



Article DDS-Based Containment Control of Multiple UAV Systems

Basem AL-Madani¹, Siddig M. Elkhider^{2,*} and Sami El-Ferik²

- ¹ Computer Engineering Department, KFUPM, P. O. 5067, Dhahran 31261, Saudi Arabia; mbasem@kfupm.edu.sa
- ² Systems Engineering Department, KFUPM, P. O. 5067, Dhahran 31261, Saudi Arabia; selferik@kfupm.edu.sa
- * Correspondence: siddig.elnaiem@kfupm.edu.sa

Received: 5 June 2020; Accepted: 29 June 2020; Published: 1 July 2020



Abstract: In this paper, we present a robust containment control design for multi Unmanned Aerial Vehicle Systems (UAVs) based on the Data Distribution Service (DDS) middleware and \mathcal{L}_1 adaptive controller. The Data Distribution Service middleware, \mathcal{L}_1 adaptive controller and graph theory technique are utilized for the navigation of the UAVs. The \mathcal{L}_1 controller is utilized as a local controller for each UAVs and the graph theory approach is utilized to constitute the followers inside their leaders. Finally, the DDS Middleware is used to exchange data between the followers and their leaders. Robust adaptation of the \mathcal{L}_1 controller makes the system robust with a high level of performance. Matlab simulation verified the robustness of the \mathcal{L}_1 controller. We provide stability proofs using Lyapunov analysis for the UAVs framework.

Keywords: containment control; leader-follower control; unmanned aerial vehicle system; data distribution service middleware; quality of service

1. Introduction

1.1. Motivation

In recent years middleware systems have been used in a wide variety of applications Essential applications of middleware systems have been found in real-time systems. One of these systems is the multi unmanned aerial vehicle system which has more advantages than a single UAV system due to its ability to perform complex tasks more effectively than a single UAV. Several development approaches have been cultivated based on the appropriate application to manage a group of agent systems. Multi-agent systems can work cooperatively in many of these applications to integrate specific activities or actions. Some researchers have utilized inflexible or unreliable middleware to share data between multiple vehicles, such as the Automotive Open Systems Architecture (AUTOSAR) and the Open Services Gateway Initiative (OSGi) that do not accept Quality of Service (QoS). In most of these researches the middleware has not been used with the formation or containment control of multiple vehicles. Other authors have utilized classical containment control for multi-agent systems without considering the issue of uncertainty in the parameters or they have considered a limit on the number of the agents, for example, by using classical state feedback, an output feedback controller, and a classical feedback linearization controller. In a similar way formation maintenance has been addressed by using a potential field approach and feedback linearization controller (see [1,2]). Some advanced controllers have been applied to a single agent like the \mathcal{L}_1 adaptive control for underwater vehicle-manipulator systems (UVMs) (see [3]), while, in [4–6] an integral SMC and adaptive SMC have been proposed and implemented to the quadrotor trajectory tracking control. This work can also be linked to the

data-driven algorithms in [7]. Therefore, using the Data Distribution Service (DDS) middleware system has many advantages like increased flexibility, reliability, and portability.

1.2. Related Research

Many researchers have worked on multi-vehicle middleware and containment control. Lee et al. [8] introduced a leader-follower formation control methodology for multiple UAVs systems. A backstepping controller was used to stabilize multi-UAV formation control. In addition, the state transformation approach was used for formation among the UAVs. Furthermore, the followers' network topology depended upon the location of their leader. Finally, they examined formation control under certain uncertainties.

De Coninck et al. [9] presented an OSGi-based middleware that enables a distributed platform of sensors, robots, and actuators to cooperate. The proposed middleware can support a platform with a new technique, such as artificial neural networks, that have become very common for robotic control and data analytics of robotic sensors. Robot Operating System (ROS) nodes have storage restrictions and a limited CPU which affects the performance of robots while performing tasks. Thus, one solution is to send the computing tasks to the Cloud.

El-Ferik et al. [10] suggested a multiple UVMs adaptive control where the role of the potential field is used for containment among the UVMs and to share information between them. Furthermore, the followers network topology depends upon the location of their leader and the repulsive feature is utilized to prevent collisions between the UVMs. Finally, the \mathcal{L}_1 controller was used to stabilize the multi-UVMs containment control.

Zhang et al. [11] developed a data-driven distributed adaptive cooperative control for multi-agent systems which involves a multi-direction queuing strength balance. For the distributed consensus between the multi-direction queuing strength, the graph theory approach was used. Then, they studied all directions of the queuing delays of a multi-direction queuing strength balance and they compared it to an unconstrained fixed sequence clearing strategy to prove the high level of efficiency of the proposed controller.

Kouba et al. [12] introduced a framework for a cooperative multi agent system based on the Robot Operating System (ROS) middleware. For multi agent systems, they presented a high-level architecture that facilitates the development and design. In addition, the proposed architecture ensures the scalability, software reuse, modularity and extensibility of the multi-agent running software. Furthermore, as ROS was designed for a single robot. This new framework introduces a perceptible solution to the ROS middleware for multi-robot software development.

Luo et al. [13] developed a middleware based on a web service technology to schedule Robot Operating System nodes in a distributed network and analyze the task requests of cloud robotics, as well as automatically schedule the ROS packages as Web services.

Vidal et al. [14] proposed a data-centric method for multiple heterogeneous and interconnected UAVs. The DDS middleware can exchange and make use of information for multiple heterogeneous UAVs. Exchanging the information between these UAVs is supported by a publisher and subscriber technique with a flexible set of quality of service rules that need to be fulfilled before the interaction between the publisher and subscriber. In addition, the data-centric method eases the integration between different UAV dynamics (heterogeneous UAVs). Furthermore, the DDS middleware enhances the adaptability of the Unmanned Aerial Vehicles for several task objectives.

Xie et al. [15] proposed a framework containing a couple of robotic clouds to exchange messages and organize multiple resources based on web service composition. In addition, the proposed architecture checks for Quality of Service before dealing with the requested task. Moreover, they introduced the concept of a cloud priority strategy and user sensitivity. Finally, a GICA-CP algorithm was used for the resource deployment approach that made the proposed framework capable of effectively allocating resources in robotic cloud workflows. Xiong et al. [16] introduced a multi–single integrator discrete time system containment control utilizing the directed graph theory methodology. They fixed the followers network topology based on the position of the leader. Furthermore, they investigated the containment control for a multi–single integrator dynamics system using a nonlinear projection algorithm. Finally, the containment control of multi–single integrator discrete-time systems under bounded network delays and switching topologies was examined. In a similar study a containment control for multi double integrator dynamics systems was introduced by Chunhua Yang et al. [17]. The graph theory method for the multi double integrator dynamics systems consensus was utilized. Furthermore, they examined a multi double integrator dynamics system containment control using a nonlinear projection algorithm coupled with non-convex control input constraints.

Junyi Yang et al. [18] developed a multi double integrator dynamics systems containment control. Based on the position leader, they fixed the network topology of the followers. In addition, they used the graph theory method for the multi double integrator dynamics systems consensus combined with edge-event-triggered control. In order to stabilize the containment control of multi double integrator dynamics systems they utilized a classical feedback linearisation controller.

Yongliang Yang et al. [19] introduced a leader-follower containment control methodology for multi general linear dynamics systems. They studied the multi general linear dynamic systems with a directed network topology and active leaders. In addition, they utilized the graph theory for the directed network topology. Furthermore, they studied the containment control of multi general linear dynamic systems using off-policy reinforcement learning (RL). Finally, an algebraic Riccati equation was solved to calculate the containment gain matrices.

Wang et al. [20] developed an adaptive containment control for multi Euler–Lagrange systems. For the network topology between the multi Euler–Lagrange systems, the graph theory approach was used. In addition, the followers network topology depended on at least one leader. Then, they studied the containment control of multi Euler–Lagrange systems under unknown nonlinear dynamics, in which fuzzy logic control was used as an approximation technique. Finally, they utilised an adaptive control approach with fuzzy logic control (FLC) to stabilize the containment of multi Euler–Lagrange systems.

Zheng et al. [21] proposed a containment control for multi high-order discrete-time systems. They used a fixed network topology between the leaders and an undirected network topology between the followers. For the multi high-order discrete-time systems consensus, they utilized the graph theory approach. Furthermore, they studied the containment control of a multi high-order discrete-time system using stochastic indecomposable and aperiodic (SIA) matrices. Finally, they studied a multi high-order discrete-time systems containment control under switching topologies.

El-Ferik et al. [22] proposed a data-centric method for formation control of multiple UAVs. Exchanging the information between these UAVs is supported by a publisher and subscriber technique with a flexible set of quality of service rules that need to be fulfilled before the interaction between the publisher and subscriber. In addition, the \mathcal{L}_1 controller was used to stabilize the multi-UAVs formation control. Finally, the repulsive feature is utilized to prevent collisions between the UAVs.

Wang and Tong [23] proposed adaptive containment control of multi general nonlinear systems. They utilized the graph theory approach for the network topology between the multi general nonlinear systems. In addition, the network topology of the followers depended on at least one leader. Furthermore, they studied the containment control of multi general nonlinear systems under some immeasurable states, in which observer-based fuzzy logic control is used. Moreover, they considered some uncertainties in the general nonlinear model. Finally, they utilized the adaptive fuzzy logic control (FLC) to stabilizing the containment of multi general nonlinear systems.

Li et al. [24] proposed a containment control of multi general linear systems with time-varying. They studied the multi general linear systems with directed network topology. Besides, they utilized the graph theory for the directed network topology. Furthermore, they studied the containment control of multi general linear systems under bounded communication network delays. Moreover, they utilized linear matrix inequalities (LMIs) controller to stabilizing the containment control of multi general linear systems.

Xu, et al. [25] proposed a multi second-order systems formation control with time-varying. They studied the formation of second-order systems under bounded disturbances. In addition, the fixed network topology of the followers depended on the leader position. Furthermore, they used the graph theory method for the multi second-order systems consensus. Moreover, they utilized a classical proportional-integral controller to stabilize the formation of multi-second order systems.

Guzey et al. [26] proposed a robust leader-follower control of multi under-actuated UAVs. They utilised the back-stepping controller to stabilizing the UAVs nonlinear model. They handled the UAVs under-actuation using a neural network. Moreover, they used the size reduction matrix for the formation between the UAVs.

Zuo et al. [27] proposed adaptive leader-follower containment control of multi heterogeneous general linear dynamical systems with unknown leader's. The fixed network topology of the followers depended on the leader's positions. Besides, they used the graph theory method for the multi-heterogeneous general linear dynamics systems consensus. Moreover, they utilized the classical state feedback controller and dynamic output feedback controller to stabilizing the containment control of multi heterogeneous general linear dynamics systems.

Qian et al. [28] proposed a super-twisting sliding mode control scheme for multi-robot formation control based on the leader–follower approach. The stability of the closed-loop formation system was investigated using Lyapunov. The presented system showed a good performance against uncertainties and external disturbances.

Wei et al. [29] investigated the consensus for second-order multiple robot systems considering both noise and time delay. Based on the frequency domain analysis a consensus algorithm was proposed where the system's characteristic equation was mapped to a quadratic polynomial with purely imaginary eigenvalues. The algorithm had low conservativeness and it could be extended to multi-robot systems with higher-order dynamics.

Xia et al. [30] studied the problem of formation control for underactuated surface vessels subjected to modeling uncertainties and external disturbances. Furthermore, the input saturation, collision avoidance, and limited communication scale was considered. For each vessel, a distributed control law was formulated.

Nguyen et al. [31] proposed a distributed dynamic event-triggered control algorithm for the class of second-order agents. These gents have limited communication and can only exchange information locally. The algorithm allows agents to track a time-varying trajectory with a stable performance.

In this article, we are proposing multi-UAVs containment control. In an uncertain area some UAVs lead the others. Every UAV throughout the group has to coordinate and establish the required forms concerning their leaders and each other. The distributed method suggests utilizing an \mathcal{L}_1 adaptive to stabilize all of the UAVs and using the DDS middleware to share the information among all the UAVs. Last, the graph theory approach is used for the UAV's navigation. The contributions of this paper are:

- A robust architecture for containment control of multi UAVs is designed.
- We suggest sharing the data among the UAVs through the DDS middleware.

The remainder of this paper is arranged as follows: we start with some preliminaries of the DDS middleware, the graph theorem approach, and the quadrotor dynamics model in Section 2. In Section 3, we prove the stability of multi UAVs by using the \mathcal{L}_1 technique with Lyapunov analysis. Finally, our simulations and concluding remarks are presented at the end.

2. Preliminaries

In this section, we introduce some preliminaries for the DDS middleware, graph theory technique and quadrotor dynamics.

2.1. Data Distribution Service (DDS)

DDS middleware is widely considered to be the best middleware for containment control of UAVs owing to the fact that it is able to manage information due to real-time subscription publishing and mission-critical applications. It also consists of a flexible set policy for Quality of Service (QoS) that can be employed for different purposes. Since DDS provides almost whole performance metrics, DDS middleware is considered in order to conduct our study for the containment control of UAVs. The DDS middleware is structured on architecture for a publishing subscription by assuming certain UAVs as subscribers and other as publishers to collaborate with each other. A flexible list of Quality of Service Policy requirements that need to be met prior to cooperation between the publisher and the subscriber includes:

- **Durability**: Specifies whether the publisher previously transmitted data to a new subscriber or not. This QoS strategy helps separate the system from reliance on start-ups.
- **Reliability**: Determines whether or not the DDS will re-send data missing across the middleware. Reliability includes two configurations: BEST EFFORT (do not resend data missing) or RELIABLE (resend data missing).
- **History**: Specifies that the data received or sent by a publisher will be retained for a subscriber. There are dual settings: KEEP ALL or KEEP LAST. KEEP ALL does not indicate that the middleware is storing unlimited data.
- **Deadline**: For subscribers: determines the maximum time for the arriving samples. On the publisher side: establishes a pledge to publish data not exceeding this time.

2.2. Algebraic Graph Theory

A directed graph (digraph) can be used to model the communication network of a multi-agent system. A digraph is generally defined as a non-empty finite group of N nodes, a group of the $A_S = [a_{ij}] \in \mathbb{R}^{N \times N}$ associated adjacency matrix and arcs or edges. In a quadrotor, agents are the nodes of the digraph of the communication. The edges of the communication network's corresponding digraph denote the communication links. The digraph is considered to be invariant in time in this article, i.e. A_S is constant. (x_j, x_i) denotes an edge from j node to i node, meaning that i node receives the data from j node, a_{ij} is the weight of (x_j, x_i) . Node i is considered a j node neighbor. The j node neighbor set is referred to as N_j . For a digraph, if i node is a j node neighbor, then j node can get information from i node, but not necessarily vice versa. The matrix of in-degree is set to $D = diag(d_i \in \mathbb{R}^{N \times N}$. Defined as $L_G = D - A_S$ is the Laplacian matrix. A direct path from the i node to the j node is an edge series. If there is a root node with a straightforward path from that node to each other node in the graph, a digraph is said to have a spanning tree.

2.3. Quadrotor's Dynamics Model

The quadrotor's translational dynamics model based on Euler-Lagrangian are given by [22]

$$\ddot{\eta}_1 = -g \begin{bmatrix} 0\\0\\1 \end{bmatrix} + J_1(\eta_2) \begin{bmatrix} 0\\0\\u/m \end{bmatrix} - \frac{k_t}{m} \dot{\eta}_1$$
(1)

where η_1 represents x, y and z position, η_2 represents $yaw(\psi)$, $pitch(\theta)$, $roll(\phi)$ motion, the mass and gravity acceleration represented by m and g respectively, k_t represents the drag coefficient , and $J_1(\eta_2)$ is the transformation matrix linked to the Euler angles: $yaw(\psi)$, $pitch(\theta)$, $roll(\phi)$, as follows

$$J_{1}(\eta_{2}) = \begin{bmatrix} c\theta c\psi & -s\psi c\phi + c\psi s\phi s\theta & s\phi s\psi + c\phi c\psi s\theta \\ c\theta s\psi & c\phi c\psi + s\psi s\phi s\theta & -s\phi c\psi + c\phi s\psi s\theta \\ -s\theta & s\phi c\theta & c\theta c\phi \end{bmatrix}$$
(2)

Assuming $\phi \neq 90^{\circ}$, $\theta \neq 90^{\circ}$, and the property of $J_1(\eta_2)$ is

$$J_1^{-1}(\eta_2) = J_1^T(\eta_2) \tag{3}$$

where $s(.) = \sin c(.) = \cos t(.) = \tan t$.

The thrust force u is as follows

$$u = f_1 + f_2 + f_3 + f_4 \tag{4}$$

where the upward lifting forces equal to $f_i = k_i \Omega_i^2$, where the positive constants denoted by k_i , and Ω_i represent the motors' angular speeds.

The quadrotor's rotational equation is as follows

$$\dot{\nu}_2 = I^{-1}(-(\nu_2 \times I\nu_2) - I_R(\nu_2 \times z_e)\Omega - k_r\nu_2 + \tau)$$
(5)

where v_2 represents the angular velocity vector, $I = \text{diag}(I_x, I_y, I_z)$ represents the quadrotor's inertia, $z_e^T = [0, 0, 1]$, propeller inertia denoted by I_R , cross product denoted by \times , rotational drag denoted by k_r , and

$$\Omega = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4 \tag{6}$$

The torques $\tau = [\tau_p, \tau_q, \tau_r]^\top$ and force u on the body frame. The quadrotor's translational and rotational motion is as follows

$$\begin{bmatrix} \tau \\ u \end{bmatrix} = \begin{bmatrix} \tau_p \\ \tau_q \\ \tau_r \\ u \end{bmatrix} = \begin{bmatrix} 0 & l & 0 & -l \\ l & 0 & -l & 0 \\ d & -d & d & -d \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$$
(7)

where the drag factor is represented by *d* and the distance between the motors and center is represented by *l*.

The system used in the above equations is an under-actuated system, because the number of the degrees of freedom is higher than the actuation. The under-actuated systems considered in this paper are as follows

$$\ddot{q} = f(q, \dot{q}) + G(q)u \tag{8}$$

where the coordinates vector is represented by q, the dynamic vector represented by f, the input matrix represented by G, and u is the control inputs. Details of quadrotor dynamic are available in [22,32].

3. Control Design of Multi-UAVs

3.1. \mathcal{L}_1 Adaptive Control

The main feature of utilizing the \mathcal{L}_1 adaptive controller is its ability to adapt rapidly and robustly, in addition to decoupling between robustness and adaptation. In addition, it is a guaranteed time-delay margin and it reduces the limitations on hardware performance. The main challenge of utilizing the \mathcal{L}_1 adaptive controller is the optimal design of the bandwidth-limited filter [33]. The \mathcal{L}_1 controller's components are illustrated in Figure 1.



Figure 1. \mathcal{L}_1 Adaptive control.

In order to apply the \mathcal{L}_1 controller on the rotational dynamics in Equation (5) the state space representation needed is as follows:

$$\dot{x} = A_m x + b(\omega u_{ad} + f(t, x(t))), \ x(0) = x_0$$

 $y = c^{\top} x(t)$ (9)

where

$$x \triangleq \nu_{2}, \qquad A_{m} \in \mathbb{R}^{n \times n} \triangleq \text{ a known Hurwitz matrix}$$

$$b = 1, \quad \omega \triangleq I_{M}^{-1}, \quad u_{ad} \triangleq \tau \triangleq \mathcal{L}_{1} \text{ Adaptive control}$$

$$f(t, x(t)) \triangleq I_{M}^{-1}(-(\nu_{2} \times I_{M}\nu_{2}) - I_{R}(\nu_{2} \times z_{e})\Omega - k_{r}\nu_{2}),$$

$$c^{\top} = I_{3 \times 3} \qquad (10)$$

We can rewrite Equation (9) as follows:

$$\dot{x} = A_m x + b(\omega u_{ad} + \theta \| x \|_{\infty} + \sigma), \quad x(0) = x_0$$

$$y = c^\top x(t) \tag{11}$$

The predictor's state of Equation (11) is

$$\dot{\hat{x}} = A_m \hat{x} + b(\omega u_{ad} + \hat{\theta} \| x \|_{\infty} + \hat{\sigma}), \quad \hat{x}(0) = x_0$$
$$\hat{y} = c^{\top} \hat{x}$$
(12)

where the estimated state denoted by $\hat{x} \in \mathbb{R}^n$, $\hat{y} \in \mathbb{R}^n$ denotes the estimated output, the estimated parameters denoted by $\hat{\theta}$ and $\hat{\sigma}$. The error defined as $\tilde{x} = \hat{x} - x$, $\tilde{\theta} = \hat{\theta} - \theta$, and $\tilde{\sigma} = \hat{\sigma} - \sigma$. The dynamic's error is as follows

$$\dot{\tilde{x}} = A_m \tilde{x} + b(\tilde{\theta} \| x \|_{\infty} + \tilde{\sigma}), \quad \tilde{x}(0) = 0$$
(13)

By choosing the Lyapunov function as follows

$$V(\tilde{x}, \tilde{\theta}, \tilde{\sigma}) = \tilde{x}^{\top} P \tilde{x} + \frac{1}{\Gamma} (\tilde{\theta}^{\top} \tilde{\theta} + \tilde{\sigma}^{\top} \tilde{\sigma})$$
(14)

The gradient of the Lyapunov function Equation (14) can be calculated as below

$$\dot{V}(\tilde{x},\tilde{\theta},\tilde{\sigma}) = \dot{\tilde{x}}^{\top} P \tilde{x} + \tilde{x}^{\top} P \dot{\tilde{x}} + \frac{2}{\Gamma} (\tilde{\theta}^{\top} \dot{\tilde{\theta}} + \tilde{\sigma}^{\top} \dot{\tilde{\sigma}}) = -\tilde{x}^{\top} Q \tilde{x} + 2 \tilde{x}^{\top} P b(\tilde{\theta} \| x \|_{\infty} + \tilde{\sigma}) + \frac{2}{\Gamma} (\tilde{\theta}^{\top} \dot{\tilde{\theta}} + \tilde{\sigma}^{\top} \dot{\tilde{\sigma}})$$
(15)

By an upper bound Lyapunov function as follows

$$\dot{V}(\tilde{x},\tilde{\theta},\tilde{\sigma}) = -\tilde{x}^{\top}Q\tilde{x} + 2\tilde{x}^{\top}Pb(\tilde{\theta}||x||_{\infty} + \tilde{\sigma}) + 2(\tilde{\theta}^{\top}\operatorname{Proj}(\tilde{\theta}, -||x||_{\infty}bP\tilde{x}) + \tilde{\sigma}^{\top}\operatorname{Proj}(\tilde{\theta}, -bP\tilde{x}))$$

$$= -\tilde{x}^{\top}Q\tilde{x} + 2\tilde{\theta}^{\top}(||x||_{\infty}bP\tilde{x} + \operatorname{Proj}(\tilde{\theta}, -||x||_{\infty}bP\tilde{x})) + 2\tilde{\sigma}^{\top}(bP\tilde{x} + \operatorname{Proj}(\tilde{\sigma}, -bP\tilde{x}))$$

$$\leq -\tilde{x}^{\top}Q\tilde{x}$$
(16)

where the adaptation law is obtained as

$$\tilde{\theta} = \hat{\theta} = \Gamma \operatorname{Proj}(\tilde{\theta}, -\|x\|_{\infty} b P \tilde{x})$$
$$\tilde{\sigma} = \hat{\sigma} = \Gamma \operatorname{Proj}(\tilde{\sigma}, -b P \tilde{x})$$
(17)

where the operator's projection is denoted by Proj as in [34], the rate of the adaptation law is represented by $\Gamma > 0$, $P = P^{\top} > 0$, with $Q = Q^{\top} > 0$ fulfilling the equation

$$A_m^\top P + P A_m = -Q \tag{18}$$

The adaptive control signal Laplace transform u_{ad} is given by

$$u_{ad}(s) = -\frac{C(s)}{\omega}(\hat{\mu}(s) - k_g r(s))$$
(19)

where r(s) and $\hat{\mu}(s)$ represent the Laplace of the reference r(t) and $\hat{\mu}(t) \triangleq \hat{\theta}(t) ||x||_{\infty} + \hat{\sigma}(t)$, subsequently. $k_g \triangleq -\frac{1}{c^\top A_m^{-1}b}$ represents the feed-forward gain. By choosing the filter C(s) as follows

$$C(s) \triangleq \frac{\omega k D(s)}{1 + \omega k D(s)} \tag{20}$$

We can rewrite Equation (20) after choosing $D(s) = \frac{1}{s}$ as follows

$$C(s) = \frac{\omega k}{s + \omega k} \tag{21}$$

The adaptive control signal can be calculated by substituting Equation (21) into Equation (19) as follows

$$u_{ad}(s) = -kD(s)(\omega u_{ad}(s) + \hat{\mu}(s) - k_g r(s))$$
(22)

Finally, based on [33], the next two conditions must be satisfied for the \mathcal{L}_1 controller:

$$\|G(s)\|_{\mathcal{L}_1} L < 1 \tag{23}$$

where

$$G(s) \triangleq H(s)(C(s)-1), \ H(s) \triangleq (sI - A_m)^{-1}b$$
(24)

and

$$\left\|\frac{\partial f(t,x)}{\partial x}\right\| \le d_{f_x} = L \tag{25}$$

the nonlinear \mathcal{L}_1 adaptive control is available in [22,32,33].

3.2. Containment Control

The containment control model is introduced in this section. Suppose we have an N vehicles dynamic in 2D as follows:

$$\dot{x}_{i}(t) = A_{uav}x_{i}(t) + B_{uav}u_{i}(t)$$

$$y_{i}(t) = C_{uav}x_{i}(t) \quad i = 1, 2...N \text{ and } x_{i}(t) \in \mathbb{R}^{2}$$
(26)

where

$$A_{uav} = diagonal\left(\begin{bmatrix} 0 & 1\\ a_{21}^1 & a_{22}^1 \end{bmatrix} \dots \begin{bmatrix} 0 & 1\\ a_{21}^n & a_{22}^n \end{bmatrix}\right),$$
$$B_{uav} = I_n \otimes \begin{bmatrix} 0\\ 1 \end{bmatrix}, x_i(t) = \begin{bmatrix} x_p\\ x_v \end{bmatrix} and \ \dot{x}_i(t) = \begin{bmatrix} x_v\\ x_a \end{bmatrix}$$

Now Equation (26) can be simplified to:

$$\begin{bmatrix} x_v \\ x_a \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_p \\ x_v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i$$
(27)

Definition 1. "A formation in Figure 2 is a vector $h = h_p \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \in \mathbb{R}^{2N}$. The N agents are in formation h if there are vectors $q, w \in \mathbb{R}^N$ such that $(x_p)_i - (h_p)_i = q, (x_v)_i = w, i = 1, ..., N$ where subscript p (position) and subscript v (velocity) are components of x_i " [35].



Figure 2. Containment of multi agents.

Furthermore, the error between agents is determined from an average relative movement of adjacent agents as follows:

$$e_i = (x_i - h_i) - \frac{1}{|J_i|} \sum_{j \in J_i} (x_j - h_j) \quad i = 1, \dots, N.$$
 (28)

We can derive the output vector of all vehicles equations in a single system by using the Laplacian matrix, L_G as follows:

$$\dot{x} = Ax + Bu \tag{29}$$
$$e = L(x - h)$$

with $A = I_N \otimes A_{uav}$, $B = I_N \otimes B_{uav}$ and $L = L_G \otimes I_{2n}$.

Our goal is to calculate feedback matrix F, a matrix of feedback that lead the vehicle to a formation h.

$$\dot{x} = Ax + BFL(x - h) \tag{30}$$

By substituting *A*, *B* and $F = I_N \otimes F_{uav}$ we obtain:

$$\dot{x}(t) = I_N \otimes A_{uav} x(t) + L_G \otimes B_{uav} F_{uav}(x(t) - h)$$
(31)

The result would be the corresponding forms of matrix, *A*, *B*, and *F* as follows:

$$(U^{-1} \otimes I_{2n})(A + BFL)(U \otimes l_{2n}) = I_N \otimes A_{uav} + \tilde{L}_G \otimes B_{uav}F_{uav}$$
(32)

Equation (32) right-hand side in the form of $A_{uav} + \lambda B_{uav}F_{uav}$ where λ : eigenvalues of L_G . With the assumption of a second order system, the feedback matrix is in the form of $F = I_N \otimes (f_1, f_2)$. Where, we can choose f_1 to solve f_2 form $\frac{(a_{22}+\lambda f_2)^2}{4\lambda} < -f_1$ for nonzero eigenvalues. A detailed of vehicle formation based graph theory is available in [35].

4. Simulation Results

The \mathcal{L}_1 adaptive controller, DDS middleware and graph theory are implemented. The quadrotor model parameters selected as in Table 1 [32].

Properties	Symbol	Value/Unit
Mass	m	0.52 kg
Gravity Acceleration	g	9.8 m/s^2
Drag's Translational	$\bar{k_t}$	0.95
Drag's Rotational	k_r	0.105
Ratio of Drag & Thrust	d	$7.5 imes10^{-7}~\mathrm{kg}\cdot\mathrm{m}^2$
Inertia of <i>x</i> -axis	I_X	0.0069 kg · m ²
Inertia of <i>y</i> -axis	$I_{\mathcal{V}}$	0.0069 kg · m ²
Inertia of <i>z</i> -axis	$\check{I_z}$	$0.0129 \text{ kg} \cdot \text{m}^2$
Arm Length	L	0.205 m
Propeller Inertia	I_R	$3.36 imes 10^{-5} \text{ kg} \cdot \text{m}^2$

Table 1. The quadrotor model parameters.

In addition, the controller parameters are selected as $\gamma = 10^6$, $k_p = 10$ and $k_d = 10$. The communication topology between the agents is shown in Figures 3 and 4 by considering a set of quality of service policies in Table 2 that need to be satisfied before the communication between the publishers and the subscribers.

A couple of cases were considered in the containment control results:

• **Containment of UAVs without parameters uncertain:** A comparison between the backstepping controller and the \mathcal{L}_1 controller for the containment of multiple UAVs is illustrated in the figures from Figures 5–8. This presents how the group of five UAVs can formalize in specific topology inside their leaders in 2D space without uncertainty in the inertia matrix.

• **Containment of UAVs with parameters uncertain:** Uncertainty of parameters (mass and UAV inertia matrix) has not been considered in the Figures 5 and 8, although it is now under consideration. The same comparison is shown in the figures from Figures 9–12. It depicts the 2D space of the group of five UAVs with uncertainty in the inertia matrix.



Figure 3. The graph topology.



Figure 4. The Unmanned Aerial Vehicles (UAVs) structure based on Data Distribution Service (DDS) middleware.

Tuble 1 : Quality of services (Q00) policies.		
QoS Policies	QoS Value	
	Publisher / Subscriber	
Durability	Volatile	
Reliability	Reliable	
History	Keep All	
Deadline	Infinite	

Table 2. Quality of services (QoS) policies.



Figure 5. UAVs containment at the first position based on L_1 adaptive control without parameters uncertain.



Figure 6. Full map of UAVs containment based on L_1 adaptive control without parameters uncertain.



Figure 7. UAVs containment at the first position based on Backstepping control without parameters uncertain.



Figure 8. Map of UAVs containment based on backstepping control without parameters uncertain.

The two Figures 5 and 6 illustrate the containment control of multi-UAVs without uncertainty in the parameters (mass and UAV inertia matrix) based on L1 adaptive control. It can be seen that all the UAVs were constituted in a star topology inside the four red leaders. On the other hand, Figures 7 and 8 illustrate the containment control of multi-UAVs without uncertainty in the parameters based on backstepping control. It is clear that the same results were obtained with lower performance than the L1 adaptive control and that is shown in Figure 7. Exchanging the information between the UAVs and the leaders was done through the DDS middleware.



Figure 9. UAVs containment at the first position based on L_1 adaptive control with parameters uncertain.



Figure 10. Full map of UAVs containment based on L_1 adaptive control with parameters uncertain.



Figure 11. UAVs containment at the first position based on backstepping control with parameters uncertain.



Figure 12. Map of UAVs containment based on backstepping control with parameters uncertain.

Figures 9 and 10 show the containment control of multi-UAVs with uncertainty in the parameters (mass and UAV inertia matrix) based on L1 adaptive control. It can be seen that all the UAVs were constituted in a star topology inside the four red leaders. The system's performance was not affected by this large variation in the mass and UAV inertia matrix. On the other hand, Figures 11 and 12 illustrate the containment control of multi-UAVs with uncertainty in the parameters based on backstepping control. It can be seen that all the UAVs were not constituted in a star topology inside the four red leaders. The system's performance was affected by these large variations in the mass and UAV inertia matrix. As in the first scenario, the exchange of information between the UAVs and the leaders was achieved through the DDS middleware.

5. Conclusions

A novel framework for the containment control of multiple UAVs is presented in this paper. Multiple UAVs containment control is developed using the \mathcal{L}_1 controller, DDS middleware, and graph theory technique. The UAVs' dynamic model stability was provided by utilizing the \mathcal{L}_1 adaptive control. Consequently, there was an exchange of information via the DDS middleware between the UAVs. In addition, the graph theory technique is used to manage the positions of the UAVs and to keep them to their preferred paths concerning their leaders. A high level of performance was shown by utilizing the \mathcal{L}_1 adaptive control in the case of sharing the information between the UAVs through the DDS middleware. The \mathcal{L}_1 controller and DDS middleware enhanced the performance and this was proven by the simulation results. The wind disturbance and quaternions in the UAVs' dynamic model will be considered in future work.

Author Contributions: Conceptualization, S.M.E., B.A.-M. and S.E.-F.; methodology, S.M.E., S.E.-F. and B.A.-M.; software, S.M.E.; validation, S.M.E., S.E.-F. and B.A.-M.; formal analysis, S.M.E.; investigation, S.M.E.; resources, B.A.-M.; data curation, S.M.E.; writing–original draft preparation, S.M.E., B.A.-M. and S.E.-F.; writing–review and editing, S.M.E., B.A.-M. and S.E.-F.; visualization, S.M.E., S.E.-F. and B.A.-M.; project administration, S.E.-F. All authors have read and agree to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The author would like to acknowledge the support of King Fahd University of Petroleum and Minerals.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Al-Buraiki, O.; El Ferik, S. Neuro-adaptive formation maintenance and control of nonholonomic mobile robots. In Proceedings of the International Conference of Control, Dynamic Systems, and Robotics, Ottawa, ON, Canada, 15–16 May 2014.
- Al-Buraiki, O.; El Ferik, S. Robotic Leader-Follower Navigation and Fleet Management Control Method. U.S. Patent 9,146,561, 29 September 2015.
- 3. Tayeb, A. L1 Adaptive Controller for Underwater Vehicle-Manipulator Systems. Master's Thesis, King Fahd University of Petroleum and Minerals (Saudi Arabia), Dhahran, Saudi Arabia, 2013. [CrossRef]
- 4. Eltayeb, A.; Rahmat, M.; Basri, A.; Mahmoud, M. An Improved Design of Integral Sliding Mode Controller for Chattering Attenuation and Trajectory Tracking of the Quadrotor UAV. *Arab. J. Sci. Eng.* **2020**. [CrossRef]
- 5. Eltayeb, A.; Rahmat, M.; Eltoum, M.; Basri, M. Robust Adaptive Sliding Mode Control Design for Quadrotor Unmanned Aerial Vehicle Trajectory Tracking. *Int. Comput. Digit. Syst.* **2020**, *9*, 249–257.
- 6. Eltayeb, A.; Rahmat, M.F.; Eltoum, M.M.; Basri, M.A.M. Adaptive Fuzzy Gain Scheduling Sliding Mode Control for quadrotor UAV systems. In Proceedings of the 2019 8th International Conference on Modeling Simulation and Applied Optimization (ICMSAO), Manama, Bahrain, 15–17 April 2019; pp. 1–5.
- Roman, R.C.; Precup, R.E.; Bojan-Dragos, C.A.; Szedlak-Stinean, A.I. Combined Model-Free Adaptive Control with Fuzzy Component by Virtual Reference Feedback Tuning for Tower Crane Systems. *Procedia Comput. Sci.* 2019, 162, 267–274. [CrossRef]
- 8. Lee, K.; Choi, Y.; Park, J. Backstepping Based Formation Control of Quadrotors with the State Transformation Technique. *Appl. Sci.* **2017**, *7*, 1170. [CrossRef]
- Coninck, E.D.; Bohez, S.; Leroux, S.; Verbelen, T.; Vankeirsbilck, B.; Dhoedt, B.; Simoens, P. Middleware Platform for Distributed Applications Incorporating Robots, Sensors and the Cloud. In Proceedings of the 2016 5th IEEE International Conference on Cloud Networking (Cloudnet), Pisa, Italy, 3–5 October 2016; pp. 218–223. [CrossRef]
- 10. El-Ferik, S.; Elkhider, S.M.; Ghommam, J. Adaptive containment control of multi-leader fleet of underwater vehicle-manipulator autonomous systems carrying a load. *Int. J. Syst. Sci.* **2019**, 7721. [CrossRef]

- 11. Zhang, H.; Liu, X.; Ji, H.; Hou, Z.S.; Fan, L. Multi-Agent-Based Data-Driven Distributed Adaptive Cooperative Control in Urban Traffic Signal Timing. *Energies* **2019**, *12*, 1402. [CrossRef]
- 12. Koubaa, A.; Dios, J. *Cooperative Robots and Sensor Networks* 2015; Springer International Publishing: Midtown Manhattan, NY, USA, 2015; Volume 604, [CrossRef]
- 13. Luo, J.; Zhang, L. Design of a Cloud Robotics Middleware based on Web. In Proceedings of the 18th International Conference on Advanced Robotics, Hong Kong, China, 10–12 July 2017. [CrossRef]
- 14. Vidal, I.; Bellavista, P.; Sanchez-aguero, V.; Garcia-reinoso, J.; Valera, F.; Nogales, B.; Azcorra, A. Data-Centric Communications. *Sensors* **2018**, *18*, 3421. [CrossRef]
- 15. Web, Q.d.R.A.b. Loosely Coupled Cloud Robotic Framework for Service Composition. *IEEE Syst. J.* **2019**, 14, 1245–1256.
- 16. Xiong, Q.; Lin, P.; Ren, W.; Yang, C.; Gui, W. Containment Control for Discrete-Time Multiagent Systems with Communication Delays and Switching Topologies. *IEEE Trans. Cybern.* **2019**, *49*, 3827–3830. [CrossRef]
- 17. Yang, C.; Duan, M.; Lin, P.; Ren, W.; Gui, W. Distributed containment control of continuous-time multiagent systems with nonconvex control input constraints. *IEEE Trans. Ind. Electron.* **2019**, *66*, 7927–7934. [CrossRef]
- Yang, J.; Xiao, F.; Ma, J. Model-based edge-event-triggered containment control under directed topologies. *IEEE Trans. Cybern.* 2019, 49, 2556–2567. [CrossRef] [PubMed]
- 19. Yang, Y.; Modares, H.; Wunsch, D.C.; Yin, Y. Optimal Containment Control of Unknown Heterogeneous Systems With Active Leaders. *IEEE Trans. Control Syst. Technol.* **2019**, *27*, 1228–1236. [CrossRef]
- 20. Wang, Z.; Wang, D.; Wang, D. Adaptive Fuzzy Containment Control for Multiple Uncertain Euler-Lagrange Systems with an Event-based Observer. *IEEE Trans. Fuzzy Syst.* **2019**. [CrossRef]
- 21. Zheng, W.X.; Shao, J.; Shi, L. Asynchronous containment control of high-order multi-agent systems with switching topologies. In Proceedings of the IEEE International Symposium on Circuits and Systems, Sapporo, Japan, 26–29 May 2019. [CrossRef]
- 22. El-Ferik, S.; Almadani, B.; Elkhider, S.M. Formation Control of Multi Unmanned Aerial Vehicle Systems Based on DDS Middleware. *IEEE Access* **2020**, *8*, 44211–44218. [CrossRef]
- 23. Wang, W.; Tong, S. Observer-based Adaptive Fuzzy Containment Control for Multiple Uncertain Nonlinear Systems. *IEEE Trans. Fuzzy Syst.* 2019. [CrossRef]
- 24. Li, Z.; Li, T.; Yuan, R.; Fei, S. Further Results on Containment Control for Multi-Agent Systems with Variable Communication Delay. *Arab. J. Sci. Eng.* **2019**, *44*, 2665–2677. [CrossRef]
- 25. Xu, Y.; Luo, D.; Li, D.; You, Y.; Duan, H. Affine formation control for heterogeneous multi-agent systems with directed interaction networks. *Neurocomputing* **2019**, *330*, 104–115. [CrossRef]
- Guzey, H.M.; Dierks, T.; Jagannathan, S.; Acar, L. Modified Consensus-based Output Feedback Control of Quadrotor UAV Formations Using Neural Networks. J. Intell. Robot. Syst. Theory Appl. 2019, 94, 283–300. [CrossRef]
- 27. Zuo, S.; Song, Y.; Lewis, F.L.; Davoudi, A. Adaptive output containment control of heterogeneous multi-agent systems with unknown leaders. *Automatica* **2018**, *92*, 235–239. [CrossRef]
- 28. Qian, D.; Zhang, G.; Chen, J.; Wang, J.; Wu, Z. Coordinated formation design of multi-robot systems via an adaptive-gain super-twisting sliding mode method. *Appl. Sci.* **2019**, *9*, 4315. [CrossRef]
- 29. Wei, H.; Lv, Q.; Duo, N.; Wang, G.; Liang, B. Consensus algorithms based multi-robot formation control under noise and time delay conditions. *Appl. Sci.* **2019**, *9*, 1004. [CrossRef]
- Xia, G.; Xia, X.; Zhao, B.; Sun, C.; Sun, X. Distributed Tracking Control for Connectivity-Preserving and Collision-Avoiding Formation Tracking of Underactuated Surface Vessels with Input Saturation. *Appl. Sci.* 2020, 10, 3372. [CrossRef]
- Nguyen, A.T.; Nguyen, T.B.; Hong, S.K. Dynamic Event-Triggered Time-Varying Formation Control of Second-Order Dynamic Agents: Application to Multiple QuadcoptersSystems. *Appl. Sci.* 2020, 10, 2814. [CrossRef]
- 32. Koesdwiady, A.B. Immersion and Invariance Control Design for Unmanned Aerial Vehicle. Master's Thesis, King Fahd University of Petroleum and Minerals (Saudi Arabia), Dhahran, Saudi Arabia, 2013.
- Hovakimyan, N.; Cao, C. L₁ Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2010; Volume 21.

- 34. Pomet, J., Praly, L. Adaptive nonlinear regulation: Estimation from the Lyapunov equation. *IEEE Trans. Autom. Control* **1992**, *37*, 729–740. [CrossRef]
- 35. Lafferriere, G.; Caughman, J.; Williams, A. Graph theoretic methods in the stability of vehicle formations. In Proceedings of the 2004 American Control Conference, Boston, MA, USA, 30 June–2 July 2004; Volume 4, pp. 3729–3734. [CrossRef]



 \odot 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).