Abstract—The problem of multi-agent formation control with $H_\infty$ robustness incorporated is addressed. A distributed control architecture is proposed for the formation system. The architecture consists of three layers where the control algorithms are divided into two parts. One of them is the cooperative part with an information flow law that endows the overall system with cooperative behaviors. Studies on the information flow law focus on discrete consensus algorithms based on the tridiagonal toeplitz matrix. The other part is the stabilization part with control law that robots depend on to maintain the formation while moving. Upon the proposed structure, a multi-agent formation system is modeled as a spatially interconnected system and spatially distributed controllers that are robust to external disturbances are developed. The effectiveness of the proposed architecture, as well as the distributed controllers, is verified by a group of wheeled robots.

I. INTRODUCTION

The study of formations for a group of agents is inspired by the motion of various animal species in nature. For instance fish, birds and ants always work in a cooperative manner so as to accomplish tasks that are beyond the capability of a single one. Formation for agents has its wide range of applications both in civilian and military life, such as aerial vehicles flying together to protect themselves against external aggressions and robots moving in formation to map out parts of the ocean floor. Agents in a formation with various sensors embedded are always competent in difficult work. Discussions over this topic include three basic parts: the initializing formation where graph rigidity is known as the fundamental property to a geometric communication topology [1], the development of cooperative algorithms that concern with an exact assignment [2], and the deployment of controllers that are responsible for formation maintenance. Researches over the first issue focus on building up an appropriate communication topology that supports our studies over the other two. In this paper, we choose a communication architecture with the underlying graph being a minimal rigid one such that the formation is translational and rotational invariant, and focus on the later two issues by means of cooperative protocols and distributed controllers.

Information flow law that proposed by Fax et al. [3] is designed to guide the evolution of variables that transmitted along communication channels such that all nodes in a network reach an agreement using only local information. Later this algorithm was merged with another algorithm known as the consensus algorithm which has wide range of applications in the field of distributed multi-agent formation control where agents are always expected to have a common knowledge over certain variables such as the heading, velocity and even coordinates of the virtual center [2], [4].

Robustness is a crucial property in practical applications for multi-agent systems over networks [5]. For instance, when robots move in a real-time situation, stochastic environmental uncertainties such as wind or road bumps would possibly hinder the robots from accomplishing their assignments or even destroy the balance of the system. Wireless network-based multi-robot cooperation system raised higher requirements on reliability under lossy links. Thus a formation system that are robust to uncertainties including parameter perturbations of robots, structural perturbations of underlying graph, external disturbances and network dynamics are required in practical applications. However, to our knowledge, literatures concerning this topic are relatively rare.

To this end, we build a distributed control architecture with $H_\infty$ robustness incorporated. The multi-agent system is modeled as a spatially interconnected system, as introduced in Section II. Some useful results of algebraic graph theory are introduced in this section as well. Section III bring forward a complete distributed control architecture for multi-agent systems where Toeplitz-based discrete consensus algorithm, as well as convergence rate, is proposed for system over 2-hop graph. Moreover, improvements are made upon existing synthesis method of $H_\infty$ distributed controllers so as to broaden its applications. Section IV is the experimental study of the proposed distributed framework, as well as the $H_\infty$ distributed controllers, which was verified on the testbed consisting of wheeled robots and computers with the objective of maintaining a formation in a plane. Finally concluding remarks are presented in Section V.

II. PRELIMINARIES

In this section, we give a basic introduction to the state-space model of spatially interconnected systems (SISs), which will be used in the multi-robot formation control. Readers are refereed to [6] for more technical details. Graph theory is a crucial tool for the discussion of consensus algorithms and thus some important results are presented in this section as well.

A. Spatially Interconnected Systems

In parallel to [6], we consider vector valued signals $U = \{u(t, s_1, s_2, \ldots, s_L)\}$, where $t$ is a time-based variable, i.e. $t \in \{\mathbb{R}^+, 0\}$, $s_i \in \mathbb{Z}$ and $L$ is the dimension of the spatial coordinates. For simplification, the $L$-tuple $(s_1, s_2, \ldots, s_L)$ is denoted by $s$, which uniquely indicate a location in the $L$
dimensional spatial coordinates. Particularly, when $L = 1$, we get the one dimensional spatial interconnected systems, such as systems over a chain topology.

We adopt the hybrid Roesser model [7], [8] to describe subsystems in a spatially interconnected system (SIS). By considering $r$ in the Roesser model as the dimension of the subsystem’s states and $(k - r)$ as the dimension of the interconnected signals, it is straightforward of rewrite the hybrid Roesser model into the form that represents the $s$-th subsystem of a SIS:

$$\begin{bmatrix} x(t,s) \\ w(t,s) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x(t,s) \\ v(t,s) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \begin{bmatrix} v(t,s) \\ w(t,s) \end{bmatrix}$$

(1)

where $x(t,s)$ is the state variables of the $s$-th subsystem, and $d(t,s)$ and $z(t,s)$ denote its disturbance and its consequent output, respectively. $v(t,s)$ and $w(t,s)$ are the bidirectional interconnection variables mapped by the spatial shift operator[6]:

$$w(t,s) = \begin{bmatrix} w_+(t,s) \\ w_-(t,s) \end{bmatrix} = \begin{bmatrix} S_{m+} & 0 \\ 0 & S^{-1}_{m+} \end{bmatrix} \begin{bmatrix} v_+(t,s) \\ v_-(t,s) \end{bmatrix}$$

(2)

with $m_+$, $m_-$ being the size of interconnection variables $v_+(t,s)$ and $v_-(t,s)$ respectively.

We may thus rewrite the interconnected system in (1) and (2) as follows:

$$\begin{bmatrix} \dot{x}(t,s) \\ (\Delta s\gamma v(t,s))(s) \end{bmatrix} = \begin{bmatrix} A_{TT} & A_{TS} & B_T \\ A_{ST} & A_{SS} & B_S \\ C_T & C_S & D \end{bmatrix} \begin{bmatrix} x(t,s) \\ v(t,s) \\ d(t,s) \end{bmatrix}$$

(4)

A closed-loop interconnected system with distributed controllers that inherits the same interconnection topology as the plants. The system structure are shown in [8].

B. Algebraic Graph Theory

A graph $G = (V, E)$ is introduced to denote the underlying graph of a formation that consists of $n$ vehicles. $V = \{v_i : i = 1, 2, \ldots, n\}$ is the set of vertices in the graph and $E = \{e_{ij} : e_{ij} = (v_i, v_j), v_i, v_j \in V\}$ is the set of edges. $|E|$ and $|V|$ denote the number of edges and vertices respectively. We consider undirected graph in this paper, meaning $e_{ij} \in E$ if and only if $e_{ji} \in E$. A graph with $E_c = \{e_{ij}, i, j \in V, |i - j| = 1\}$ and $E_hc = E_c \cup \{e_{im}, m, n \in V, |m - n| = 2\}$ are called the undirected chain graph $G_c$ and 2-hop graph $G_hc$ respectively. Set $N(i) = \{v_j \in E : (v_i, v_j) \in E\}$ be the neighbors of agent $i$. The out-degree(in-degree) of $v_i$ in an undirected graph, denoted by $d_+(v_i)(d_-(v_i))$, is the number of edges whose start points(end points) are $v_i$. An adjacency matrix of graph $G$ is denoted by $A(G) \in R^{n \times n}$ that satisfies $a_{ij} = 1$, for $(v_i, v_j) \in E$ and 0 otherwise. The out-degree matrix of a graph G is $D(G)$ with $d_i = \sum_{j \neq i} a_{ij}$ and $d_j = 0, i \neq j$, and the Laplacian of a graph $G$ is defined as $L(G) = D(G) - A(G)$. The notation $\Gamma(G)$ is usually omitted when it is clear from context.

Proposition 2.1: Zero is an eigenvalue of $L$ with $1^T$ being its right eigenvector.

III. MAIN RESULTS

In this section, we propose a distributed control architecture for multi-agent formation systems. Fig. 1 gives an example for the case that the underlying graph for the formation is a chain graph. The proposed architecture consists of three layers with the middle one being an information flow layer. In this layer distributed algorithms are implemented so as to endow the system with cooperative behaviors. The base one is a stabilization layer with, for instance, proportional controllers that guarantee the stability of the overall system. This architecture gives an hierarchical and explicit way to observe the internal structure of the distributed cooperation systems.

Definition 3.1: A complete distributed control architecture in multi-agent formation systems is a framework with the following properties:

- It is a three-layer framework;
- The middle layer is termed information flow layer with the purpose of shaping the cooperative behaviors according to assigned tasks;
- The base layer is termed stabilization layer which has exactly the same communication topology as the agents and guarantees the convergence of the formation.

Most of current literatures on the topic of distributed multi-agent formations were concerned themselves with interconnection rules or consensus algorithms, and thus are limited to the middle layer. However, in those literatures, formations were stabilized under decentralized controllers such as some proportional controllers [4] or PI controllers [9]. The architecture they followed is the one shown in Fig. 1(b). In contrast, here we introduce dynamic output feedback controllers into the base layer and deploy them over a network topology that is identical to the one of the agents. This could increase the sensitivity of the formation system, that is, malfunction of a single agent would immediately be reflected to all the other ones as well as their controllers, which results in an immediate response. The structure of the system follows the one in Fig. 1(a).

In the hierarchical framework we build above, the information flow law is isolated from the stabilization controllers. This property allows one to study on the specific part of the control algorithm without affecting the other part. For instance in a consensus-based formation system using the virtual structure approach, we can test the control effects of different types of controllers while adopt the same consensus algorithm.
The remaining context of this article will focus on the system control architecture in Fig. 1(a), and will discuss these two layers separately.

### A. Information Flow Loop

We consider formation persistence over distributed networks while tracking a desired trajectory with transparency and rotation. Information of the desired trajectory is only known to a small group of agents, sometimes even to a single one. Thus distributed consensus algorithms that converge to a desired value, which is the coordinates of the virtual center in most of cases rather than weighted average value of the initial values, are required. In this section, the underlying graph defining the communication topology of a formation is chosen to be a 2-hop graph as shown in Fig. 2. Note that we focus on situation where only one agent has access to the coordinates of the desired trajectories.

**Theorem 3.1:** An undirected 2-hop graph $G_{hc} = (V,E_{hc})$ is a minimal rigid graph.

**Proof:** A $G_{hc}$ has exactly $2|V| - 3$ edges. Among all the induced subgraphs $G_{hc} = (V,E_{hc})$, there are two situations one of which is $V = \{v_i : i = k, k+1, \cdots, k+l\}$ with $k \in [1,n]$ and $k+l \leq n$. For this class of $V$, we have $|E'| = 2|V| - 3$. Otherwise, if the consistency of $i$ in $V$ is not satisfied, $|E'| < 2|V| - 3$. According to Laman’s theory and the definition of minimal rigidity in [1], we proved our conclusion. \[\square\]

Rigidity of a graph ensures the consistency of a formation under both rotational and translational movements. That is the reason for which we chose $G_{hc}$ rather than some arbitrary graph. Before proposing the distributed consensus algorithms over $G_{hc}$, we introduce a special set of matrix called the $T$-matrix.

Set a transformation $T : t \rightarrow T$ with $t = [t_{-n+1}, t_{-n+2}, \ldots, t_{0}, t_{1}, \ldots, t_{n-1}]^T \in \mathbb{R}^{2n-1}$ and $T \in \mathbb{R}^{n \times n}$. $T$ is of the form that $[t_{0}, \ldots, t_{-n+2}, t_{-n+1}]^T$ and $[t_{0}, t_{1}, \ldots, t_{n-1}]$ are its first column and row respectively. Moreover, entries along each diagonal of $T$ that are parallel to the main diagonal are equal. Matrix $T$ is called a toeplitz matrix ($T$-matrix) \[10\].

We focus on a special set of $T$-matrix called the tridiagonal toeplitz matrices, or the $TT$-matrix denoted by $TT(t_{-1},t_{0},t_{1},n)$.

**Definition 3.2 ([11]):** Consider a discrete-time algorithm

$$x(k+1) = Ax(k)$$ (5)

with $A \in \mathbb{R}^{n \times n}$ and $\lambda(A) \in B(0,1)$.

(i) For a given $\varepsilon \in (0,1)$, there exist $k_0 \in \mathbb{R}^+$ such that for all initial conditions $x(0) \in \mathbb{R}^n$ and $k \geq k_0$,

$$\|x(k) - \bar{x}\|_2 \leq \varepsilon \|x(0) - \bar{x}\|_2.$$ (6)

$\bar{x} \in \mathbb{R}^n$ is called the convergence value; the convergence time of $x$ with respect to $A$ and $\varepsilon$ is

$$T_\varepsilon(A) = \min\{k_0|6\}$$ (7)

(ii) $x(k)$ convergence with a rate

$$\gamma(A) = \limsup_{k \rightarrow +\infty} \frac{\|x(k) - \bar{x}\|_2}{\|x(0) - \bar{x}\|_2}^{1/k}$$ (8)

**Remark 3.1:** The discrete algorithm in (5) under the assumption $\lambda(A) \in B(0,1)$ could be either an average consensus algorithm (when $A$ is stochastic) or a zero consensus algorithm (when $\lambda(A) \in B(0,1)$). Specifically, if $A$ is a doubly stochastic matrix, $\bar{x} = \frac{1}{n}1_{n \times n}x(0)$, where $1_{n \times n}$ is an $n \times n$ matrix with all entries equal to 1.

When $A$ in (5) is specified as a $TT$-matrix, the convergence of the discrete algorithm is determined according to the following theorem:

**Proposition 3.1:** Let $A$ in (5) be $A = TT(t_{-1},t_{0},t_{1},n)$ with $t_{-1},t_{0},t_{1} \in \mathbb{R}$, or equivalently,

$$x(k+1) = Tx(k)$$ (9)

solves a zero consensus algorithm with convergence time $T_\varepsilon(A)$ under the following two conditions:

$$T_\varepsilon(A) \in \begin{cases} O(n \log n + \log e^{-1}), & t_{-1} \neq 0 \text{ and } t_{1} \neq 0 \text{ and } 0 < |t_0| < 1 \\ \Theta(n^2 \log e^{-1}), & t_{-1} = t_{1} \neq 0 \text{ and } |t_0| + 2|t_1| = 1 \end{cases}$$ (10)

In multiagent formation systems, the convergence value that is identical to a desired variable is always required, especially for systems based on the virtual structure approach. For agents that are coupled under an undirected chain, the Toeplitz matrix provide us an applicable consensus algorithm:

**Theorem 3.2:** Set an affine transformation $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ of the form:

$$\bar{x}(k) = F(x(k)) = Lx(k) + \gamma(i)x_i - x_d(k),$$ (11)

where $L$ is the graph Laplacian of $G_{hc}$ and $\gamma(i)$ is a vector with the $i$-th entry equals to one and the others to zero. Suppose the desired value $x_d \in \mathbb{R}$ is known to one and only one agent $i$ in $G_{hc}$, then $\lim_{k \rightarrow +\infty} x(k) = 1_{nx_d}$ if $\bar{x}(k)$ in (9) solves a zero consensus algorithm.

**Proof:** According to (11) and (5) we have

$$\bar{L}x(k+1) = T \bar{L}x(k) + (T-I)(-\gamma)x_d$$ (12)

with $L = L + \text{diag}\{\gamma(i)\}$. It can be inferred that $L$ is invertible. Set $A = L^{-1}T\bar{L}$ and $B = L^{-1}(T-I)(-\gamma)$, then (12) can be rewritten as $x(k+1) = Ax(k) + Bx_d$, or equivalently,

$$x(k+1) - 1_{nx_d} = Ax(k) - 1_{nx_d} + (A1_n - 1_n + B)x_d.$$ (13)

Actually,

$$A1_n - 1_n + B = L^{-1}T\bar{L}1_n - 1_n + L^{-1}(T-I)(-\gamma) = L^{-1}T(L-I)'1_n + L^{-1}T\gamma - L^{-1}1\gamma - 1_n = -1_n - L^{-1}\gamma$$

According to Proposition 2.1 we know that $\sum_{j=1}^{n} \tilde{I}_{ij} = 0$, and

$$\sum_{j=1}^{n} \tilde{I}_{ij} = 1.$$ Thus $A1_n - 1_n + B = 0_n$. Moreover, when $k \rightarrow \infty$, $x(k) \rightarrow 1_{nx_d}$ if and only if $\bar{x}(k) \rightarrow 1_n0$, which is given in Proposition 3.1. \[\square\]
Fig. 2. Undirected 2-hop graph $G_{hc}$

The corresponding algorithm of Theorem 3.3 is demonstrated in Algorithm 1.

One can tell from (11) and (5) that under the affine transformation, the $TT$-matrix-based consensus algorithms (11) serve our purpose with some existing results.

**Algorithm 1** The Toeplitz-Based Consensus Algorithm

1: Initialize $x_d$ indicating the desired convergence value and $k$ indicating the index of agent that have access to $x_d$.  
2: Initialize $x[i], i=1, \ldots, n$ indicating agent $i$’s position.  
3: Initialize $l_r[k] = r_1 x[k] + r_2 x_d$ and $l_p[k] = p_1 x[k] + p_2 x_d$ indicating the output signals from agent $k$ to agent $k-1$ and $k+1$ respectively, where the coefficients are determined according to (12),  
4: Initialize $a[i]$.indicating a weighted summary of $x[l], m \in \{i, N(i)\}$  
5: while $|x[i] - x[i - 1]| > \varepsilon, i = 1, \ldots, n$ do  
6: $l_r[k] = r_1 x[k] + r_2 x_d, l_p[k] = p_1 x[k] + p_2 x_d$  
7: for $i = 0$ to $n$ do  
8: Compute $a[i]$ according to the right side of (12).  
9: end for  
10: Compute $x[i]$ based on $a[i]$ according to (12).  
11: end while

The distributed consensus algorithm in the information flow loop allows agents in the formation system reach a common knowledge of the desired value using only the local information. According to the analysis process, information flow layer is independent of the stabilization layer, and is a transplantable algorithm for various stabilization rules.

**B. Stabilization Loop with Robustness**

As we have pointed out at the beginning of this section, the stabilization loop is isolated from the interconnection loop while inhibits exactly the same structure as the multi-agent interconnected system in aspects of network topology and local dynamics, as shown in Fig. 1(a). If we treat each individual agent as a subsystem, and the cooperative algorithm as the interconnections, we bridge multi-agent cooperative systems with SISs such that distributed control architecture with $H_{\infty}$ robustness incorporated for SISs is transplantable, as discussed below. One thing that distinguishes most of the multi-agent cooperative systems from typical SISs is that agents are coupled through the tasks they are assigned rather than physically interrelated with their neighbors.

In the remaining context we will concentrate on interconnected system over symmetric graphs. Distributed control design of systems interconnected over an arbitrary graph is discussed in [12].

The state-space model of the open-loop system is presented in (4) and the distributed controllers have exactly the same structures as their related subsystems with the state-space equations as follows

$$
\begin{bmatrix}
x^k(t, s) \\
w^k(t, s) \\
u(t, s)
\end{bmatrix} =
\begin{bmatrix}
A_T^{k} & A_S^{k} & B_S^{k} \\
A_T^{k} & A_S^{k} & B_S^{k} \\
C_T & C_S & D^k
\end{bmatrix}
\begin{bmatrix}
x^k(t, s) \\
w^k(t, s) \\
u(t, s)
\end{bmatrix} +
\begin{bmatrix}
Y^g(t, s) \\
y^c(t, s)
\end{bmatrix}
$$

(14)

By canceling signals $y$ and $u$, the close-loop state space equation can be written as

$$
\begin{bmatrix}
x^k(t, s) \\
w^k(t, s) \\
u(t, s) \\
w^c(t, s) \\
w^l(t, s) \\
z(t, s) \\
d(t, s)
\end{bmatrix} =
\begin{bmatrix}
A_T^{k} & A_S^{k} & B_S^{k} \\
A_T^{k} & A_S^{k} & B_S^{k} \\
C_T & C_S & D^k
\end{bmatrix}
\begin{bmatrix}
x^k(t, s) \\
w^k(t, s) \\
u(t, s)
\end{bmatrix} +
\begin{bmatrix}
Y^g(t, s) \\
y^c(t, s)
\end{bmatrix}
$$

(15)

It is well known that for a system with dynamic output feedback controllers, the stability of the close-loop system is determined by a group of nonlinear matrix inequalities that is a non-convex NP-hard problem.

we apply an algorithm called the change variable algorithm originally proposed by Gahinet [13] to solve this problem. The details can be found in [8] and here we just give the conclusion.

**Theorem 3.3:** For a given $\gamma$, a close-loop subsystem $f_{DCE}(M^c) = MC = \{A, B, C, D\}$ is said to be well-posed, stable and satisfying $\|T_{cd}\| < \gamma$ if and only if there exist $A, B, C, D$ with suitable dimension and scalar matrices $X^g, Y^c$ such that the following two conditions are satisfied

(i)

$$
\begin{bmatrix}
\Psi_{11} & \Psi_{12} \\
\Psi_{21} & \Psi_{22}
\end{bmatrix} < 0
$$

(16)

with the shorthand notation

$$
\Psi_{11}(1, 1) =
\begin{bmatrix}
A_T^c X^g + X^g (A_T^c)^T + B_S^c C_T + (B_S^c C_T)^T \\
(A_T^c + (A_T^c + B_S^c D^c C_T)^T
\end{bmatrix}
$$

(ii)

$$
\begin{bmatrix}
\Psi_{11}(1, 2) \\
\Psi_{21}
\end{bmatrix} =
\begin{bmatrix}
A_T^c + (A_T^c + B_S^c D^c C_T)^T \\
(A_T^c + (A_T^c + B_S^c D^c C_T)^T
\end{bmatrix}
$$

$$
\Psi_{21} =
\begin{bmatrix}
(B_S^c + B_S^c D^c D^c C_T)^T \\
C_T^c X^g + D^c C_T^c C_T^c + D^c D^c C_T^c
\end{bmatrix}
$$

$$
\Psi_{22} =
\begin{bmatrix}
-\gamma \cdot (D^c C_T^c + D^c D^c D^c C_T^c) \\
-\gamma
\end{bmatrix}
$$

where $*$ follows the symmetry of the matrix.

$$
\begin{bmatrix}
X^g & I \\
I & Y^c
\end{bmatrix} \succeq 0,
$$

(17)

where

$$
X^g = (X^g)^T = \text{diag}(X^g, X^g), Y^g = (Y^g)^T = \text{diag}(Y^c, Y^c)
$$

and

$$
\hat{A} = Y^g (A_T^c + B_S^c D^c C_T^c) X^g + N B_T^c C_T^c X^g + Y^g B_S^c C_T M_T + N A_T^c M_T
$$

$$
\hat{B} = Y^g B_S^c D^c K + N B_K^c, \hat{C} = D^c C_T^c X^g + C_K^c M_T, \tilde{D} = D^c
$$

(19)
with $M, N$ satisfying

$$
\hat{X} = \begin{bmatrix}
\gamma^G & N \\
N^T & W
\end{bmatrix}, \hat{X}^{-1} = \begin{bmatrix}
\gamma^G & M \\
M^T & Z
\end{bmatrix}.
$$

And the controllers for each subsystem can be obtained by inverse substitution of (18)-(19).

We can get the $H_{\infty}$ optimal controller by minimizing $\gamma$ in (16). This can easily be implemented in the LMI-toolbox.

There are situations when the elimination algorithm in [6] results in infeasible solutions while we have to depend on the change variable algorithm. The example in Section IV has proved this point.

A formation system under the the distributed $H_{\infty}$ stabilization controllers maintains stronger robustness to the exogenous disturbance due to the synchronous interconnections between controllers. This property ensures greater reliability of the formation system under poor environment with accented ground or winds.

On the other hand, a great challenge in decentralized control of formation systems is its scalability. In general, decomposition is a most frequently used approach in decentralized control of formation systems. However the decomposition depends globally on the underlying communication topology. Local changes of the underlying graph may require new set of decentralized controllers. For example in [3] the overlapped subsystems are decoupled by the eigenvalues of the graph Laplaciant. In this paper, we introduced the spatial shift operator to model the formation system such that its stability is determined by the set of individual agents. The distributed controllers we developed upon this model are robust to the local changes of communication topology. This property is of particular importance to merging, splitting and closing ranks of the formations.

Fig. 3 presents simulation results of decentralized control and distributed control of a three-block system with spring-damper restrictions between block 1 and block 2, block 2 and block 3 respectively. White noise disturbance is imposed continuously on block 2 and the offsets of the three blocks are shown in Fig. 3. The $H_{\infty}$ performance of each block is given in Table I.

### IV. Experiments

In this section, we implement the complete distributed control architecture on three AmigoBots with the objective of maintaining a triangular formation while tracking a desired trajectory. Let $\Sigma_v$ be the coordinate system on the virtual structure where robot $i$’s desired position is represented by $\eta_{di}$. To express the positions of all the robots in a common coordinate system, we define a static coordinate system, $\Sigma_o$,

where we denote by $\eta_{vi}$ the estimated virtual center on robot $i$. Finally, robot $i$’s globally desired position in $\Sigma_o$ is calculated by $\eta_{di} = \eta_{vi} + R_{0, \theta} \eta_{ui}$, where $R_{0, \theta}$ is a rotation transformation defined by $R_{0, \theta} : \Sigma_v \rightarrow \Sigma_o$. Fig. 3 illustrates the coordinates of the formation system.

The communication topology we considered here is captured by $G_{hc}$ shown in Fig. 2. Note that when $n = 3$, $e_{ij} \in E_{hc}, \forall i, j \in V$ and $i \neq j$, which results in an undirected ring graph. Virtual structure approach [2] is adopted where agreement over coordinates of the virtual center is reached according to the triangular toepplitz-based discrete consensus algorithm shown in Algorithm 1.

We consider nonholonomic wheeled robots moving in a plane for which the Kinematic equation is a nonlinear one:

$$
\begin{bmatrix}
\dot{x}_i \\
\dot{y}_i \\
\dot{\theta}_i
\end{bmatrix} = \begin{bmatrix}
\cos \theta_i & 0 & 0 \\
\sin \theta_i & 0 & 1
\end{bmatrix} \begin{bmatrix}
v_i \\
w_i
\end{bmatrix}
$$

with $x_i,y_i$ being robot $i$’s positions in $\Sigma_o$ and $v_i,w_i$ being the linear and angular velocity of robot $i$ respectively. $\theta_i$ denotes robot $i$’s rotation with respect to $\Sigma_0$. By linearizing (20) on a fixed point $d$ off the center of the wheel axis, we get

$$
\begin{bmatrix}
v_i \\
w_i
\end{bmatrix} = \begin{bmatrix}
\cos \theta_i & \sin \theta_i \\
-\frac{1}{2} \sin \theta_i & \frac{1}{2} \cos \theta_i
\end{bmatrix} \begin{bmatrix}
\bar{x}_i \\
\bar{y}_i
\end{bmatrix}
$$

with $\bar{x}_i = x_i + d \cos \theta_i$ and $\bar{y}_i = y_i + d \sin \theta_i$.

The information flow law on robot $i$ in the networked interconnected system is captured by a double-integral linear equation with respect to the bias position $\eta_i$:

$$
\dot{\eta}_i = -2\gamma \eta_i + (-2\gamma_i) \eta_i + \sum_{j \in N(i)} (\gamma \eta_j + \gamma \eta_j) + \gamma u_i
$$

with

$$
\eta_i = \begin{bmatrix}
x_i \\
y_i
\end{bmatrix} - \eta_{di}, u_i = \begin{bmatrix}
u_{ix} \\
u_{iy}
\end{bmatrix}
$$

Set the $H_{\infty}$ performance index $\gamma = 1$ and choose $\gamma_1 = 0.56$, $\gamma_2 = 0.83$ and $\gamma_3 = 0.28$ in (22). The control signal $u_{ix}$ in (22) is calculated according to the distributed $H_{\infty}$ controller of robot $i$ along $x$ axis:

### TABLE I

$H_{\infty}$ PERFORMANCE OF THE OVERALL SYSTEM

<table>
<thead>
<tr>
<th></th>
<th>Sub1</th>
<th>Sub2</th>
<th>Sub3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decentralized</td>
<td>0.13</td>
<td>0.14</td>
<td>0.14</td>
</tr>
<tr>
<td>Distributed</td>
<td>0.01</td>
<td>0.09</td>
<td>0.02</td>
</tr>
</tbody>
</table>
In this paper, we presented a complete distributed control architecture composed of three layers for multiple agents formation control, and implemented it on a three-robot formation system. For the information flow layer, we proposed toelitz-based consensus algorithms and proved that it converged to a prescribed value known to only one node in the network. For the stabilization loop, we developed distributed controllers that were robust to external disturbances with $H_\infty$ gain $\gamma < 1$. This property ensures reliability of the formation system under poor environment. The distributed framework was a hierarchical one with independent layers that allowed flexible development. Both numerical simulations and real-time experiments were carried out on a group of three robots moving in a plane to make triangular formation. Double-integral dynamic model of nonholonomic robot was considered such that they could carry out tasks in a cooperative manner. Real-time experiments showed the validity of this distributed framework and the $H_\infty$ robustness of the overall system.

Fig. 4. Performance of the three robots with initial offsets: $e_1$=−1, $e_2$=0.3, $e_3$=−0.6, and with persistent external white noise disturbance of power 0.1 acting on robot 1.

Fig. 5. Three AmigoBots triangular formation

\[
\begin{bmatrix}
  \dot{x}_1^k \\
  \dot{x}_2^k \\
  \dot{w}_k \\
  \dot{u}_k
\end{bmatrix} =
\begin{bmatrix}
  -1.01 & -5.14 & -0.01 & -0.01 & -0.00 \\
  9.47 & -10.92 & -0.01 & -0.01 & 6.91 \\
  -22.55 & -0.14 & 0.00 & -0.88 & -2.95 \\
  -22.57 & -0.14 & -0.88 & 0.00 & -2.96 \\
  -13.70 & -108.04 & -0.08 & -0.05 & 1.58
\end{bmatrix}
\begin{bmatrix}
  x_1^k \\
  x_2^k \\
  y_v^k \\
  y_r^k \\
  y_s
\end{bmatrix}
\]

where $i$ is omitted. Recalling the distributed scheme in Fig.1, the distributed controller we developed in (24) follows the structure of Fig.1(a).

Simulations of three-robot formation system under the complete distributed framework are carried out with the results shown in Fig.4. We impose persistent white noise disturbance of power 0.1 on robot 1, and assign initial offsets for robot 1, 2 and 3 respectively, as illustrated in Fig. 4. The overall system approached stability within $t = 1\sec$ under satisfactory system performance. The $H_\infty$ robustness, i.e. the contractiveness of the system with respect to the white noise disturbance $d$ is measured by $\|z\|_2/\|d\|_2 = 0.21$.

Real-time experiments are carried out on nonholonomic wheeled robots called AmigoBots, which are differential driven robots equipped with sensors and actuators on board. The prescribed trajectory of the the virtual center is set to be a circular arc centered at $(0, 1000 + 1000\sqrt{3})(mm)$ with radius $R = 1000 + 1000\sqrt{3}(mm)$, and the desired trajectories of the three robots are denoted in dashed lines in Fig. 5(a). The desired formation is a triangle with side length of $d = 2000 mm$. Assume only robot 1 has access to the virtual center and the others agree upon the coordinates of the virtual center according to Algorithm 1. Fig.5(a) shows the trajectory of the three robots under the distributed architecture while Fig.5(b) records their absolute position errors. Because of Amigobots, intrinsic constrains, we set the error disk to 100mm. The experimental results show the effectiveness of the proposed distributed controllers.

V. CONCLUSION

In this paper, we presented a complete distributed control architecture composed of three layers for multiple agents formation control, and implemented it on a three-robot formation system. For the information flow layer, we proposed toelitz-based consensus algorithms and proved that it converged to a prescribed value known to only one node in the network. For the stabilization loop, we developed distributed controllers that were robust to external disturbances with $H_\infty$ gain $\gamma < 1$. This property ensures reliability of the formation system under poor environment. The distributed framework was a hierarchical one with independent layers that allowed flexible development. Both numerical simulations and real-time experiments were carried out on a group of three robots moving in a plane to make triangular formation. Double-integral dynamic model of nonholonomic robot was considered such that they could carry out tasks in a cooperative manner. Real-time experiments showed the validity of this distributed framework and the $H_\infty$ robustness of the overall system.

REFERENCES