Resilient Distributed Internal-Model Control Allocation Under Pinned-Agent Failures

Shahab Chehraghi[1], Mohammadreza Nematollahi[1], Khashayar Khorasani[1]

***Abstract*— Over-actuated systems can physically remain capable of meeting control demands after actuator removals, but a distributed allocator remains prone to failures if allocation is tied to lost agents. This paper addresses the architectural failure mode in pinned-agent distributed control allocation by introducing a leader-like supervisory-level virtual agent that coordinates the network, while physical agents exchange only local allocation variables with their neighbors and control their actuators. Moreover, an internal-model structure is embedded within the framework to achieve zero steady-state error in the presence of control demands generated by a known exosystem. Under graph reconfiguration that preserves feasibility and connectivity to the remaining pinned agents, the remaining actuators converge to the optimal time-varying allocation, while the failed agents' contribution vanishes. A marine vessel thrust allocation case study presents results for the removal of critical pinned and non-pinned agents.**

I. INTRODUCTION

Over-actuated systems offer greater freedom in generating control demands, thereby reducing actuator energy consumption and actuator wear & tear [1]. A supervisory control allocation scheme can generate optimal actuator references while also satisfying physical constraints and secondary performance objectives [2]. Conventional control allocation is commonly formulated as an optimization layer within the Command-and-Control (C&C) framework, thereby providing pointwise optimal actuator references.

The conventional control allocation approach leads to significant growth in online computational cost as the number of actuators and constraints increases [3], and might fail if it relies on a fixed configuration and global information [4], even if a redundant structure is feasibly fault-tolerant. Dynamic optimization-based allocation methods address part of this issue by replacing repeated static optimization with continuous-time solver dynamics whose trajectories converge asymptotically to the optimal allocation [5], [6].

Distributed optimization methods constitute another important class of dynamic optimizers that are widely applicable to areas such as resource allocation and economic dispatch [7], [8]. The core concept originates in multi-agent systems (MAS), where local agents communicate with their neighbors on a graph [9] to achieve a global objective. The distributed nature of these mechanisms offers a modular scheme that is resilient to additions and removals, while still satisfying the global objective.

Distributed control allocation methodologies [10], [11] can be viewed as a subset of distributed optimization methods. In this framework, each actuator is equipped with a local computation unit, modeled as an agent capable of making autonomous decisions. These agents communicate with neighboring actuator agents and collectively solve the underlying control allocation problem subject to global constraints.

In such networked setups, some agents may be designated as pinned agents, meaning that they directly receive supervisory information and help propagate it through the network [12]. In conventional distributed control allocation formulations, the pinned agents receive portions of the commanded reference input. The agents then coordinate by exchanging information about their local contributions and optimizing local objective functions that encode secondary performance criteria, while the overall actuator set generates the required control demand.

While the distributed optimization mechanism enforces the global equality constraint through the injected shares [13], [14], the failure or removal of any pinned agent results in a loss of the control demand share and a persistent error, as the effective command seen by the surviving distributed network is no longer the original demand. If more agents receive information to reduce sensitivity by sharing fewer portions, the communication load from the C&C will be high. Furthermore, the presence of a single pinned agent may lead to excessive dependence on its healthy performance, thereby reducing modularity.

Another limitation of the dynamics optimizers and the distributed subsets is the occurrence of persistent errors under time-varying control demands and optimization formulations. Typically, the control demands are piecewise constant, whereas in reality, they represent a time-varying signal. Consequently, the dynamics optimizers solution can result in persistent tracking errors that degrade closed-loop accuracy [15]. Recent literature on dynamic optimization has addressed this issue by embedding the internal dynamics of time-varying signals into the dynamics [16], [17].

Driven by these limitations, a distributed control allocation framework is presented here that separates command injection from physical actuation. A virtual agent is placed in the C&C layer to inject the control demand into the distributed coordination layer. In this setup, the pinned agents are used

[1]Shahab Chehraghi (s_chehra@encs.concordia.ca), Mohammadreza Nematollahi (m_nemato@encs.concordia.ca) and Khashayar Khorasani (kash@ece.concordia.ca) are with the Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada.

K. Khorasani would like to acknowledge the support received from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Department of National Defence (DND) under the Discovery Grant and DND Supplemental Programs. This work was also supported in part by funding from the Innovation for Defence Excellence and Security (IDEaS) program from DND. Any opinions and conclusions in this work are strictly those of the authors and do not reflect the views, positions, or policies of - and are not endorsed by - IDEaS, DND, or the Government of Canada.

only to maintain a path to propagate the control demand from the virtual agent to the rest of the network. Additionally, the framework incorporates an internal model of time-varying control demands and optimization variables to achieve zero steady-state error.

The main contributions of this paper are:

- A virtual-agent command injection architecture that separates command availability from physical actuator availability and preserves the full demanded input after admissible reconfiguration.
- A distributed internal-model allocation law with a convergence proof showing that the surviving agents solve the reduced time-varying allocation problem.

The remainder is organized as follows. Section II presents the problem formulation. Section III introduces the proposed solution. Section IV reports simulation results demonstrating resilience. Finally, Section V concludes the paper.

II. Problem Formulation

Let $\tau \in \mathbb{R}^p$ denote the control demand and $u_i \in \mathbb{R}^{n_i}$ denote the actuation vector presenting the i-th actuator, with the corresponding agent $i \in \mathcal{N}$. The communication network among the agents is represented by an undirected graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$, with $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ being the set of existing communication links among the agents. We denote by $\mathcal{N}_i$ the neighbor set for each agent $i \in \mathcal{N}$. We have the following standing assumptions.

Assumption 1. Each actuator effort is a vector, $u_i \in \mathbb{R}^{n_i}$, with effectiveness matrix $B_i \in \mathbb{R}^{p \times n_i}$ where $\sum_{i=1}^{n} n_i = m$. The stacked effort vector is $u = \mathrm{col}(u_1, u_2, \cdots, u_n)$ with $B = [B_1, \ldots, B_n] \in \mathbb{R}^{p \times m}$ where, B is full row rank, implying that $Bu = \tau$ is feasible for any $\tau \in \mathbb{R}^p$.

Assumption 2. The communication graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$ is undirected, connected, and weighted with $a_{ij} = a_{ji} > 0$ for $(i, j) \in \mathcal{E}$. A nonempty set of agents $\mathcal{P} \subseteq \mathcal{N}$ is pinned and can communicate directly with C&C.

Assumption 3. Each agent is assigned a quadratic cost function to minimize, i.e., for each $i \in \mathcal{N}$, $J_i(u_i) = \frac{1}{2} u_i^\top P_i u_i + q_i^\top u_i$, where $P_i \in \mathbb{R}^{n_i \times n_i} \succ 0$ and $q_i \in \mathbb{R}^{n_i}$. The overall network cost to be minimized is then $J(u) = \sum_{i=1}^{n} \frac{1}{2} u_i^\top P_i u_i + q_i^\top u_i$.

The distributed optimization-based control allocation problem to be solved is

$$\min_u \sum_{i=1}^{n} \frac{1}{2} u_i^\top P_i u_i + q_i^\top u_i \qquad \text{s.t.} \sum_{i=1}^{n} B_i u_i = \tau. \tag{1}$$

Since (1) is strictly convex with affine constraints, the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient for optimality [18]. A distributed dynamic optimizer [5] on the methodology of [14] solves (1) by

$$\begin{aligned} \dot{u}_i &= -\alpha(P_i u_i + q_i + B_i^\top \mu_i) - B_i^\top (B_i u_i - \tau_i - z_i), \\ \dot{\mu}_i &= B_i u_i - \tau_i - \beta \sum_{j \in \mathcal{N}_i} a_{ij}(\mu_i - \mu_j) - z_i \\ \dot{z}_i &= \alpha\beta \sum_{j \in \mathcal{N}_i} a_{ij}(\mu_i - \mu_j), \end{aligned} \tag{2}$$

where μ_i is the local dual variable, z_i is an integral term compensating the local mismatch. The coupling terms in (2) enforce consensus of the local dual variables μ_i on the centralized equivalent dual solution $\mu^\star$, while each u_i converges to its optimal component $u_i^\star$.

A key limitation of (2) is its dependence on pinned agents, where if a pinned agent is removed, such as due to failure or intentional malicious disconnection, then its share τ_i is no longer injected into the network, and the remaining agents may fail to realize the full demand. Moreover, (2) can only ensure asymptotic convergence to an optimal solution for fixed or slowly varying demands τ, whereas in a controlled system, the control demand commonly changes over time, tailored to the system states and control objectives. In such a scenario, as shown in [17], the dynamic optimizer should include the internal model of the time-varying parameters so that the optimizer can actually track the time-varying optimal solution of (1). Specifically, here, we consider the case where each $\tau \in \mathbb{R}^p$ is generated by an exosystem of the form

$$\dot{\zeta} = S\zeta, \quad \tau = R\zeta. \tag{3}$$

Assumption 4. The matrix $S \in \mathbb{R}^{s \times s}$ is in real Jordan canonical form, and all of its eigenvalues are semi-simple and lie on the imaginary axis. In particular, S includes one eigenvalue at the origin. There exists an arbitrary vector $Q \in \mathbb{R}^{s \times 1}$ where the pair (S, Q) is controllable and (Q, S) is observable.

This formulation represents τ_k (k-th element of τ) as a finite sum of harmonics with known frequencies, thereby capturing any signal whose Fourier series contains only those harmonics and can be used to improve tracking performance for a wide class of signals, including those generated by linear time-invariant finite-dimensional systems. We now define the main problem addressed in this paper.

Problem (Resilient distributed control allocation with time–varying demand)**.** Under Assumptions 1-4, consider the time-varying control demand $\tau(t)$ generated by (3). For any admissible sequence of agent removals/failures (possibly including pinned agents) that preserves at least one pinned agent and admits a physically feasible solution (i.e., Assumption 1 for the surviving agents), design a distributed control-allocation law using only local and neighbor information, such that after each admissible removal and reconfiguration, the surviving agents asymptotically solve the reduced allocation problem.

III. Main Results

Define the internal state for the i-th agent $w_i \in \mathbb{R}^{ps}$ denoting the p-copy stacked internal-model state with $w_i = \mathrm{col}(w_{i_1}, \ldots, w_{i_p})$ and $w_{i_k} \in \mathbb{R}^s$ associated with estimating the k-th element of τ. This internal-model method, which embeds p internal-model channels, is more robust to variations in system variables [19]. To address Problem II with the demanded control τ generated by the exosystem (S, R), we augment the network with a virtual agent placed in the C&C. Let the virtual agent be indexed 0 with dynamics given by

$$\dot{w}_0 = Fw_0 - \gamma H\tau - \sum_{j\in\mathcal{N}_0} a_{0j}\big(\beta(w_0 - w_j) + (v_0 - v_j)\big),$$
$$\dot{v}_0 = Fv_0 + \alpha\beta \sum_{j\in\mathcal{N}_0} a_{0j}(w_0 - w_j).$$

where $\alpha, \beta, \gamma > 0$ are design parameters, $F = \mathrm{I}_p \otimes S$, $H = (-F + \alpha\mathrm{I}_{ps})G$ with $G = \mathrm{I}_p \otimes Q$, and $v_i \in \mathbb{R}^{ps}$. The virtual agent does not contribute to the input, implying that $u_0 = 0$. For the rest of the agents, we use the following dynamic

$$\begin{aligned}\dot{u}_i &= -\alpha u_i - \alpha P_i^{-1} q_i - P_i^{-1} B_i^\top H^\top w_i,\\ \dot{w}_i &= Fw_i + \gamma H B_i u_i - \sum_{j\in\mathcal{N}_i} a_{ij}\big(\beta(w_i - w_j) + (v_i - v_j)\big),\\ \dot{v}_i &= Fv_i + \alpha\beta \sum_{j\in\mathcal{N}_i} a_{ij}(w_i - w_j).\end{aligned} \tag{4}$$

During operation, agents may fail to realize their intended input contribution due to faults, cyber-attacks, or physical removal. We assume that failed agents are detected locally. The communication graph is then repaired using the distributed network reconstruction strategy in [20], in which the neighbors of a failed agent restore local connectivity. Since the agents in this work represent actuator groups, we adopt the same reconstruction structure, replacing the degree-based priority with an actuation authority priority. The objective is to balance the reconstructed topology by connecting lower-authority agents to higher-authority agents outside their local connected set. For agent i, define the actuation authority score

$$\rho_i \triangleq \log\det\big(\mathrm{I}_p + B_i P_i^{-1} B_i^\top\big)\,, \tag{5}$$

where a larger ρ_i indicates larger generalized actuation authority relative to the local allocation cost. Assuming that the agents know their neighbors and second-hop neighbors, and the corresponding actuation authority scores after the failed node is removed, the reconfiguration policy is summarized as follows:

1) *Failure isolation:* When agent j fails, all links incident to j are removed and its actuator contribution is disabled. Let $\mathcal{N}_j^-$ denote the healthy neighbors of j immediately before the failure.
2) *Private locally connected set:* Each affected agent $i \in \mathcal{N}_j^-$ forms a private locally connected set $\tilde{\mathcal{C}}_i(j) \subset \mathcal{N}_j^-$, which contains itself, its healthy neighbors, and its locally known second-hop neighbors.
3) *Authority-based reconnection:* Each neighboring agent $i \in \mathcal{N}_j^-$ checks whether it has the lowest authority score within its private set $\tilde{\mathcal{C}}_i(j)$. If so, it initiates a new connection to the highest authority agent in $\mathcal{N}_j^- \setminus \tilde{\mathcal{C}}_i(j)$.

The distributed reconstruction strategy in [20] establishes connectivity recovery using local neighbor information. The actuation authority score modifies the priority used to select admissible reconnection links. Therefore, the convergence result below is stated for the connected post-reconfiguration graph. Assuming that at least one pinned agent survives, the repaired graph preserves a path to the virtual agent, and the allocator operates on the resulting reduced network.

Without loss of generality, we relabel the agents so that the set of surviving agents is $\mathcal{N}_c = \{0, 1, \cdots, n_c\}$ and the set of failed agents is $\mathcal{N}_f = \{n_c+1, n_c+2, \cdots, n\}$ with $\mathcal{N}_c \cup \mathcal{N}_f = \mathcal{N}$. For any stacked variable $x = \mathrm{col}([x]_i)_{i\in\mathcal{N}}$ (in particular $x \in \{u, \mathbf{w}, \mathbf{v}\}$), define $x_c := \mathrm{col}([x]_i)_{i\in\mathcal{N}_c}, x_f := \mathrm{col}([x]_i)_{i\in\mathcal{N}_f}, x = \mathrm{col}(x_c, x_f)$. Assume L_c and L_f are the Laplacians of the surviving and failed agents, respectively, with $L = \mathrm{blkdiag}(L_c, L_f)$ describing the overall network Laplacian. Let $P_c = \mathrm{blkdiag}(P_i)_{i\in\mathcal{N}_c}$, $P_f = \mathrm{blkdiag}(P_i)_{i\in\mathcal{N}_f}$, $P = \mathrm{blkdiag}(P_c, P_f)$, $q_c = \mathrm{col}(q_i)_{i\in\mathcal{N}_c}$, $q_f = \mathrm{col}(q_i)_{i\in\mathcal{N}_f}$, $q = \mathrm{col}(q_c, q_f)$, $B_c = \big[B_i\big]_{i\in\mathcal{N}_c}$, $B_f = \big[B_i\big]_{i\in\mathcal{N}_f}$, $\mathbf{B}_c = \mathrm{blkdiag}(B_c)$, $\mathbf{B}_f = \mathrm{blkdiag}(B_f)$, $\mathbf{B} = \mathrm{blkdiag}(B)$, $\mathbf{F} = \mathrm{I}_n \otimes F$, $\mathbf{G} = \mathrm{I}_n \otimes G$, $\mathbf{H} = \mathrm{I}_n \otimes H$, $\mathbf{L} = L \otimes \mathrm{I}_{ps}$, and similarly $\mathbf{F}_c = \mathrm{I}_{n_c+1} \otimes F$, $\mathbf{G}_c = \mathrm{I}_{n_c+1} \otimes G$, $\mathbf{H}_c = \mathrm{I}_{n_c+1} \otimes H$, $\mathbf{L}_c = L_c \otimes \mathrm{I}_{ps}$. The compact form of the network dynamics is

$$\begin{aligned}\dot{u} &= -\alpha u - \alpha P^{-1} q - P^{-1}\mathbf{B}^\top \mathbf{H}^\top \mathbf{w},\\ \dot{\mathbf{w}} &= \mathbf{F}\mathbf{w} + \gamma \mathbf{H}(\mathbf{B}u - \boldsymbol{\tau}) - \beta \mathbf{L}\mathbf{w} - \mathbf{L}\mathbf{v},\\ \dot{\mathbf{v}} &= \mathbf{F}\mathbf{v} + \alpha\beta \mathbf{L}\mathbf{w},\end{aligned} \tag{6}$$

where, with the virtual agent indexed by the first element of $\mathcal{N}_c$, we have $\boldsymbol{\tau} = [\tau^\top, 0_{1\times p}, \cdots, 0_{1\times p}]^\top$, and consistent with number of agents in $\mathcal{N}_c$ we define $\boldsymbol{\tau}_c = [\tau^\top, 0_{1\times p}, \cdots, 0_{1\times p}]^\top$. Isolating the surviving agents, we have

$$\begin{aligned}\dot{u}_c &= -\alpha u_c - \alpha P_c^{-1} q_c - P_c^{-1}\mathbf{B}_c^\top \mathbf{H}^\top \mathbf{w}_c,\\ \dot{\mathbf{w}}_c &= \mathbf{F}_c\mathbf{w}_c + \gamma \mathbf{H}_c(\mathbf{B}_c u_c - \boldsymbol{\tau}_c) - \beta \mathbf{L}_c\mathbf{w}_c - \mathbf{L}_c\mathbf{v}_c,\\ \dot{\mathbf{v}}_c &= \mathbf{F}_c\mathbf{v}_c + \alpha\beta \mathbf{L}_c\mathbf{w}_c,\end{aligned} \tag{7}$$

In this framework, the set $\mathcal{N}_c$ will satisfy the demand after the reconfiguration policy is applied, and the input contributions of $\mathcal{N}_f$ will converge to zero, leaving the realized demand unchanged. Theorem 1 now states our solution to resilient distributed control allocation with the time-varying demand problem.

Theorem 1. Suppose the reconfiguration policy yields a connected post-failure graph with a path to the virtual agent, Assumptions 1–4 hold for the surviving network $\mathcal{N}_c$, and $\tau(t)$ be generated by (3). Consider the distributed internal-model dynamics (6). The surviving dynamics (7) asymptotically solve the reduced allocation problem

$$\min_{u_c} \sum_{i\in\mathcal{N}_c} \frac{1}{2} u_i^\top P_i u_i + q_i^\top u_i \qquad \text{s.t.} \sum_{i\in\mathcal{N}_c} B_i u_i = \tau(t). \tag{8}$$

In particular, for all $i \in \mathcal{N}_c$,

$$\lim_{t\to\infty} u_{c_i}(t) = u_{c_i}^\star(t), \quad \lim_{t\to\infty} G^\top w_i(t) = \mu_c^\star(t).$$

where the $(u_c^\star(t), \mu_c^\star(t))$ pair is the optimal time-varying saddle trajectory of (8) satisfying the KKT conditions

$$P_c u_c^\star(t) + q_c + B_c^\top \mu_c^\star(t) = 0, \qquad B_c u_c^\star(t) = \tau(t),$$

and $B_f u_f(t) \to 0$ as $t \to \infty$.

Proof: The proof has two steps: (i) stability of the reduced homogeneous dynamics, and (ii) characterization of the steady-state solution and verification of the KKT conditions.

Step 1: Since $\mathcal{G}$ is undirected and connected (Assumption 2), L_c is symmetric and admits an orthogonal decomposition $T := \begin{bmatrix} r & X \end{bmatrix}^\top$ with $T^\top T = \mathrm{I}_{n_c+1}$, $TL_cT^\top = \mathrm{diag}(0, \Lambda)$, $r = \frac{1}{\sqrt{n_c+1}}\mathbf{1}_{n_c+1}$ and $\Lambda = \mathrm{diag}(\lambda_2, \ldots, \lambda_{n_c+1}) \succ 0$. Let $\mathbf{r} = r \otimes \mathrm{I}_{ps}$, $\mathbf{X} = X \otimes \mathrm{I}_{ps}$, $\mathbf{T} = \begin{bmatrix} \mathbf{r} & \mathbf{X} \end{bmatrix}^\top$, $\bar{\mathbf{F}}_c = \mathrm{I}_{n_c} \otimes F$ and $\mathbf{\Lambda}_c = \mathrm{I}_{ps} \otimes \Lambda$. Introduce the transformed coordinates [21]

$$\begin{bmatrix} y_1^\top & \mathbf{y}_2^\top \end{bmatrix}^\top = \mathbf{T}\mathbf{w}_c, \quad \begin{bmatrix} z_1^\top & \mathbf{z}_2^\top \end{bmatrix}^\top = \mathbf{T}\mathbf{v}_c. \tag{9}$$

The transformed dynamics will be given by

$$\dot{u}_c = -\alpha u_c - \alpha P_c^{-1} q_c - P_c^{-1}\mathbf{B}_c^\top \mathbf{H}_c^\top (\mathbf{r}y_1 + \mathbf{X}\mathbf{y}_2), \tag{10a}$$

$$\dot{y}_1 = Fy_1 + \gamma \mathbf{r}^\top \mathbf{H}_c(\mathbf{B}_c u_c - \boldsymbol{\tau}_c), \tag{10b}$$

$$\dot{\mathbf{y}}_2 = \bar{\mathbf{F}}_c \mathbf{y}_2 + \gamma \mathbf{X}^\top \mathbf{H}_c(\mathbf{B}_c u_c - \boldsymbol{\tau}_c) - \beta \mathbf{\Lambda}_c \mathbf{y}_2 - \mathbf{\Lambda}_c \mathbf{z}_2, \tag{10c}$$

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{F}}_c \mathbf{z}_2 + \alpha\beta \mathbf{\Lambda}_c \mathbf{y}_2, \tag{10d}$$

while $\dot{z}_1 = Fz_1$ is marginally stable and decoupled. Let the candidate Lyapunov function for the $x \triangleq \mathrm{col}(u_c, y_1, \mathbf{y}_2, \mathbf{z}_2)$ setting of homogeneous system with $\boldsymbol{\tau}_c \equiv 0$ and $q_c \equiv 0$ be

$$V(\cdot) = \frac{1}{2}u_c^\top P_c u_c + \frac{1}{2\gamma}y_1^\top y_1 + \frac{1}{2\gamma}\mathbf{y}_2^\top \mathbf{y}_2 + \frac{1}{2\alpha\beta\gamma}\mathbf{z}_2^\top \mathbf{z}_2. \tag{11}$$

Since S is in real Jordan canonical form with semi-simple imaginary axis eigenvalues, $\mathbf{F}_c + \mathbf{F}_c^\top = 0$ yields

$$\dot{V}(\cdot) = -\alpha u_c^\top P_c u_c - \frac{\beta}{\gamma}\mathbf{y}_2^\top \mathbf{\Lambda}_c \mathbf{y}_2 \le 0. \tag{12}$$

Let $\mathcal{M} := \{x : \dot{V} = 0\}$. Restricting (12) to largest invariant set $\mathcal{I} = \{x : u_c = 0, \mathbf{y}_2 = 0\}$ the equations are given by

$$\begin{aligned} &\mathbf{\Lambda}_c \mathbf{z}_2 = 0, \quad \dot{\mathbf{z}}_2 = \bar{\mathbf{F}}_c \mathbf{z}_2, \\ &\mathbf{B}_c^\top \mathbf{H}_c^\top \mathbf{r} y_1 = 0, \quad \dot{y}_1 = Fy_1, \end{aligned} \tag{13}$$

where $\mathbf{\Lambda}_c \mathbf{z}_2 = 0$ implies $\mathbf{z}_2 = 0$. $\mathbf{B}_c^\top \mathbf{H}_c^\top \mathbf{r} y_1 = 0$ implies $B_i^\top H^\top y_1 = 0$ for all $i \in \mathcal{N}_c$, hence $B_c^\top H^\top y_1 = 0$. Since B_c is full rank and $(-F^\top + \alpha \mathrm{I}_{ps})$ is nonsingular, $H^\top y_1 = G^\top(-F^\top + \alpha \mathrm{I}_{ps})y_1 = 0$ implies $G^\top y_1 = 0$. Observability of $(G^\top, F)$ and $\dot{y}_1 = Fy_1$ with $G^\top y_1 = 0$ implies $y_1 = 0$. Thus, the dynamics (10) are asymptotically stable.

Step 2: We express the constant vector q_c as an output of the same exosystem that generates the time-varying demand. Let $e \in \mathbb{R}^s$ denote the unit basis vector associated with the 0-eigenvalue block of the S matrix and define $\zeta_0 = e^\top \zeta(0)$ assuming that it is nonzero. Since $Se = 0$, we have $e^\top \dot{\zeta} = e^\top S\zeta = 0$, hence $e^\top \zeta(t) = \zeta_0$ for all t. Then

$$q_c = \Omega\zeta(t), \quad \Omega = \frac{1}{\zeta_0}\, q_c\, e^\top \in \mathbb{R}^{m_c \times s}. \tag{14}$$

where $m_c = \sum_{i \in \mathcal{N}_c} n_i$. Thus (10) can be written as $\dot{x} = \mathcal{A}x + \mathcal{B}\zeta$. Since $\mathcal{A}$ is Hurwitz and $\sigma(S) \subset j\mathbb{R}$, the Sylvester equation $\Pi S = \mathcal{A}\Pi + \mathcal{B}$ admits a unique solution Π. Defining $e_x \triangleq x - \Pi\zeta$ yields $\dot{e}_x = \mathcal{A}e_x$, hence $\lim_{t\to\infty} x(t) = \Pi\zeta(t)$. Let $Y_1\zeta$ and $U_c\zeta$ be the steady state values of y_1 and u_c. The regulator equation of y_1 in (10b) is given by

$$Y_1 S = FY_1 + \gamma H(B_c U_c - R). \tag{15}$$

Since the pair (F, G) is controllable, the pair (F, H) is also controllable. By [19, Lemma 7.4], if there exists a solution Y_1 for (15) with the controllable pair (F, H), $B_c U_c - R$ will be equal to zero, which implies the feasibility condition. With $\lim_{t\to\infty} B_c u_c = \tau$ satisfied, $\mathbf{y}_2$ and $\mathbf{z}_2$ will be stabilized. Thus, $\mathbf{w}_c = \mathbf{r}y_1$ implies a consensus in $w_i = \frac{1}{\sqrt{n_c+1}}y_1$ for all $i \in \mathcal{N}_c$. The steady state dynamics (10a) and (10b) are reduced to

$$\begin{aligned} &U_c S = -\alpha U_c - \alpha P_c^{-1}\Omega - \frac{1}{\sqrt{n_c+1}}P_c^{-1}B_c^\top H^\top Y_1, \\ &Y_1 S = FY_1 \end{aligned} \tag{16}$$

Reordering the equations implies that

$$\begin{aligned} &(U_c + P_c^{-1}\Omega + P_c^{-1}B_c^\top G^\top W_c)S \\ &= -\alpha(U_c + P_c^{-1}\Omega + P_c^{-1}B_c^\top G^\top W_c). \end{aligned} \tag{17}$$

where $W_c\zeta$ is the steady state value of w_i for all $i \in \mathcal{N}_c$. Since S and $-\alpha \mathrm{I}_{m_c}$ share no eigenvalues, the only solution to the Sylvester equation satisfies the stationarity condition $P_c U_c + \Omega + B_c^\top G^\top W_c = 0$ with $\lim_{t\to\infty} G^\top w_i(t) = \mu^\star(t)$. The optimality and feasibility conditions are proved. Finally, for each failed agent, the τ information is not injected; repeating Step 1 shows that the failed agents' demand satisfies $B_f u_f(t) \to 0$. This completes the proof of the theorem. □

Remark 1. The proposed algorithm can be viewed as a resilient time-varying extension of the implicit-tracking distributed optimization framework in [14] tailored for control allocation. To enhance the framework's resilience, a virtual-agent–based demand injection at the C&C is introduced that obviates the need for demand partitioning among pinned agents, a requirement in [14]. Meanwhile, the reconfiguration policy maintains connectivity of the surviving subgraph, allowing the network to converge to the optimal allocation for the remaining actuators, even after removing multiple agents, including pinned ones. Furthermore, by exchanging the integral compensation term v_i among neighbors, the initialization requirement on $\mathbf{v}$ is eliminated.

Remark 2. In our work, the allocation problem with a time-varying equality constraint is cast as an output-regulation problem, with a p-copy internal model embedded to track a general class of admissible demand signals. Compared with [22], which employs an additional auxiliary feedforward-estimator loop and thus requires exchanging two extra neighbor-coupled information sets, our proposed scheme requires only two sets of internal model states. Moreover, the proposed method does not require agents to access the exosystem state explicitly, while accommodating a more general coupling constraint.

Remark 3. In the distributed framework, actuator magnitude limits can be enforced locally. For projection-based distributed allocation, stability results are available in [14]. In the internal-model setting, if $\tau(t)$ remains within the local/coupled feasible region, the closed-loop dynamics follow the same analysis; when feasibility is violated, the dynamics remain bounded in practice, but a separate nonlinear analysis is required.

IV. SIMULATION RESULTS

In this section, we evaluate the proposed control allocation algorithm by considering the closed-loop dynamic positioning of a linearized marine vessel modeled as [23]

$$\dot{\eta} = \nu, \qquad M\dot{\nu} + D\nu = \tau, \tag{18}$$

where $\eta = \begin{bmatrix} \mathrm{x} & \mathrm{y} & \psi \end{bmatrix}^\top$ and $\nu = \begin{bmatrix} \mathrm{u} & \mathrm{v} & \mathrm{r} \end{bmatrix}^\top$ with

$$M = \begin{bmatrix} 23.8 & 0 & 0 \\ 0 & 33.8 & 1.095 \\ 0 & 1.095 & 2.764 \end{bmatrix}, \; D = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 7 & 0.1 \\ 0 & 0.1 & 0.5 \end{bmatrix}. \tag{19}$$

The reference trajectory $r(t)$ is generated by an exosystem $\dot{\xi} = S\xi, \; r = R\xi$ with

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0 & 0 \\ 0 & -0.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & -0.5 & 0 \end{bmatrix}, R^\top = \begin{bmatrix} -3 & 1 & -4 \\ 0 & 0 & 0 \\ 7 & 2 & -4 \\ 0 & 0 & 0 \\ 2 & 5 & 6 \end{bmatrix} \tag{20}$$

A guidance controller produces a generalized force/moment demand $\tau_{\text{in}}(t)$ of the form $\tau_{\text{in}} = K_1\eta + K_2\nu + L\xi$. In this output-regulation structure, if τ_{in} were applied directly to (18), then $\eta(t) \to r(t)$ with zero steady-state tracking error. The focus hereon is the allocation layer, where the allocator must realize $\tau_{\text{in}}(t)$ in the presence of topology changes and pinned-agent failures.

We consider eight azimuth thrusters as the main actuators to achieve the desired generalized force, with each thruster represented by an agent. The i-th actuator effectiveness matrix is given by

$$B_i = \begin{bmatrix} 1 & 0 & -\ell_{y_i} \\ 0 & 1 & +\ell_{x_i} \end{bmatrix}^\top, \tag{21}$$

where ℓ is the propeller's x and y position in the body frame as given by

$$\ell = \begin{bmatrix} 4 & 0 & 0 & 2 & 2 & -2 & -2 & -4 \\ 0 & -2 & 2 & -2 & 2 & -2 & 2 & 0 \end{bmatrix} \tag{22}$$

The realized generalized force/moment is Bu, and the allocation error is $e_\tau(t) := Bu(t) - \tau_{\text{in}}(t)$. The quadratic allocation objective is $\sum_{i=1}^{8} \frac{1}{2} u_i^\top P_i u_i + q_i^\top u_i$ with

$$P_1 = \begin{bmatrix} 0.84 & 2.29 \\ 2.29 & 7.00 \end{bmatrix}, P_2 = \begin{bmatrix} 6.40 & 4.78 \\ 4.78 & 5.27 \end{bmatrix}, P_3 = \begin{bmatrix} 7.22 & 3.52 \\ 3.52 & 4.52 \end{bmatrix},$$
$$P_4 = \begin{bmatrix} 6.11 & 5.91 \\ 5.91 & 6.74 \end{bmatrix}, P_5 = \begin{bmatrix} 8.02 & 3.55 \\ 3.55 & 5.83 \end{bmatrix}, P_6 = \begin{bmatrix} 5.06 & 4.29 \\ 4.29 & 6.32 \end{bmatrix},$$
$$P_7 = \begin{bmatrix} 7.15 & 6.31 \\ 6.31 & 5.80 \end{bmatrix}, P_8 = \begin{bmatrix} 6.75 & 5.74 \\ 5.74 & 8.12 \end{bmatrix},$$
$$q_1^\top = \begin{bmatrix} -0.94 & -1.17 \end{bmatrix}, q_2^\top = \begin{bmatrix} -0.16 & -0.19 \end{bmatrix},$$
$$q_3^\top = \begin{bmatrix} -0.15 & -0.27 \end{bmatrix}, q_4^\top = \begin{bmatrix} -0.53 & 1.53 \end{bmatrix},$$
$$q_5^\top = \begin{bmatrix} 1.68 & -0.25 \end{bmatrix}, q_6^\top = \begin{bmatrix} -0.88 & -1.06 \end{bmatrix},$$
$$q_7^\top = \begin{bmatrix} -0.49 & 1.60 \end{bmatrix}, q_8^\top = \begin{bmatrix} -0.71 & 1.23 \end{bmatrix}. \tag{23}$$

We compare two main cases, where in the first scenario C&C sends demand shares to four pinned agents 4, 5, 6, and 7, with $\tau_i = \frac{1}{4}\tau_{\text{in}}$ for each $i \in \mathcal{P}$, and the network runs an internal model distributed optimizer with those injected shares. Critically, the shares are not recomputed after failures: if a pinned agent is removed, its share is lost, requiring centralized intervention that undermines the "resilient distributed" premise. In the second scenario, the

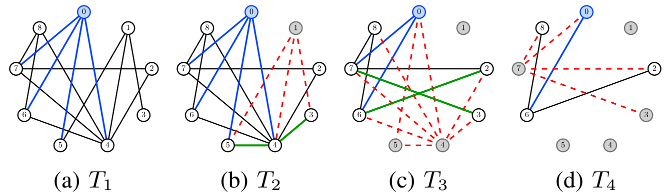

Fig. 1: Communication graphs under sequential agent removals and local reconfiguration.

C&C introduces the virtual agent 0, injecting the full demand τ_{in} through (4). Pinned agents serve only to maintain connectivity to the virtual agent, with no command partition across them.

Both cases use the same tuning to make the comparison structural rather than parametric; in particular, we have $\alpha = 100, \gamma = 1, \beta = 2000$. The communication graphs are shown in Fig. 1. The pinned set is the same in both cases, and the proposed method uses the same physical agent, together with the virtual node. The simulation runs with topology switching given by:

- $T_1 : t \in [0, 400)$: Nominal graph.
- $T_2 : t \in [400, 800)$: Agent 1 is removed.
- $T_3 : t \in [800, 1200)$: Agents 4 and 5 are also removed.
- $T_4 : t \in [1200, 1600)$: Agents 3 and 7 are also removed.

After each removal, the network management applies minimal edge rewiring. Hence, the surviving agents remain connected and at least one pinned agent maintains a path to the C&C or the virtual agent, consistent with the reconfiguration policy described in Section III. Fig. 2 reports the demand-tracking error $e_\tau(t) = Bu(t) - \tau_{\text{in}}(t)$ in surge, sway, and yaw. When the non-pinned actuator 1 is removed ($t \in [400, 800)$), both schemes recover and achieve zero steady-state allocation error, since the full demand information remains injected and the reduced allocation remains feasible. In contrast, when

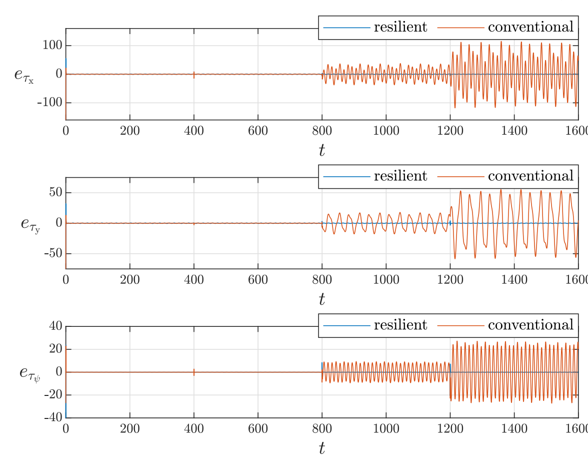

Fig. 2: Demand error $Bu - \tau$ in the face of agent failures.

the pinned agents 4 and 5 are removed in the second interval ($t \in [800, 1200)$), the baseline suffers a structural loss of injected command: the removed pinned agents no longer contribute their combined $\frac{1}{2}\tau_{\text{in}}$ share. Hence, the network converges to an allocation that matches only the remaining injected share of demand. This produces a nonzero steady-state allocation bias in e_τ. The proposed virtual-agent scheme continues to inject the full τ_{in} through the virtual node, so $e_\tau(t) \to 0$ again after the transient.

In the last interval ($t \in [1200, 1600)$), the pinned agent 7 and non-pinned agent 3 are removed. The removal of the pinned agent 7 eliminates another injected share, $\frac{1}{4}\tau_{\text{in}}$, from the conventional partitioned framework. In contrast, in the resilient framework, the presence of the last pinned agent allowed the system to generate the control demand with zero steady-state error.

The impact on closed-loop dynamic positioning is shown in Fig. 3. When the allocation layer exhibits a steady-state bias, the vessel cannot track $r(t)$ with zero steady-state error, and the output error $\eta(t) - r(t)$ inherits a persistent offset. Under the proposed scheme, since $Bu(t) \to \tau_{\text{in}}(t)$ is recovered, the supervisory output-regulation objective is restored and $\eta(t)$ tracks $r(t)$ asymptotically.

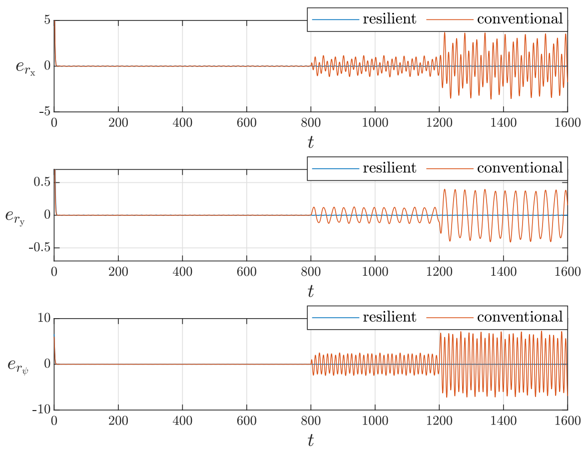

Fig. 3: The dynamic positioning output error.

These simulations illustrate the two design goals: the internal-model-based distributed allocator maintains accurate tracking under time-varying demands, and the virtual-agent injection removes the pinned-demand-partition failure mode by keeping the full command available after the removal of pinned agents. This recovery holds provided that connectivity to the virtual agent is preserved and the surviving actuator set remains feasible.

V. CONCLUSIONS

We formulate control allocation as a multi-agent optimization problem in which local agents coordinate via limited information exchange to satisfy a vector-valued control demand. To enhance the framework's resilience, we augment the network with a virtual agent at the supervisory level, enabling operation even when multiple agents are removed, provided that the remaining agents can physically meet demand. Moreover, to address time-varying demands, we embedded a p-copy internal model of demand dynamics into the distributed allocation loop via a gradient-based/implicit dynamic-average-tracking structure, improving steady-state accuracy and tracking of time-varying references. Simulation results in a marine vessel thrust allocation case study demonstrate accurate tracking after the removal of multiple actuators. The internal-model extension further eliminates steady-state tracking error for multi-frequency demand signals. Future work will establish closed-loop stability guarantees for the coupled plant–controller–allocator interconnection.

REFERENCES

[1] T. I. Fossen and T. A. Johansen, "A survey of control allocation methods for ships and underwater vehicles," in *2006 14th Mediterranean Conference on Control and Automation*, pp. 1–6, IEEE, 2006.

[2] T. A. Johansen and T. I. Fossen, "Control allocation—a survey," *Automatica*, vol. 49, no. 5, pp. 1087–1103, 2013.

[3] M. Bodson, "Evaluation of optimization methods for control allocation," tech. rep., American Institute of Aeronautics and Astronautics (AIAA), 2002. AIAA report/paper; PDF available from University of Utah.

[4] A. Casavola and E. Garone, "Fault-tolerant adaptive control allocation schemes for overactuated systems," *International Journal of Robust and Nonlinear Control*, 2010. Preprint PDF available online.

[5] T. A. Johansen, "Optimizing nonlinear control allocation," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 4, pp. 3435–3440, IEEE, 2004.

[6] J. Tjonnas and T. A. Johansen, "Optimizing adaptive control allocation with actuator dynamics," in *2007 46th IEEE Conference on Decision and Control*, pp. 3780–3785, IEEE, 2007.

[7] A. Cherukuri and J. Cortés, "Distributed generator coordination for initialization and anytime optimization in economic dispatch," *IEEE Transactions on Control of Network Systems*, vol. 2, no. 3, pp. 226–237, 2015.

[8] M. Luan, G. Wen, X. Ge, and Q.-L. Han, "Fully distributed resource allocation over unbalanced digraphs in prescribed time: A relaxed time-base generator approach," *Automatica*, vol. 177, p. 112313, 2025.

[9] 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.

[10] X. Lang and A. de Ruiter, "A control allocation scheme for spacecraft attitude stabilization based on distributed average consensus," *Aerospace Science and Technology*, vol. 106, p. 106173, 2020.

[11] S. Ghodsi, "Cyber-security of over-actuated cyber-physical systems: A study on dynamic positioning systems," Master's thesis, Concordia University, 2024.

[12] Q. Song, J. Cao, and W. Yu, "Second-order leader-following consensus of nonlinear multi-agent systems via pinning control," *Systems & Control Letters*, vol. 59, no. 9, pp. 553–562, 2010.

[13] S. S. Kia, B. Van Scoy, J. Cortes, R. A. Freeman, K. M. Lynch, and S. Martinez, "Tutorial on dynamic average consensus: The problem, its applications, and the algorithms," *IEEE Control Systems Magazine*, vol. 39, no. 3, pp. 40–72, 2019.

[14] J. Li and H. Su, "Implicit tracking-based distributed constraint-coupled optimization," *IEEE Transactions on Control of Network Systems*, vol. 10, no. 1, pp. 479–490, 2022.

[15] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, "Prediction-correction interior-point method for time-varying convex optimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1973–1986, 2017.

[16] N. Bastianello, R. Carli, and S. Zampieri, "Internal model-based online optimization," *IEEE Transactions on Automatic Control*, vol. 69, no. 1, pp. 689–696, 2023.

[17] G. Bianchin and B. Van Scoy, "The internal model principle of time-varying optimization," *IEEE Transactions on Automatic Control*, 2026.

[18] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[19] H. Cai, Y. Su, and J. Huang, "Cooperative control of multi-agent systems," *Cham: Springer-Verlag*, 2022.

[20] Y. Feng, F. Wang, Z. Liu, and Z. Chen, "Distributed network reconstruction strategy for multi-agent systems," *IFAC-PapersOnLine*, vol. 55, no. 3, pp. 119–124, 2022.

[21] R. A. Freeman, P. Yang, and K. M. Lynch, "Stability and convergence properties of dynamic average consensus estimators," in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 338–343, IEEE, 2006.

[22] Z. Ding, "Distributed time-varying optimization—an output regulation approach," *IEEE Transactions on Cybernetics*, vol. 54, no. 4, pp. 2155–2165, 2022.

[23] I.-A. Ihle, R. Skjetne, and T. I. Fossen, "Nonlinear formation control of marine craft with experimental results," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 1, pp. 680–685, IEEE, 2004.