techniques including basic concepts and available FDD and FTC methods as well as their classifications.

2.1 The Motivation of Fault-Tolerant Control

Nowadays, control systems are involved in nearly every aspects of our lives. They are all around us, but their presence is not always really apparent. Control systems are present in every industry, however for safety-critical applications such as aircraft, spacecraft, nuclear power plants, chemical plants, etc. they must have capabilities beyond conventional control systems where the cost and consequences of a malfunction is too great. As a result, there is an increasing demand for safety and performance requirements in modern control systems.

2.2 Definition of Fault-Tolerant Systems

Since the terminology used in this field is not unique and differs among authors, this section starts with a brief definition of some terms and expressions frequently used throughout the literatures.

As the systems of interest are said to be fault-tolerant, let us first clarify the terminological distinction between a fault and a failure [5].

"A fault is an unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable/usual/standard condition." [5]

Based on this definition, a fault corresponds to an abnormal behavior of the system, which may lead to a small reduction in efficiency, but could also lead to overall system failure (defined later). Finally, faults may be small or hidden, and therefore be difficult to be accurately predicted in time, and to be prevented.

In contrast to fault,"a failure is a permanent interruption of a system's ability to perform a required function under specified operating conditions" [5]. Resulting from one or more faults, a failure is therefore an event that terminates the functioning of a unit in the system. Thus, a failure is more severe than a fault. However, these terms are dependent on relativity. For instance, an actuator failure occurs on one of the ailerons leaving this control surface unusable. Relative to the overall control, the failure can be interpreted as a fault. This is because by using the rest of control surfaces differently, the aircraft is still controllable.

In the fault-tolerant control literature, faults are classified according to their location of occurrence in the system: actuator faults, sensor faults and component faults.

Faults in actuators range from loss of partial control effectiveness to a complete loss of control [6]. For example, stuck is one of complete loss of control situations which produces no actuation regardless of the input applied to it. Since an actuator is usually regarded as the entrance to the plant, actuator faults will have severe effect on the system performance. Moreover, it is generally difficult to duplicate actuators in the system in order to achieve increased hardware redundancy and reliability due to their high cost and large size and weight.

Sensor faults represent incorrect readings from the sensors due to malfunction in sensor circuits or transducers. Similar to actuator faults, sensor faults can also be subdivided into partial loss and total loss. Bias, freezing, drift, loss of accuracy and calibration error are some common faults in sensors [7]. Fortunately, sensors can be duplicated in the system to increase fault tolerance due to their smaller sizes and weights. The so-called "majority voting" method can be utilized to pinpoint the faulty sensor. Moreover, faulty sensors can be separated from the rest of the plant and solved independently with the use of sensor redundancy.

Component faults are faults in the components of the plant itself. These faults represent changes in the physical parameters of the system, for instance, mass, aerodynamic coefficients, damping constant, etc., which are often due to structural damage. They often result in a change in the dynamical behavior of the controlled system.

Further more, faults can also be classified as additive and multiplicative, with respect to the way they are modelled. Additive faults result in changes only in the mean value of the system output signal. Whereas, multiplicative faults result in changes in variance, correlations of the system output signal as well as changes in the spectral characteristics and dynamics of the system [6]. Additive faults are suitable for representing component faults in the system, while sensor and actuator faults are in practice most often multiplicative by nature [8].

2.3 Fault-Tolerant Control Systems

A fault-tolerant control system (FTCS) is a control system that can accommodate system component faults and is able to maintain stability and acceptable degree of performance not only when the system is fault-free but also when there are component malfunctions [6]. Fault-tolerant control system prevents faults in a subsystem from developing into failures at the system level. More precisely, fault-tolerant control systems are control systems that possess the ability to accommodate system component failures automatically. So it is of great importance to design control systems which are able to tolerate against potential faults in the system.

Generally speaking, FTCS can be classified into two types: passive FTCS (PFTCS) [9, 10] and active FTCS (AFTCS) [11, 12, 13]. A particular approach to be employed depends on the ability to determine the faults that a system may undergo at the design phase, the behavior of fault-induced changes, and the type of redundancy being utilized in the system.

In PFTCS, controllers are fixed and are designed to be robust against a class of presumed faults [14]. This approach requires no on-line detection of the faults, and is therefore computationally more attractive. However, if these deviations become excessively large and exceed the robustness margin, some actions need to be taken. Therefore, an active fault-tolerant control architecture is needed in order to achieve extended fault-tolerance capability.

As opposed to PFTCS, the design of AFTCS is based on controller redesign, or selection of predesigned controllers. This technique usually requires a fault detection and diagnosis (FDD) scheme that has the task of detecting and localizing the faults if they occur in the system [5]. To be more precise, the FDD part uses input and output measurements from the system to detect and localize the faults. The estimated faults are subsequently passed to a reconfiguration mechanism which changes the parameters and/or the structure of the controller in order to achieve an acceptable post-fault system performance.

Since AFTCS involves a significant amount of on-line fault detection, real-time decision making and controller reconfiguration, it accepts a graceful degradation in overall system performance in the case of faults and the trade-off between the stability and the performance of the overall system should be carefully considered. The critical issue in any AFTCS is the limited amount of time available for the FDD and for the control system reconfiguration [13].

Many methods have been proposed to solve the problem of fault-tolerant control. The existing reconfigurable control approaches are classified and summarized in Figure 1. Interested readers are referred to a recent review paper [13] for detailed information on reconfigurable FTCS and the classifications in this field.

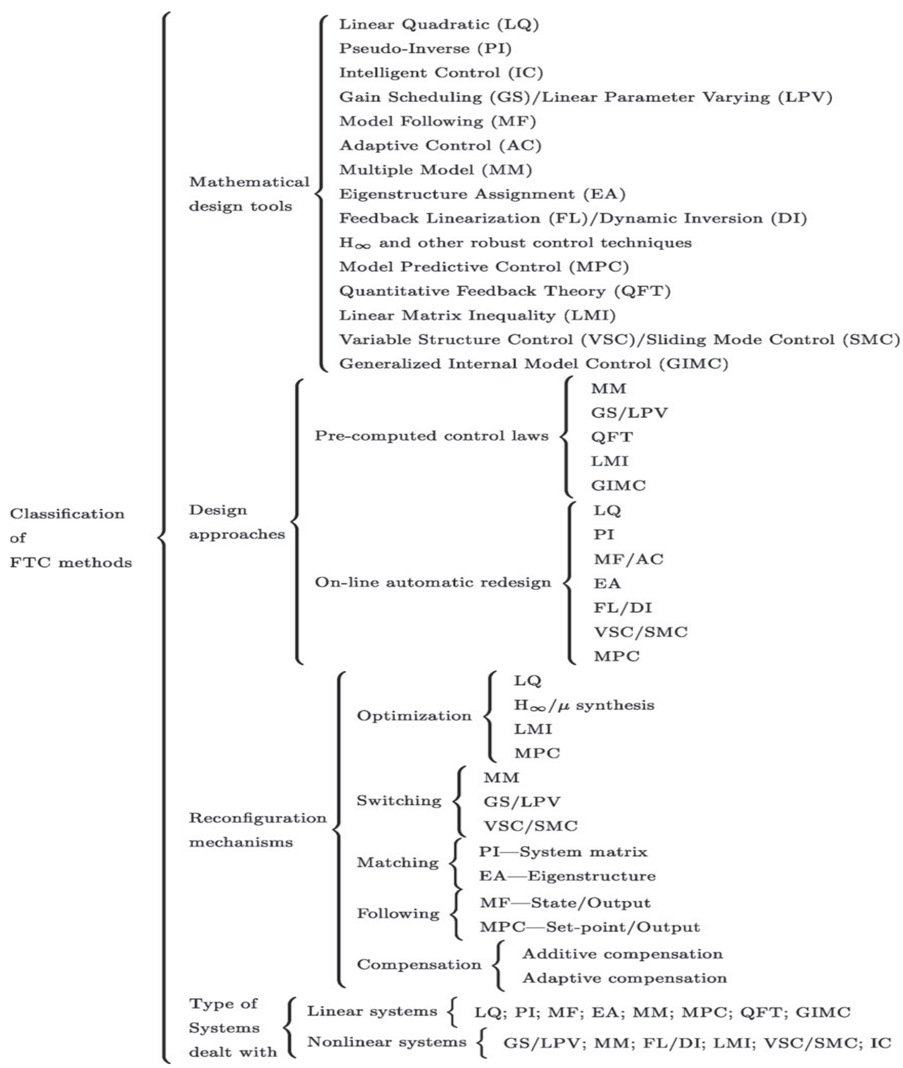

Figure 1: Classification of AFTCS [13]

2.4 Fault Detection and Diagnosis

One of the critical issues of AFTCS is on-line reconfiguration of the controller. For this to be possible, detailed information about fault-induced changes is required [6]. The scheme monitors system performance to detect the occurrence of faults, and to determine their magnitudes that is composed of multiple parts for detection, isolation, and in some cases estimation or identification of faults.

Fault detection indicates that something is wrong in the system, for example, the occurrence of a fault and the time of the fault occurrence [13]. From the FDD point of view, faults can be classified into two categories: abrupt faults and incipient faults [5]. Generally speaking, detection of an abrupt fault is easier than an incipient fault, but an abrupt fault might induce severe consequences for the system, due to its instaneous nature. However, an efficient FDD scheme should be highly sensitive to faults and at the same time robust to model uncertainties, variations of system operating condition, and external disturbances. Fault isolation determines the location and the type of the fault. Fault identification determines the magnitude of the fault. The isolation and identification parts are together referred to as fault diagnosis.

Fault detection addresses the challenge of real-time monitoring the occurrence of fault in a system. Due to real-time requirements and the dynamic nature of the system, there is usually only a very limited amount of time available to carry out the post-fault model construction and control reconfiguration actions [13]. Therefore, the trade-off among different design objectives has to be carried out on-line in real-time.

The existing FDD algorithm can be generally divided into two main parts: model-based and data-based methods. These two approaches can further be classified as quantitative and qualitative methods. Since most of control techniques are model-based, and fault-tolerant controllers need to be designed on the basis of the mathematical model of the system being analyzed, particularly the post-fault model of the system, therefore, model-based approaches have wider range of applications. Moreover, as the complexity of control systems and the use of computers increase, quantitative model-based FDD systems become more prominent. Model-based methods rely on analytical redundancy by using explicit mathematical models of the monitored system to detect and diagnose faults. Methods commonly used to construct mathematical models are state estimation, parameter estimation, parity space, and combination of the first three. Figure 2 presents a generalized classification of FDD approaches available in the literature.

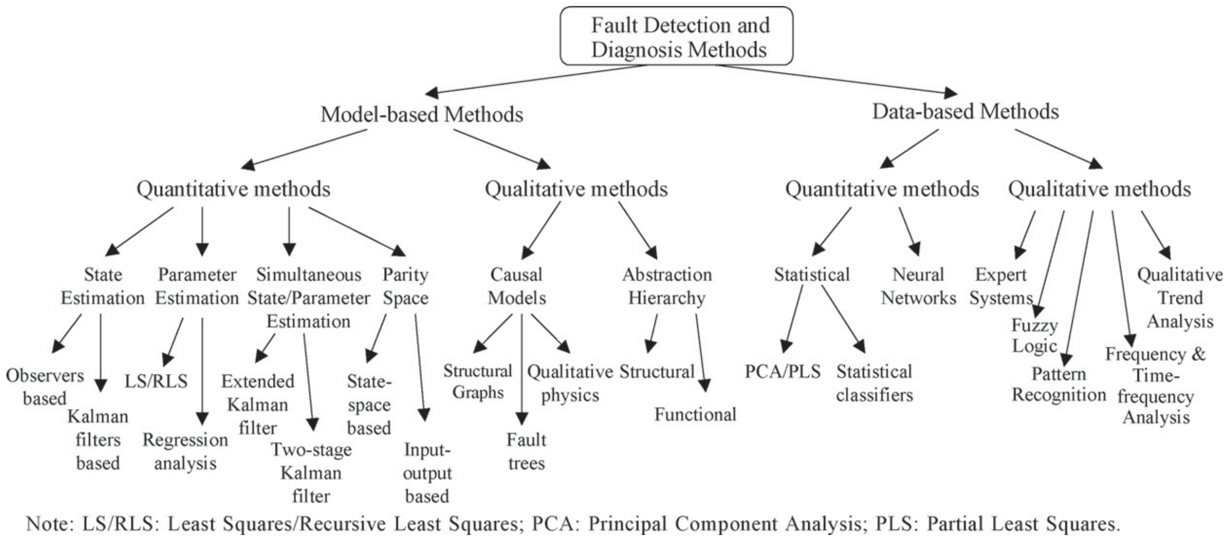

Figure 2: Classification of FDD Approaches [13]

3 Fault-Tolerant Cooperative Control on Unmanned Aircraft Systems

The field of unmanned aircraft systems (UASs) is an exciting segment of the aerospace industry. Recent technology advances and customer acceptance have led to the widespread employment of these amazing systems [15, 16, 17]. This section is organized to provide the readers with a basic understanding of an unmanned aircraft system and some fundamentals about control of unmanned aircraft systems.

The U.S. Department of Defense (DOD) defines an unmanned aircraft system as follows: A powered vehicle that does not carry a human operator, can be operated autonomously or remotely, can be expendable or recoverable, and can carry a lethal or nonlethal payload. Ballistic or semi-ballistic vehicle, cruise missiles, artillery projectiles, torpedoes, mines, satellites, and unattended sensors (with no form of propulsion) are not considered unmanned vehicles. Unmanned vehicles are the primary component of unmanned systems [18]. To be precise, an unmanned aircraft system comprises a number of sub-systems which include the aircraft (often referred to as a UAV or unmanned air vehicle), its payloads, the control stations, aircraft launch and recovery sub-systems where applicable, support sub-systems, communication sub-systems, transport sub-systems, etc.

Recent military and civil actions worldwide have highlighted the potential utility for unmanned aerial vehicles (UAVs). Both fixed-wing and rotary-wing aircraft have contributed significantly to the success of several military and surveillance/rescue operations. Future operations will continue to place unmanned aircraft in challenging conditions such as the urban warfare environment and forest surveillance. However, the poor reliability, insufficient autonomy level and heavy operator workload requirements of current unmanned vehicles present a roadblock to their success and practical applications. It is anticipated that future operations will require multiple UAVs performing in a cooperative mode, sharing resources and complementing other air or ground assets [19].

In fact, there is extensive literature on control and coordination for multiple unmanned systems, and application to tasks such as exploration, surveillance, search and rescue, mapping of unknown or partially known environments, distributed manipulation [20, 21, 22, 23, 24].

3.1 Definition of Group Cooperation and Coordination

A cooperative system is defined to be multiple dynamic entities that share information or tasks to accomplish a common, though perhaps not singular, objective [25]. This may be motivated by applications where single vehicle is impossible to accomplish the assignment. On the other hand, observations made based on natural behavior of animals operating as a team have inspired scientists in different disciplines to investigate the possibilities of networking a group of systems to accomplish a given set of tasks without requiring an explicit supervisor [3]. Some examples of such natural behaviors can be found in flocks of birds, swarms of insects, herds of quadruped, and schools of fishes. In this sense, such a control issue for a cooperative system is generally referred as to cooperative control. Cooperative control for multi-agent systems can be categorized as either formation control problems with applications to mobile robots, unmanned air vehicles (UAVs), autonomous underwater vehicles (AUVs), satellites, aircraft, spacecraft, and automated highway systems, or non-formation cooperative control problems such as task assignment, payload transport, role assignment, air traffic control, timing and search [26].

3.1.1 Formation Control

Formation flight is defined as intended motion of two or more flying objects, connected by a common control law, which is focused on achieving and maintaining a particular structure of the entire formation and collision avoidance [27]. Based on this property, a predefined trajectory is usually provided for the team motion.

As discussed in [28], various architectures and strategies have been developed in either centralized or decentralized methods in order to control and coordinate a multiple vehicles group, namely behavior-based, virtual structure, leader-follower, graph-based, and potential field approaches.

The beginning of work on formation flight can date back to 1977, Scholomitsky, Prilutsky, and Rodnin [29] worked on the concept of an infrared interferometer for a few flying objects. This study was continued by Lyberie et al. [29] for formation flight of spacecrafts, their maintenance and refueling in the early 1980s. Significant interest in formation flight started to develop in the late 1990s. The first formal study of formation flight control was by Wang and Hadaegh [30] in 1996, who analyzed the leader-follower architecture for spacecraft.

In the leader-follower structure, individuals in the formation follow one UAV, which is designated as a leader. A formation flight mission trajectory is loaded in the leader, and the followers track their leader. This structure is simple and widely implemented in multi-UAVs formation [31, 32, 33]. In this method, there is no explicit feedback from the followers to the leader and that is the disadvantage of this method.

Fabrizio et al. [33] develop a two-loop control system. The main objective of the inner-loop controller is to allow tracking of commanded velocity, altitude, and heading angle, and it actually operates as a preset autopilot for the formation management. In the outer loop, the formation controller generates a reference path command for the inner loop to follow the desired formation trajectory and to maintain the aircraft position inside the formation. In this literature, the authors also consider aerodynamic effects, for example, the vortex leaving from the trailing edge of the wing of the leader aircraft producing an up-wash on the wing of the following aircraft.

In [32], Roldao et al. present a strategy for real-time generation of formation trajectories using a leader-follower approach. This strategy differs from the standard approach of defining the desired distance vector in an inertial frame and can be used to obtain rich formation trajectories with varying curvatures between vehicles. By imposing adequate constraints on the motion of the vehicles, the generation of valid formation trajectories is naturally guaranteed without the need to explicitly compute the path parameters of the leader. Zhao et al. [34] addresses the application of model predictive control (MPC) approach for the leader-follower formation flight problem since MPC can handle constraints in relative ease. Under the robust decentralized unified MPC framework, a collision avoidance scheme is developed and extended to take care of any shape and small pop-up obstacles.

The virtual leader structure consists of replacing a formation leader, in the leader-follower shceme, by a virtual one. The entire formation is treated as a single entity. Each member of the formation receives the mission trajectory that is assigned to the virtual structure. One of the main drawbacks of this structure is the centralization, which may lead a single point of failure for the whole system.

In [35], Lalish et al. present an approach to formation tracking under the virtual structure scheme while treating the vehicle model as a nonholonomic particle. Lewis et al. [36] develop a control strategy using virtual structure to force an ensemble of robots to behave as if they were particles embedded in a rigid structure while maintaining a geometric configuration during movement.

Ren et al. [37] investigate a novel idea of introducing formation feedback under the scheme of virtual structure through a detailed application of this idea to the problem of synthesizing multiple spacecraft in deep space. It overcomes some drawbacks of virtual structure. For instance, when the virtual structure moves too fast for the spacecraft to track or the total system must sacrifice convergence speed in order to keep the spacecraft in formation, if there is no formation feedback from spacecraft to the virtual structure, the spacecraft will get out of the formation.

In the behavior-based control, each member of the group follows some rules to achieve the formation. It employs several behaviors for each agent and the final control is derived from a weighting of relative importance of each behavior. The first distributed behavioral model can be found in [38] by Reynolds. It is inspired from biologists study on collective motion of animals. Reynolds considered that each individual in a flock ought to follow some rules in order to perform the flocking behavior where the rules are: 1) Collision Avoidance, 2) Velocity Matching, and 3) Flock Centering. Upon these rules, Saif et al. [39] address the control problem of multi-UAVs flocking by using a behavior-based strategy.

Balch et al. [40] present a behavior-based approach to robot formation keeping. The formation behaviors are integrated with other navigational behaviors to enable a robotic team to reach navigational goals, avoid hazards and simultaneously remain in formation. In [41], Monteiro et al. design a distributed control architecture that generates navigation in formation, integrated with obstacle avoidance for a team of three autonomous robots using dynamical systems theory to model behavior-based formation control. This work is based on the so called Dynamic Approach to Behavior Generation.

In [42], Vadakkepat et al. combine fuzzy control with behavior-based architecture for the control of mobile robots in a multi-agent environment. The behavior-based architecture decomposes the complex multi-robotic system into smaller modules of roles, behaviors and actions. Fuzzy logic is used to implement individual behaviors to coordinate the various behaviors, to select roles for each robot and for robot perception, decision-making, and speed control.

In the graph-based control architecture, the dynamics of multi-agent system are linked to each other by a communication graph. The graph in this context refers to a collection of vertices or nodes and a collection of edges that connect pairs of nodes. The graph represents the allowed information flow between the agents which may be undirected, meaning that there is no distinction between the two nodes associated with each edge, or it may be directed from one node to another along the edge. In cooperative systems, any control protocol must be distributed in the sense that is respecting the prescribed graph topology. That is, the control protocol for each agent is allowed to depend only on information about that agent and its neighbors in the graph.

Olfati-Saber et al. in [43] provide a theoretical framework for analysis of consensus algorithms for multi-agent networked systems with an emphasis on the role of directed information flow, robustness to changes in network topology due to node failures, time-delays, and performance guarantees. This work is based on tools from matrix theory, algebraic graph theory, and control theory. In [44], Fax et al. consider the problem of cooperation among a collection of vehicles performing a shared task using inter-vehicle communication to coordinate their actions. They use algebraic graph theory to model the communication network and relate its topology to formation stability.

In [45], Qu et al. present an algorithm combining graph theory with virtual structure to maintain a specified formation configuration of multiple UAVs. There is only the local neighbor-to-neighbor information between vehicles as no explicit leader exists in the team. They also consider communication limits and measurement errors to improve robustness. Chen et al. [46] apply consensus algorithm with graph theory on multiple UAVs cooperative tracking. The UAV model used in this article is the particle motion model.

As to potential field approach, potentials define interaction control forces between neighboring vehicles and are designed to enforce a desired inter-vehicle spacing. This approach can also be used for obstacle and collision avoidance. The collision avoidance strategy could be used separately for the sense and avoid problem.

In [47], Paul presents a solution for formation flight and formation reconfiguration of UAVs based on an extended local potential field combined with a virtual leader approach. Rezaee et al. [48] propose a formation control of mobile robots based on virtual and behavior structures combined with potential field approach. Each robot is modeled by an electric charge, because of the repulsive force between the identical charges, each robot finds its desired position in the formation.

Eun et al in [49] address a hierarchical and automated control strategy applicable to the typical suppression of enemy air defense missions that involve multiple UAVs. The new path finding and planning method is based on the potential field theory. In [50], Nagao et al. describe a design of guidance law using potential functions for a swarm of UAVs. The proposed guidance law is derived by the artificial potential field method. They propose a new potential function that consists of steering, repulsive, and circular functions.

3.1.2 Non-Formation Cooperative Control

Besides the formation control, there is another higher level network cooperation control such as task assignment, timing and scheduling, navigation and path planning, reconnaissance, and map building, to name a few. Within specific frameworks, each vehicle determines their own mission by simultaneously choosing tasks for all vehicles in the fleet. It is typically assumed that each vehicle then executes its own plan. To ensure consistency, information is shared to update the situational awareness and to negotiate on the designed plans [51].

Shao et al. [52] address a cooperative control of multiple robotic fish in order to achieve a disk-pushing task since the capacity of a single fish robot is often limited. In [53], Jin et al. consider a heterogeneous team of UAVs drawn from several distinct classes and engaged in a search and destroy mission over an extended battlefield. Each class of UAVs has its own sensing and attack capabilities with respect to the different target types, so the need for appropriate and efficient assignment is paramount. They present a simple cooperative approach to this problem based on distributed assignment mediated though centralized mission status information.

In [54], Oh et al. present a coordinated road network search algorithm for multiple heterogeneous UAVs. They also consider physical constraints of UAVs into the search problem. Alighanbari et al. [51] investigate the problem of decentralized task assignment for a fleet of UAVs since centralized task assignment for a fleet of UAVs is often not practical due to communication limits, robustness issues, and scalability. They assume that each UAV has the same situational awareness. Nett et al. [55] present an architecture that allows autonomous mobile systems to schedule shared resources in real-time using their own wireless distributed infrastructure. In this architecture, there is a clear separation between the application-specific scheduling part and the application independent communication part that constitutes the real-time and reliable hard-core of the system.

Schwager et al. in [56] present a distributed control algorithm to drive a group of robots to be spread out over an environment and provide adaptive sensor coverage of the environment. Burgard et al. [20] present a probabilistic approach for the coordination of multiple robots considering the problem of exploring an unknown environment by a team of robots. Willmann et al. in [57] propose a novel field of architectural research - aerial robotic construction (ARC) - where aerial robotics is used not only for construction, but also as a guiding principle in the design and fabrication process. In [58], Maza et al. present an architecture to perform cooperative missions with multiple UAVs with sensing and actuation capabilities. The interactions between UAVs are not only information exchanges but also physical couplings required to cooperate in the joint transportation of a single load.

3.2 Fault-Toletant Cooperative Control

Achieving cooperative control, however, is quite challenging. Many issues must be addressed in order to develop a working cooperative team, such as action selection, coherence, conflict resolution, and communication. Furthermore, these cooperative teams often work in dynamic and unpredictable environments, requiring the group members to respond robustly, reliably, and adaptively to unexpected environmental changes, failures in the inter-agent communication system, and modifications in the cooperative team that may occur due to mechanical failure, or the addition or removal of robots from the team by human intervention [4].

As described in the previous section, while much research in recent years has addressed the issues of multi-agent cooperation, current cooperative technology is still far from achieving many of these real world applications. One reason for this technology gap may be that previous work has not adequately addressed the issues of fault tolerance. Here, by fault tolerance, it means the ability of the cooperative group to respond to individual robot failures or failures in communication that may occur at any time during the mission. Agent faults occur inside an agent and change its dynamics. Communication faults affect the communication performance among agents.

3.2.1 Component Fault

In [59], Marino et al. investigate the team robustness to faults of individual agent while carrying out multi-robot border patrolling in the framework of the Null-Space-based Behavioral control. Chang et al. [60] present an adaptive dynamic surface controller for multi-robot systems where the kinematic model of wheeled robots is addressed with the consideration of actuator faults. The proposed controller overcomes partial loss of actuator effectiveness with graph theory used to describe the communication topology between agents. Samsar et al. in [61] address results on performance analysis of a team of agents in the presence of team members faults for three types of faults, i.e. loss of effectiveness, float, and lock-in-place. The team goal is to accomplish a cohesive motion in a modified leader-follower architecture using a semi-decentralized optimal control.

Xu et al. in [62] study fault-tolerant control method of incipient and severe actuator faults of UAVs formation. In the event of actuator faults, a compensation is added into the initial controller to achieve fault tolerance and stabilize the whole formation as well. Zhang et al. in [63] investigate the cooperative interception of a moving target by multiple vehicles with tolerance of actuator or network failures, where novel fault-tolerant consensus protocols are proposed to address actuator failures and network failures, respectively. Mead et al. [64] present a cellular automata-based robot control architecture with respect to necessary characteristics to handle real-world occurrences, i.e. formation repair, obstacle avoidance, and changes in the formation. In [65], Chamseddine et al. propose a trajectory planning/re-planning approach which can be effectively employed to minimize the effect of potential faults on vehicles in formation. The fault tolerance is achieved by changing the reference trajectories of the entire formation so that to allow the damaged UAV to follow the healthy ones under the virtual structure formation framework.

3.2.2 Topology Fault

In [66], Abdessameud et al. address the formation control problem of a group of VTOL aircraft with delayed communication. The communication between aircrafts is assumed to be undirected and fixed. Three formation control schemes with input constraints, under delay-dependent and delay-independent conditions, have been proposed. Yang et al. in [67] study the fault-tolerant formation keeping problem of multi-agent systems with consideration of shortest connection topology. Given a formation shape, a connection topology design method and its reconfiguration strategy are proposed in the sense that the whole connection path is shortest despite of communication faults.

Franco et al. in [68] address the problem of cooperative control of a team of distributed agents with decoupled nonlinear discrete-time dynamics, which operate in a common environment and exchange-delayed information between them. In [69], Izadi et al. investigate the decentralized model predictive control of multiple cooperative vehicles with the possibility of communication loss/delay. In the event of a communication loss (packet dropout), the most recent available information, which is potentially delayed, is used. Since large communication delays can lead to poor cooperation performance and unsafe behaviors such as collisions, they develop a new decentralized MPC approach to improve the cooperation performance and achieve safety in the presence of the large communication delays.

4 Conclusion and Future Directions

As computer processors and wireless communication system become smaller and cheaper, UAVs will continue to be desirable in new applications and in replacement of manned aircraft, with increasing requirements for autonomy and reliability. They operate in uncertain environments, and are often exposed to large disturbances, so efficiency and reliability of such systems are highly affected by the selected control strategy.

In general, cooperative network design problem can be formulated into three main subproblems. 1) Control design: especially issues related to stabilization, controllability, and robustness linked with each individual agent. 2) Topology design: especially issues related to information exchanges and resource allocation. 3) Path planning: especially issues related to path planning for each individual agent and path reconfiguration for the team.

Although there has been substantial work in cooperative control over the past decade, there are still many open problems that remained to be solved. Most of the results that have appeared in the literature deal with a point-mass model of multi-agent systems, and there is an urgent need to design a distributed collaborative team consisting of agents with more complex dynamics. Dynamic network topologies with time-varying structures and changing number of nodes in the network and communication issues in the presence of uncertainties and partial failures should indeed be considered. Robustness issues, particularly in the cases of presence of faults and anomalies in the team members, and considerations for various network structures need to be investigated.

References

[1] L. E. Parker, "Current state of the art in distributed autonomous mobile robotics," in *Distributed Autonomous Robotic Systems 4*, pp. 3–12, Springer, 2000.

[2] R. M. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 571–583, 2007.

[3] E. Semsar-Kazerooni and K. Khorasani, *Team Cooperation in a Network of Multi-vehicle Unmanned Systems: Synthesis of Consensus Algorithms*. Springer New York, 2013.

[4] L. E. Parker, "Alliance: An architecture for fault tolerant, cooperative control of heterogeneous mobile robots," in *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems, Advanced Robotic Systems and the Real World*, vol. 2, pp. 776–783, 1994.

[5] R. Isermann, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer Berlin Heidelberg, 2006.

[6] M. M. Mahmoud, J. Jiang, and Y. M. Zhang, *Active Fault Tolerant Control Systems: Stochastic Analysis and Synthesis*. Springer Berlin Heidelberg, 2003.

[7] H. Alwi, C. Edwards, and C. P. Tan, *Fault Detection and Fault-Tolerant Control Using Sliding Modes*. Springer London, 2011.

[8] C. Edwards, T. Lombaerts, and H. Smaili, *Fault Tolerant Flight Control – A Benchmark Challenge*. Lecture Notes in Control and Information Sciences, Springer Berlin Heidelberg, 2010.

[9] J. Jiang and Q. Zhao, "Design of reliable control systems possessing actuator redundancies," *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 4, pp. 709–718, 2000.

[10] F. Liao, J. L. Wang, and G. H. Yang, "Reliable robust flight tracking control: an LMI approach," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 1, pp. 76–89, 2002.

[11] Y. M. Zhang and J. Jiang, "Integrated active fault-tolerant control using IMM approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 4, pp. 1221–1235, 2001.

[12] Y. M. Zhang and J. Jiang, "Active fault-tolerant control system against partial actuator failures," *IEE Proceedings-Control Theory and Applications*, vol. 149, no. 1, pp. 95–104, 2002.

[13] Y. M. Zhang and J. Jiang, “Bibliographical review on reconfigurable fault-tolerant control systems,” *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252, 2008.

[14] J. S. Eterno, J. L. Weiss, D. P. Looze, and A. Willsky, “Design issues for fault tolerant-restructurable aircraft control,” in *Proceedings of the 24th IEEE Conference on Decision and Control*, pp. 900–905, 1985.

[15] J. Gundlach, *Designing Unmanned Aircraft Systems: A Comprehensive Approach*. American Institute of Aeronautics and Astronautics, 2012.

[16] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, “Cooperative forest fire surveillance using a team of small unmanned air vehicles,” *International Journal of Systems Science*, vol. 37, no. 6, pp. 351–360, 2006.

[17] N. Nigam, S. Bieniawski, I. Kroo, and J. Vian, “Control of multiple UAVs for persistent surveillance: algorithm and flight test results,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 5, pp. 1236–1251, 2012.

[18] J. Clapper, J. Young, J. Cartwright, and J. Grimes, “Unmanned systems roadmap 2007-2032,” tech. rep., 2007.

[19] G. Vachtsevanos, B. Ludington, J. Reimann, P. Antsaklis, and K. Valavanis, “Modeling and control of unmanned aerial vehicles–current status and future directions,” *Modeling and Control of Complex Systems*, vol. 30, no. 1, pp. 30–33, 2008.

[20] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, “Collaborative multi-robot exploration,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 476–481, 2000.

[21] J. T. Feddema and D. Schoenwald, “Decentralized control of cooperative robotic vehicles,” in *Proc. SPIE 4364, Unmanned Ground Vehicle Technology III*, pp. 136–146, 2001.

[22] J. S. Jennings, G. Whelan, and W. F. Evans, “Cooperative search and rescue with a team of mobile robots,” in *Proceedings of the 8th International Conference on Advanced Robotics*, pp. 193–200, 1997.

[23] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer, and C. J. Taylor, “A vision-based formation control framework,” *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 813–825, 2002.

[24] D. Rus, B. Donald, and J. Jennings, “Moving furniture with teams of autonomous robots,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 235–242, 1995.

[25] R. Murphey and P. M. Pardalos, *Cooperative Control and Optimization*. Springer US, 2002.

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

[27] L. A. Weitz, J. E. Hurtado, and A. J. Sinclair, “Decentralized cooperative-control design for multivehicle formations,” *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 4, pp. 970–979, 2008.

[28] Y. M. Zhang and H. Mehrjerdi, “A survey on multiple unmanned vehicles formation control and coordination: normal and fault situations,” in *Proceedings of the 2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1087–1096, 2013.

[29] D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, “A survey of spacecraft formation flying guidance and control. part ii: control,” in *Proceedings of the 2004 American Control Conference.*, vol. 4, pp. 2976–2985, 2004.

[30] P. Wang and F. Y. Hadaegh, “Coordination and control of multiple microspacecraft moving in formation,” *Journal of the Astronautical Sciences*, vol. 44, no. 3, pp. 315–355, 1996.

[31] B. Seanor, Y. Gu, M. Napolitano, G. Campa, S. Gururajan, and L. Rowe, “3-aircraft formation flight experiment,” in *Proceedings of the 14th Mediterranean Conference on Control and Automation*, pp. 1–6, 2006.

[32] V. Roldao, R. Cunha, D. Cabecinhas, C. Silvestre, and P. Oliveira, “A leader-following trajectory generator with application to quadrotor formation flight,” *Robotics and Autonomous Systems*, vol. 62, no. 10, pp. 1597–1609, 2014.

[33] F. Giulietti, L. Pollini, and M. Innocenti, “Autonomous formation flight,” *IEEE Control Systems*, vol. 20, no. 6, pp. 34–44, 2000.

[34] Z. Weihua and T. H. Go, “Robust cooperative leader-follower formation flight control,” in *Proceedings of the 11th International Conference on Control Automation Robotics & Vision (ICARCV)*, pp. 275–280, 2010.

[35] E. Lalish, K. A. Morgansen, and T. Tsukamaki, “Formation tracking control using virtual structures and deconfliction,” in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 5699–5705, 2006.

[36] M. A. Lewis and K.-H. Tan, “High precision formation control of mobile robots using virtual structures,” *Autonomous Robots*, vol. 4, no. 4, pp. 387–403, 1997.

[37] W. Ren and R. W. Beard, “Virtual structure based spacecraft formation control with formation feedback,” in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 2002–4963, 2002.

[38] C. W. Reynolds, “Flocks, herds and schools: A distributed behavioral model,” *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[39] O. Saif, I. Fantoni, and A. Zavala-Rio, “Flocking of multiple unmanned aerial vehicles by LQR control,” in *Proceedings of the 2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 222–228, 2014.

[40] T. Balch and R. C. Arkin, “Behavior-based formation control for multirobot teams,” *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998.

[41] S. Monteiro and E. Bicho, “A dynamical systems approach to behavior-based formation control,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2606–2611, 2002.

[42] P. Vadakkepat, O. C. Miin, X. Peng, and T. H. Lee, “Fuzzy behavior-based control of mobile robots,” *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 559–565, 2004.

[43] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[44] J. A. Fax and R. M. Murray, “Information flow and cooperative control of vehicle formations,” *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[45] Y. Qu, X. Zhu, and Y. M. Zhang, “Cooperative control for UAV formation flight based on decentralized consensus algorithm,” in *Intelligent Robotics and Applications*, pp. 357–366, Springer, 2012.

[46] X. Chen and C. Zhang, “The method of multi unmanned aerial vehicle cooperative tracking in formation based on the theory of consensus,” in *Proceedings of the 5th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 2, pp. 148–151, 2013.

[47] T. Paul, T. R. Krogstad, and J. T. Gravdahl, “UAV formation flight using 3D potential field,” in *Proceedings of the 16th Mediterranean Conference on Control and Automation*, pp. 1240–1245, 2008.

[48] H. Rezaee and F. Abdollahi, "Mobile robots cooperative control and obstacle avoidance using potential field," in *Proceedings of the 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 61–66, 2011.

[49] Y. Eun and H. Bang, "Cooperative control of multiple unmanned aerial vehicles using the potential field theory," *Journal of Aircraft*, vol. 43, no. 6, pp. 1805–1814, 2006.

[50] Y. Nagao and K. Uchiyama, "Formation flight of fixed-wing UAVs using artificial potential field," in *Proceedings of the 29th Congress of the International Council of the Aeronautical Sciences*, 2014.

[51] M. Alighanbari and J. P. How, "Decentralized task assignment for unmanned aerial vehicles," in *Proceedings of the 44th IEEE Conference on Decision and Control and the 2005 European Control Conference*, pp. 5668–5673, 2005.

[52] J. Shao, L. Wang, and J. Yu, "Cooperative control of multiple robotic fish in a disk-pushing task," in *Proceedings of the 2006 American Control Conference*, pp. 2730–2735, 2006.

[53] Y. Jin, A. A. Minai, and M. M. Polycarpou, "Cooperative real-time search and task allocation in uav teams," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 7–12, 2003.

[54] H. Oh, H. Shin, A. Tsourdos, B. White, and P. Silson, "Coordinated road network search for multiple uavs using dubins path," in *Advances in Aerospace Guidance, Navigation and Control*, pp. 55–65, Springer, 2011.

[55] E. Nett and S. Schemmer, "Reliable real-time communication in cooperative mobile applications," *IEEE Transactions on Computers*, vol. 52, no. 2, pp. 166–180, 2003.

[56] M. Schwager, M. P. Vitus, D. Rus, and C. J. Tomlin, "Robust adaptive coverage for robotic sensor networks," in *Proceedings of the International Symposium on Robotics Research (ISRR 11)*, 2011.

[57] J. Willmann, F. Augugliaro, T. Cadalbert, R. D'Andrea, F. Gramazio, and M. Kohler, "Aerial robotic construction towards a new field of architectural research," *International Journal of Architectural Computing*, vol. 10, no. 3, pp. 439–460, 2012.

[58] I. Maza, K. Kondak, M. Bernard, and A. Ollero, "Multi-UAV cooperation and control for load transportation and deployment," *Journal of Intelligent and Robotic Systems*, vol. 57, no. 1-4, pp. 417–449, 2010.

[59] A. Marino, L. E. Parker, G. Antonelli, F. Caccavale, and S. Chiaverini, "A fault-tolerant modular control approach to multi-robot perimeter patrol," in *Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 735–740, 2009.

[60] Y.-H. Chang, W.-S. Chan, C.-Y. Yang, C.-W. Tao, and S.-F. Su, "Adaptive dynamic surface control for fault-tolerant multi-robot systems," in *Proceedings of the 2013 International Conference on System Science and Engineering (ICSSE)*, pp. 25–30, 2013.

[61] E. Semsar-Kazerooni and K. Khorasani, "Analysis of actuator faults in a cooperative team consensus of unmanned systems," in *Proceedings of the 2009 American Control Conference*, pp. 2618–2623, 2009.

[62] Q. Xu, H. Yang, B. Jiang, D. Zhou, and Y. M. Zhang, "Adaptive fault-tolerant control design for UAVs formation flight under actuator faults," in *Proceedings of the 2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1097–1105, 2013.

[63] P. Zhang, H. H. Liu, X. Li, and Y. Yao, "Fault tolerance of cooperative interception using multiple flight vehicles," *Journal of the Franklin Institute*, vol. 350, no. 9, pp. 2373–2395, 2013.

[64] R. Mead, R. Long, and J. B. Weinberg, "Fault-tolerant formations of mobile robots," in *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4805–4810, 2009.

[65] A. Chamseddine, Y. M. Zhang, and C. Rabbath, "Trajectory planning and re-planning for fault tolerant formation flight control of quadrotor unmanned aerial vehicles," in *Proceedings of the 2012 American Control Conference (ACC)*, pp. 3291–3296, 2012.

[66] A. Abdessameud and A. Tayebi, "Formation control of VTOL unmanned aerial vehicles with communication delays," *Automatica*, vol. 47, no. 11, pp. 2383–2394, 2011.

[67] H. Yang, B. Jiang, and Y. M. Zhang, "Fault-tolerant shortest connection topology design for formation control," *International Journal of Control, Automation and Systems*, vol. 12, no. 1, pp. 29–36, 2014.

[68] E. Franco, L. Magni, T. Parisini, M. M. Polycarpou, and D. M. Raimondo, "Cooperative constrained control of distributed agents with nonlinear dynamics and delayed information exchange: A stabilizing receding-horizon approach," *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 324–338, 2008.

[69] H. A. Izadi, B. W. Gordon, and Y. M. Zhang, "Decentralized model predictive control for cooperative multiple vehicles subject to communication loss," *International Journal of Aerospace Engineering*, vol. 2011, 2011.