

Distributed Reconfiguration of a Hybrid Shipboard Power System

Wanlu Zhu, *Member, IEEE*, Jian Shi, *Senior Member, IEEE*, Pengfei Zhi, *Member, IEEE*, Lei Fan, *Senior Member, IEEE*, and Gino Lim, *Member, IEEE*

Abstract—This paper presents a novel distributed reconfiguration strategy to enable the secure and reliable operation of the zonal shipboard power system (SPS). To adapt to the latest distributed control structure, the proposed strategy features two levels of reconfiguration: zonal reconfiguration and global reconfiguration. An extended hybrid model of SPS is first developed to accurately capture the interactions between the discrete events and the continuously evolving system dynamics involved in the reconfiguration process. The concept of zonal and global reconfigurability is then proposed along with the evaluation criteria to examine if the system can be reconfigured to return to steady-state operation. Provided that the SPS is reconfigurable, executable algorithms are proposed to determine the optimal sequence of operation events for both zonal and global reconfiguration. To evaluate the performance of the proposed strategy, four case studies are presented in which a four-zone SPS is faced with multiple random faults and requires reconfiguration. Simulation results demonstrate that the proposed reconfiguration strategy outperformed previous algorithms in examining the reconfigurability of the system and determining the optimal reconfiguration solution.

Index Terms—Reconfiguration; Dynamic Response; Distributed Control; Shipboard Power System

NOMENCLATURE

Indices

i	Normal operation configuration, $i = 1, \dots, N$
j	Faulty configuration, $j = N + 1, \dots, N + M$
k	Subsystem, $k = 1, \dots, K$
l_k	Configuration of subsystem S_k , $l = 1, \dots, (N + M)_k$
p_k	Reconfiguration event of subsystem S_k , $p_k = 1, \dots, P_k$

Parameters

\mathcal{C}	A distributed configuration of SPS
\mathcal{C}_0	The initial distributed configuration
\mathcal{H}	Set of all possible configurations in subsystem
\mathcal{R}	A global reconfiguration scenario
\mathcal{C}	Set of all possible distributed configurations
\mathcal{R}	Set of all possible global reconfigurations
\mathcal{S}	SPS
Φ/S	The controlled subsystem
τ	Time instant
ε	Empty configuration of subsystem not involved in the global reconfiguration
FT	Set of all possible fault events

This work was supported by the National Natural Science Foundation of China under Grant No.51807198.

Wanlu Zhu and Pengfei Zhi are with the School of Electronics and Information, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu, China. E-mail: (guel.z@hotmail.com, zhipengfei@just.edu.cn).

Jian Shi and Lei Fan are with the Department of Engineering Technology, University of Houston, Houston, Texas, USA. Their emails are jshi14@uh.edu and lfan8@central.uh.edu.

Gino Lim is with the Department of Industrial Engineering, University of Houston, Houston, Texas, USA. His e-mail is: ginolim@uh.edu.

H_0	The initial configuration
I_{ab}^{max}	Maximum current limit for each branch
$P_{Gen}^{min/max}$	Minimum/Maximum power capacity can be provided by each generator
q	Discrete state
RE	Set of all events leading the subsystem back to normal operation configuration
$V_a^{min/max}$	Minimum/Maximum voltage limit for each node
Variables	
μ_τ	Control input at time τ
π_{opt}	Optimal control event sequence at τ
Σ_j	Set of all events in the j -th faulty configuration
$\sigma_s(\tau_d)$	Discrete control event at τ_d
$\Sigma_{uc,j}$	Set of all uncontrollable events in the j -th faulty configuration
τ_d	Time instant at d -th discrete event occurs
φ_j	State-based controller in faulty configuration H_j
H_i	The i -th normal operation configuration
H_j	The j -th faulty configuration
H_{k,l_k}	The l_k -th configuration of S_k
I_{ab}	Intra-zone current flow from component a to b
$L_{reachable}(q_{le})$	Control event sequences set associated with q_{le}
$P_{S_k, S_{k-1}}$	Power delivered from subsystem S_k to S_{k-1}
Q_i	Discrete states set in the i -th normal operation configuration
Q_j	Discrete states set in the j -th faulty configuration
$Q_{0,j}$	Initial states set in faulty configuration H_j
$Q_{il,j}^\uparrow$	Set of all states that uncontrollably enter $Q_{il,j}$
$Q_{re,j}^\uparrow$	Set of all states that can reach $Q_{re,j}$
$Q_{il,j}$	Illegal states set in faulty configuration H_j
$Q_{le,j}$	Legal states set in faulty configuration H_j
$Q_{reachable}$	Set of all states in $Q_{le,j}$ can be reached from q_0
r_{k,p_k}	The p_k -th reconfiguration event of S_k
S_k	The k -th subsystem
T_j	Set of all transitions in j -th faulty configuration
$u(\tau)$	Continuous control input at time τ
V_a	Voltage at component a in the subsystem
ft_j^i	Fault event leading subsystem from H_i to H_j
re_r^l	Event leading the subsystem back to normal operation configuration H_r

I. INTRODUCTION

MEDIUM-Voltage DC (MVDC) Shipboard Power Systems (SPS) is the emergent technology for future all-electric ships (AES), developed to meet the increasing onboard power demand. As an islanded micro-grid with over 100 MW of combined loads, SPS is playing a critical role in supplying energy to all electrical equipment onboard with widely varying characteristics, ranging from continuous loads such as propulsion motors, to intermittent high-power loads like weaponry and radar systems

[1]. While the power-electronic based DC distribution system offers advantages such as weight and size reduction, better power density, and improved reliability, it is also faced with a series of significant challenges. Compared to the terrestrial distribution system, the closed electrical and physical proximity of a ship, the limited generation capacity and inertia, and the stringent operation requirements have made the SPS more fragile and prone to faults and failures [2]. Therefore, an effective reconfiguration strategy needs to be deployed to handle system damages and faults, transfer high-quality power to vital loads, and maintain the ship in an operational state to fulfill its mission in both normal and adverse conditions.

The reconfiguration of an SPS can minimize the effects of disturbances and maintain system performance by rerouting the electric power [3]. It does not only isolate the compromised parts and alter the distribution network topology after disturbances, but also provides a solution for system power balance and overall performance optimization [4]. In order to achieve these objectives, the reconfiguration needs to coordinate the discrete events involved in this reconfiguration process, such as the switching on and off of circuit breakers and bus transfers, with the continuously evolving system dynamics. This suggests that the reconfiguration problem should be studied from the perspective of a *hybrid* system that accurately captures the complex interactions between the discrete and continuous aspects of an SPS occurring within different time scales.

Moreover, with the development of the Next Generation Integrated Power System (NGIPS), the reconfiguration becomes more complicated when the zonal distribution structure is adopted to save space and weight and improve reliability. In the reference architecture of the NGIPS outlined in [5][6], the whole SPS can be divided into multiple zones. Each zone (i.e. subsystem) consists of modules such as power generation modules (PGM), power conversion modules (PCM), propulsion modules (PM), special loads (pulsed load and radars), etc. While each zone involved in the SPS can be separately considered as a system that is both administratively and operationally self-governing, they share both physical interconnectivity (e.g., shared energy resources and connections) and functional interconnectivity (e.g., generation and load management). Thus, the cooperation among the zones is necessary to accomplish the goal of seamlessly transitioning the required power to loads collaboratively. As such, a zonal SPS should be controlled in a distributed fashion, which suggests that the reconfiguration needs to be performed distributedly as well.

While the reconfiguration of SPS has been extensively studied in the literature with objectives such as enhancing stability margins [7], increasing power delivery [8], and minimizing the number of switching operations [9], most existing reconfiguration algorithms are designed in a centralized manner based on methods such as graph theory [10] and heuristic/intelligent approaches (e.g. Particle Swarm Optimization (PSO) [8], Genetic Algorithm (GA) [11], and Q-learning [12]). However, due to their centralized nature, these algorithms may not be directly applicable to our problem settings. For distributed reconfiguration, notable literature mainly focused on performing reconfiguration using the multi-agent system (MAS) approach [13][14][15][16] where shipboard components were represented as agents, and the reconfiguration results were obtained through the interactions among these agents according to a pre-specified set of rules. However, as pointed out in [17], existing MAS-based approaches are primarily designed to

handle relatively simple scenarios. When multiple random faults are present simultaneously, which are considered common in the SPS operation, the applicability and effectiveness of the MAS-based approaches have not been fully evaluated yet.

Another major issue with the existing literature is that the system behaviors have been greatly simplified in the reconfiguration algorithm design. As aforementioned, a hybrid model can best describe the reconfiguration process of an SPS. While research efforts have been made to study SPS as a hybrid system [18][19][20][21][22], the focus was prioritized to studying operations such as load shedding [20] and supervisory control [23]. A hybrid supervisory framework was developed in [21] to obtain the complete system representation and analyze the optimal recovery actions after the system lost a generator. However, the hybrid system dynamics was formulated by state equations with integer variables in this work, making it difficult to be solved on a large scale. Existing literature made another common assumption that, after system disturbance(s), the reconfiguration always needed to be performed as an attempt to maintain system operational and return the system to steady-state. This assumption can be problematic as there might be occasions with little or no chance for restoration. If the outcome of the reconfiguration can be estimated beforehand under these circumstances, time and other valuable resources can be saved through bypassing the reconfiguration step and pursuing alternative options. Readers are referred to [17] for a comprehensive review of the existing SPS reconfiguration efforts.

Realizing the limitations of the existing research efforts in this area, this paper investigates the distributed reconfiguration problem of a zonal SPS with hybrid dynamics. To be suitable for the zonal structure, we propose a novel two-level reconfiguration strategy that consists of a set of *zonal* (i.e., subsystem-level) reconfiguration and a set of *global* (i.e., ship-wide) reconfiguration. Hence, zonal reconfigurations can be executed within the boundary of a zone to adjust its zonal configuration to meet the current mission requirements or react to disturbances, while global reconfiguration involves two or more zones to address the potential trade-offs and conflicts among the zones and reach a ship-wide optimal configuration within an execution period. To adequately capture the complex characteristics of the SPS during the reconfiguration process, an extended hybrid model is developed. Furthermore, a novel concept, *reconfigurability*, is defined for both the zonal and global reconfiguration processes. Reconfigurability measures a system's capability to remain in steady-state under faulty conditions as well as its ability to return to a normal operation configuration. Their evaluation criteria and implementation algorithms are then derived, respectively. Case studies are performed to evaluate the effectiveness of the proposed reconfiguration strategy.

Overall, the contributions of this paper include:

- 1) A novel distributed reconfiguration strategy is proposed to adequately and simultaneously capture the hybrid and distributed nature of the zonal SPS in a comprehensive and computationally efficient way.
- 2) An extended hybrid model of SPS is developed to accurately describe the evolution of the complex dynamics of the SPS during the reconfiguration process. This model is tailored to be incorporated into the proposed reconfiguration method.
- 3) A novel concept, reconfigurability, is proposed for the zonal and global reconfiguration, respectively. Reconfigurability provides theoretical criteria to estimate if the SPS can

be restored through reconfiguration *a priori*. For systems that are deemed reconfigurable, it offers the guidance on determining the optimal sequence of control actions.

- 4) The effectiveness of the proposed reconfiguration strategy is verified against the performance of two commonly adopted reconfiguration strategies in the literature, including a MAS-based reconfiguration and a GA-based reconfiguration.

The rest of this paper is organized as follows: Section II describes the extended hybrid model of SPS. Section III and IV discuss the operation principles and execution procedures of the subsystem reconfiguration and global reconfiguration, respectively. Four case studies are conducted, and the results are presented in Section V to validate the proposed reconfiguration strategy. Finally, conclusions are drawn in Section VI.

II. HYBRID SPS MODEL FOR RECONFIGURATION

In this section, the distributed zonal control structure of an MVDC SPS is briefly reviewed. The extended hybrid model of SPS is then developed to be incorporated into the distributed reconfiguration strategy design.

A. Distributed Zonal Control Structure of SPS

To set the stage for the rest of this paper, a generic two-layer distributed control architecture is considered: global layer and subsystem layer. As shown in Fig. 1, each zone has its own local controller on the subsystem layer that performs in-zone control. More specifically, a zonal controller handles internal fault protections, basic energy and power management (such as load shedding and zonal reconfiguration), and deals with perturbations within its boundary. To ensure the effective collaboration among multiple distributed zones, the coordinator on the global layer communicates with each zonal controller to determine a converged ship-wide, multi-zone control solution. Therefore, distributed control decisions can collectively determine the energy allocations within the system as well as the global point of interest. This structure ensures that the separately optimized zones can be reconciled to maintain the efficiency and reliability of the entire SPS.

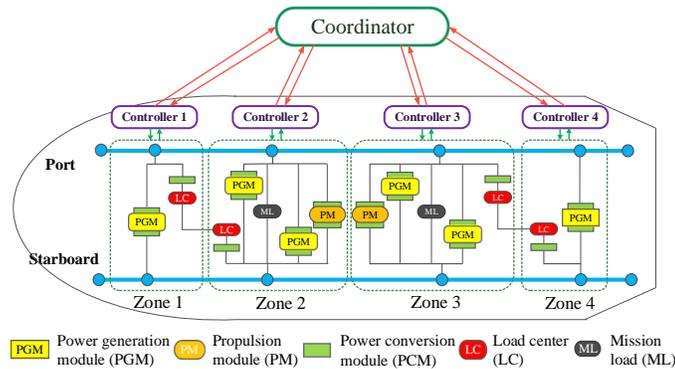


Fig. 1. The two-layer distributed control structure of a zonal SPS

To adapt to the control structure set forth above, we assume that zonal reconfigurations can only be executed directly by zonal controllers, and global reconfigurations can only be executed by the coordinator. While local controllers are permitted to send requests to the coordinator demanding a certain global reconfiguration operation, the coordinator decides whether to respond and execute the received requests. A local controller does not participate in any global reconfiguration operation unless it receives commands from the coordinator that directs it to do so.

B. Extended Hybrid Model of Subsystems

To formulate the reconfiguration problem of an SPS, an extended hybrid automaton model is first developed in this section to describe the complex dynamics of SPS subsystems.

A general hybrid automaton model can be described as:

$$H = (Q, X, U, Init, f, \Sigma, EG, T, R) \quad (1)$$

where Q denotes the finite set of all discrete states, X denotes the set of continuous states, $Q \cup X$ denotes the state space, U is the continuous control input set, and $Init \subseteq Q \times X \times U$ denotes the initial set. $f : Q \times X \times U \rightarrow X$ is used to illustrate the continuous dynamics in state $q \in Q$. $\Sigma = \Sigma_c \cup \Sigma_u$ is the discrete event sets, while Σ_c denotes the set of controllable events, and Σ_u denotes the set of uncontrollable events. Σ^* is the set of all finite-length sequences of events. A subset of Σ^* is called a language, and all the languages of H are denoted as $L(H)$. $EG : X \times U \rightarrow \Sigma$ is the event generation function. $T : \Sigma \times Q \rightarrow 2^Q$ denotes the transitions between discrete states. $R : Q \times X \times U \rightarrow 2^{X \times U}$ is the reset map, which also represents the control event generation.

Under the common assumption that a fault can be modeled as a discrete event [24], the operation of a subsystem can be divided into N normal operation configurations and M faulty configurations based on the general hybrid automaton (1). The normal operation configurations can be formulated as:

$$H_i = (Q_i, X_i, U_i, Init_i, f_i, \Sigma_i, EG_i, T_i, R_i) \quad (2)$$

where H_i denotes the i -th normal operation configuration, $i = 1, 2, \dots, N$.

Likewise, the faulty configuration can be formulated as:

$$H_j = (Q_j, X_j, U_j, f_j, \Sigma_j, EG_j, T_j, R_j) \quad (3)$$

where H_j denotes the j -th faulty configuration, $j = N + 1, N + 2, \dots, N + M$. It should be noted that a faulty configuration does not have an initial state because it is dependent on the state of the normal operation configuration when the fault occurs.

For a subsystem entering a faulty configuration from a normal operation following one or more faults and let ft_j^i be the fault event, FT can be defined as the set of all possible fault events in the form of:

$$FT = \{ft_j^i : i = 1, 2, \dots, N; j = N + 1, \dots, N + M\} \quad (4)$$

Let us define the state transition when fault ft_j^i occurs:

$$Q_i = q_{1,i}, q_{2,i}, \dots, q_{|Q_i|,i} \quad (5)$$

$$Q_j = q_{1,j}, q_{2,j}, \dots, q_{|Q_j|,j}$$

Then, there exists the mapping:

$$ft_j^i : Q_i \rightarrow Q_j \quad (6)$$

and $ft_j^i(q_{l,i}) = q_{p,j}$ denotes that: following fault ft_j^i that occurs at state $q_{l,i}$, the subsystem would enter faulty configuration H_j and the next state would be $q_{p,j}$.

In this way, the operation that leads a subsystem to enter the faulty configuration can be described as follows: at first, the subsystem was in normal configuration H_i and its initial state was $q_{0,i}$. After a certain sequence of events, the subsystem was in a new state $q_{l,i}$. A fault event ft_j^i then occurred, and this caused the subsystem to enter faulty configuration H_j where its state was determined by $q_{p,j} = ft_j^i(q_{l,i})$.

After the subsystem enters faulty configuration H_j , there may exist multiple events that can lead the subsystem back to normal operation configuration H_r . Since these events cannot occur at all of the states contained in H_j , $Q_{re,j}$ is used to denote the set of all states in which these recovery events can occur within H_j .

Let us define the event that leads the subsystem from H_j to H_r as:

$$re_r^j : Q_j \rightarrow Q_r \quad (7)$$

Similar to (5), $re_r^j(q_{p,j}) = q_{o,r}$ indicates that event re_r^j occurs at state $q_{p,j} \in Q_{re,j}$ and that it will return the subsystem to normal operation configuration H_r with the next state being $q_{o,r}$.

The set of all events that lead the subsystem back to normal operation configuration can be described as:

$$RE = \{re_r^j : r = 1, 2, \dots, N; j = N + 1, \dots, N + M\} \quad (8)$$

Unlike fault events, events in the set RE can be actively enforced by the controllers.

Transitions between fault operation configurations, which involve two or more faults, are similar as above. Therefore they are not included here for the sake of brevity. Similarly, transitions between normal operation configurations are considered normal and are thus not included here.

Then a subsystem, S , can be represented as:

$$S = (\mathcal{H}, FT, RE, H_0) \quad (9)$$

where \mathcal{H} denotes the set of all possible configurations in S , and H_0 denotes the initial configuration that S started from.

C. Ship-wide Model

Assume that an SPS consists of K subsystems, and a subsystem $S_k (k = 1, 2, \dots, K)$ has $l_k (l_k = 0, 1, \dots, (N + M)_k)$ configurations, and $p_k (p_k = 0, 1, \dots, P_k)$ reconfiguration events. We can use \mathcal{C} to present a distributed configuration of the SPS as:

$$\mathcal{C} = (H_{1,l_1}, H_{2,l_2}, \dots, H_{K,l_K}) \quad (10)$$

where H_{k,l_k} denotes the l_k -th configuration of S_k . The set of all possible distributed configurations can be denoted by \mathcal{C} . Note that since each subsystem S_k contains a finite set of configurations \mathcal{H} , and each configuration can be modeled as a finite automaton as described in (1), it is always feasible to construct a finite distributed configuration set of the SPS \mathcal{C} .

A global reconfiguration scenario can be defined as:

$$\mathcal{R} = (r_{1,p_1}, r_{2,p_2}, \dots, r_{K,p_K}) \quad (11)$$

where r_{k,p_k} is the p_k -th reconfiguration event of S_k . It should be noted that r_{k,p_k} may not exist since not all subsystems are necessarily involved in a particular global reconfiguration request.

Define the set of all possible global reconfigurations as \mathcal{R} , there exists the mapping:

$$\mathcal{R} : \mathcal{C} \rightarrow \mathcal{C} \quad (12)$$

Then the SPS, \mathcal{S} , can be represented as:

$$\mathcal{S} = (\mathcal{C}, \mathcal{R}, \mathcal{C}_0) \quad (13)$$

where \mathcal{C}_0 denotes the initial distributed configuration from which \mathcal{S} started.

III. SUBSYSTEM RECONFIGURATION

Based on the hybrid SPS model developed in Section II, the reconfiguration process can be developed. In this section, we focus on addressing the design of the subsystem reconfiguration strategy.

A. Subsystem Reconfigurability

Disturbances and contingencies constantly change subsystem dynamics, and may lead a subsystem from normal operation configurations to faulty configurations. While the subsystem remains safe in normal operation configurations, it may become unsafe when the system enters a faulty configuration. Here we define the subsystem reconfigurability as the ability that: following one or

more disturbances/faults, a subsystem can 1) stay in a safe state under a faulty configuration, or 2) have the chance to return to a normal operation configuration. Note that subsystem performance is inevitably impacted under faulty configurations. Therefore, returning the subsystem to normal operation configuration always receives a higher priority in the reconfiguration process.

Based on this description, the states in a faulty configuration can be divided into the set of legal states $Q_{le,j}$ and the set of illegal states $Q_{il,j}$. As a general rule of thumb, this division is performed based on a set of pre-specified, application-specific logical and functional constraints. Note that in this manuscript, *safe* states refer to the states in which the SPS could remain in stable operation, whereas *unsafe* states refer to the states in which the SPS cannot remain stable and would uncontrollably enter other states or fail. Meanwhile, we define *illegal* states based on a set of general SPS operation requirements and arbitrary specifications. Therefore, the difference between an illegal state and an unsafe state is that: an illegal state may not be an unsafe state, while all legal states must be safe states. For example, as will be discussed in the case studies later, we define that it is illegal yet safe for the vital loads to be offline while the normal loads are online.

Hence, following a fault, control actions need to be performed to prevent it from entering illegal states in order to ensure subsystem safety under faulty configuration. Furthermore, the subsystem should be enforced by the controller to return to a normal operation configuration when it enters $Q_{re,j}$.

Considering all the states are observable, we can define a state-based controller as:

$$\varphi_j : Q_j \rightarrow 2^{\Sigma_j} \quad (14)$$

where $\varphi_j(q)$ defines the set of events that can occur at state q . For $\forall q \notin Q_j$, $\varphi_j(q) = \emptyset$.

Now the controlled subsystem can be formulated as: Φ/S , where $\Phi := \varphi_{N+1} \wedge \varphi_{N+2} \wedge \dots \wedge \varphi_{N+M}$ and $\Phi(q) = \varphi_{N+1}(q) \cup \varphi_{N+2}(q) \cup \dots \cup \varphi_{N+M}(q)$.

Denoting all the reachable states of the controlled subsystem as $R(\Phi/S)$, the definition of subsystem reconfigurability can be described as:

Definition 1: A subsystem S is reconfigurable if for any faulty configuration $H_j (j = 1, 2, \dots, m)$, whenever the subsystem enters H_j , there exists a control φ_j such that

- 1) The subsystem will never enter illegal states, which means $R(\Phi/S) \cap Q_{il,j} = \emptyset$;
- 2) When 1) is satisfied, the subsystem has the chance to return to a normal operation configuration, which means $R(\Phi/S) \cap Q_{re,j} \neq \emptyset$.

B. Subsystem Reconfigurability Criteria

When a subsystem enters faulty configuration H_j from normal operation configuration H_i following a fault, we define the set of initial states $Q_{0,j}$ in faulty configuration H_j as:

$$Q_{0,j} = \{q \in Q_j : (\exists q' \notin Q_j) ft(q') = q\} \quad (15)$$

Define the set of all states that uncontrollably enter the illegal states set $Q_{il,j}$ as:

$$Q_{il,j}^\uparrow = \{q \in Q_j : (\exists s \in \Sigma_{uc,j}^*) T_j(q, s) \in Q_{il,j}\} \quad (16)$$

Then define the set of all states that can reach $Q_{re,j}$ as:

$$Q_{re,j}^\uparrow = \{q \in Q_j : (\exists s \in \Sigma_j^*) T_j(q, s) \in Q_{re,j}\} \quad (17)$$

The following theorem provides the guideline on how to check the subsystem reconfigurability under full observation.

Theorem 1: Under full observation, subsystem S is reconfigurable if and only if

- 1) $Q_{0,j} \cap Q_{il,j}^\dagger = \emptyset$.
- 2) When 1) is satisfied, $Q_{0,j} \cap Q_{re,j}^\dagger \neq \emptyset$.

If the subsystem is reconfigurable, there exists the control:

- 1) $\Phi(q) = \begin{cases} \{\sigma \in \Sigma_j : T_j(q, \sigma) \in Q_j - Q_{il,j}^\dagger\} & q \in Q_j - Q_{il,j}^\dagger \\ \emptyset & \text{other} \end{cases}$
- 2) $\Phi(q) = \{\sigma \in \Sigma_j^* : T_j(q, \sigma) \in Q_{re,j}\} \quad q \in Q_{re,j}^\dagger$

The proof of this theorem is provided in the Appendix.

C. Hybrid Subsystem Reconfiguration Algorithm

While the previous discussion on subsystem reconfigurability outlines the controller $\Phi(q)$ which specifies all the possible control events in state q , it is still necessary to find an optimal event sequence of such events to achieve optimal subsystem performance under a faulty configuration. While the performance of a subsystem can be measured in different ways, maximizing the power delivered to loads is used as a representative case of performance measurement for the subsystem reconfiguration in this paper.

The continuous dynamics of subsystem S_k in state $q \in Q$ can be discretized and modeled in the form of:

$$x(\tau + 1) = f_q(x(\tau), u(\tau)) \quad (18)$$

where τ is the time instant. Use τ_d to denote the time instant at which the d -th discrete event occurs, and τ_d is an integer multiple of τ . Define the control input at time τ as $\mu_\tau : Q \times X \rightarrow 2^{\Sigma \times U}$. If no discrete control event takes place at time τ , the system dynamics is determined by the continuous control input $u(\tau)$. Otherwise, it is determined by both the discrete control event and the continuous control input as:

$$\mu_\tau(x(\tau), q(\tau)) = \begin{cases} (\sigma_s(\tau_d), u(\tau)) & \tau \in \{\tau_d\} \\ (\emptyset, u(\tau)) & \tau \notin \{\tau_d\} \end{cases} \quad (19)$$

where $\sigma_s(\tau_d) \in \Phi(q(\tau))$ is the discrete control event at τ_d .

We formulate the subsystem performance as:

$$J_\pi = \sum P_{load}(x(\tau), \mu_\tau(x(\tau), q(\tau))) \quad (20)$$

Therefore, the subsystem reconfiguration problem can be defined as follows. Given an initial condition (q_0, x_0, u_0) , find the optimal control event sequence $\pi_{opt} \in \Sigma_{S_k}^*$ as:

$$\pi_{opt} = \text{argmax} J_\pi \quad (21)$$

subject to the following intra- and inter-zone constraints:

Intra-zone constraints:

$$I_{ab} \leq I_{ab}^{max} \quad (22)$$

$$V_a^{min} \leq V_a \leq V_a^{max} \quad (23)$$

$$P_{gen}^{min} \leq P_{gen} \leq P_{gen}^{max} \quad (24)$$

Inter-zone constraints:

$$\sum P_{gen, S_k} + P_{S_{k+1}, S_k} = \sum P_{load, S_k} + P_{S_k, S_{k-1}} \quad (25)$$

where V_a is the voltage at component a in the subsystem, and I_{ab} is the intra-zone current flow from component a to component b . $P_{S_k, S_{k-1}}$ denotes the power delivered from subsystem S_k to S_{k-1} . (22) captures the line current constraint between component a and b . (23) represents the node voltage constraint. (24) denotes the operation limit for each generator. (25) assures power balance within subsystem S_k . It should be noted starboard and port bus need to be calculated separately.

Algorithm 1 Hybrid Subsystem Reconfiguration Algorithm

```

1: Input:  $Q_{reachable}, (q_0, x_0, u_0), \Sigma_{c, S_{k-1}}, \Sigma_{c, S_{k+1}}$ 
2: for all  $q_{re}$  in  $Q_{reachable}$  do
3:   for all  $s_{re}$  in  $L_{reachable}(q_{re})$  do
4:     Apply the discrete control events in  $s_{re}$  to the system and
     execute the simulation.
5:     if intra-zone constraints (22)(23) are satisfied then
6:       calculate the objective function  $J_\pi$ 
7:     end if
8:   end for
9: end for
10: Order and label  $J_\pi$  such that  $J_\pi^1 \geq J_\pi^2 \dots \geq J_\pi^n$ 
11: for  $i=1$  to  $i=n$  do
12:   if inter-zone constraint (26) is satisfied then
13:     OUTPUT:  $\pi_{opt}=s_{re}$  which is associated with  $J_\pi^i$ 
14:   END
15:   else
16:     for all  $e_g$  in  $\Sigma_{c, S_{k-1}} \cup \Sigma_{c, S_{k+1}}$  do
17:       if constraint (26) is satisfied then
18:         OUTPUT:  $e_g$  and  $\pi_{opt}=s_{re}$  which is associated with  $J_\pi^i$ 
19:       END
20:     end if
21:   end for
22: end if
23: end for

```

From (24) and (25), we can derive the following constraints for the inter-zone power transfer $P_{S_k, S_{k-1}}$ between subsystem S_k and S_{k-1} :

$$P_{S_k, S_{k-1}}^{min} < P_{S_k, S_{k-1}} < P_{S_k, S_{k-1}}^{max} \quad (26)$$

and

$$P_{S_k, S_{k-1}}^{min} = \max \left\{ \sum_{v=S_k}^{S_K} \left(\sum P_{Gen, v}^{min} - \sum P_{load, v} \right), \sum_{v=S_1}^{S_{k-1}} \left(\sum P_{load, v} - \sum P_{Gen, v}^{max} \right) \right\} \quad (27)$$

$$P_{S_k, S_{k-1}}^{max} = \min \left\{ \sum_{v=S_k}^{S_K} \left(\sum P_{Gen, v}^{max} - \sum P_{load, v} \right), \sum_{v=S_1}^{S_{k-1}} \left(\sum P_{load, v} - \sum P_{Gen, v}^{min} \right) \right\} \quad (28)$$

where $\sum P_{Gen, v}^{min/max}$ denote the minimum/maximum power capacity can be provided by zone v , and $\sum P_{load, v}$ denotes the load demand in zone v , $v \in [S_1, S_2, \dots, S_K]$. The derivation details of (26)-(28) can be found in the Appendix.

Similarly, the constraint for the inter-zone power transfer $P_{S_k, S_{k+1}}$ between subsystem S_k and S_{k+1} can be derived as well. The detailed formulation is not included here for the sake of brevity.

To solve (20), we first determine the reachable states of the subsystem. Define the set of all the states in $Q_{le, j}$ that can be reached from the initial state q_0 as:

$$Q_{reachable} = \{q \in Q_{le, j} : (\exists s \in \Gamma(q_0)) T_j(q_0, s) = q\} \quad (29)$$

Define the set of all the control event sequences associated with $q_{le} \in Q_{reachable}$ as:

$$L_{reachable}(q_{le}) = \{s \in \Gamma(q_0) : T_j(q_0, s) = q_{le}\} \quad (30)$$

The process to find the optimal subsystem reconfiguration events can be divided into two steps. In the first step, the subsystem reconfiguration algorithm calculates all the objective function (20) for system states that satisfy the intra-zone constraints (22)-(24). The second step checks the inter-zone constraint

(26) for the obtained J_π (i.e., $J_\pi^1, J_\pi^2, \dots, J_\pi^n$) in descending order. If constraint (26) is satisfied, then the subsystem controller requests the reconfiguration event s_{re} that is associated with J_π . Otherwise, the subsystem controller finds an event e_g from the adjacent zone(s) to meet the constraint (26) so it can request the zonal reconfiguration event s_{re} . The event e_g is submitted as the global reconfiguration event. If the subsystem is unable to find an event from the adjacent zone(s) to meet (26), the next J_π will be considered. The subsystem reconfiguration algorithm is described in Algorithm 1.

IV. GLOBAL RECONFIGURATION

Because all of the involved subsystems are seeking to enter their targeted configurations, the global reconfiguration is essentially a coordination process that deals with the individual requests submitted by subsystems and aims to produce a converged solution.

Suppose that a global reconfiguration $\mathcal{R} = (r_1, r_2, \dots, r_K)$ is applied to an SPS that is initially in distributed configuration $C_x = (H_1^x, H_2^x, \dots, H_K^x)$, and the system enters configuration C_y . As r_k ($k = 1, 2, \dots, K$) may not exist, the associated subsystem configuration H_k will not change after the global reconfiguration. In this way, we can get the reduced C'_y where ε is used to present the configuration of subsystems that are not involved in the global reconfiguration \mathcal{R} . The set of all reduced C'_y can be denoted by \mathcal{C}' .

Then we can define global reconfigurability as:

Definition 2: An SPS is global reconfigurable if for any subsystem S_k ($k = 1, 2, \dots, K$) that requests a global reconfiguration \mathcal{R}_k to enter the target configuration H_k^{aim} in one execution period, we can find at least one global reconfiguration strategy $\mathcal{R} = (r_1, r_2, \dots, r_K)$ to satisfy all the requests. That is, for $k = 1, 2, \dots, K$

$$\begin{aligned} (\forall H_k^{aim} \in \mathcal{H}_k)(\exists C'_y \in \mathcal{C}') H_k^y = H_k^{aim} \vee H_k^y = \varepsilon \\ \Rightarrow (\exists \mathcal{R} \subset \mathcal{R}) \mathcal{R}(C_x) = C_y \end{aligned} \quad (31)$$

Under the distributed control structure as depicted in Fig.1, the coordinator must always provide a converged coordination solution after receiving all the concurrent reconfiguration requests from different subsystems. This coordination solution (i.e., global reconfiguration strategy) should satisfy all the requests while adjusting as few subsystems as possible.

First, we discuss the *restricted* global reconfiguration algorithm, which entirely fulfills the requirements above.

Algorithm 2 Global Reconfiguration Algorithm (Restricted)

- 1: **Input:** all the target configurations H_k^{aim} , \mathcal{C}'
- 2: **for** all C'_y in \mathcal{C}' **do**
- 3: **if** $H_k = H_k^{aim}$ (H_k is the configuration of subsystem S_k in C'_y) **or** $H_k = \varepsilon$ **then**
- 4: save C'_y into set C_{temp}
- 5: **end if**
- 6: **end for**
- 7: **if** $C_{temp} = \emptyset$ **then**
- 8: **OUTPUT:** No optimal global reconfiguration
- 9: **END**
- 10: **else**
- 11: **for** all elements in C_{temp} **do**
- 12: choose C_{min} that impacts the least amount of subsystems which are not requesting global configurations
- 13: **end for**
- 14: **OUTPUT:** C_{min}
- 15: **end if**

Since all of the subsystems are aiming to operate in their own best interests, there may not always exist a global reconfiguration solution that is capable of satisfying all the requests. This necessitates another *less restricted* global reconfiguration algorithm. To do so, global reconfiguration requests are first classified into two categories: critical requests and non-critical requests, according to the nature of the request. Moreover, the subsystems can be divided into vital and non-vital subsystems. Subsequently, configuration set \mathcal{H}_k of S_k can be divided into four sets: \mathcal{H}_k^{v-v} (all the target configurations of critical requests from vital subsystems), \mathcal{H}_k^{nv-v} (all the target configurations of critical requests from non-vital subsystems), \mathcal{H}_k^{v-nv} (all the target configurations of non-critical requests from vital subsystems) and \mathcal{H}_k^{nv-nv} (all the target configurations of non-critical requests from non-vital subsystems).

Then the less restricted global reconfiguration algorithm can be derived as illustrated in Algorithm 3 in the Appendix. Algorithm 3 indicates that the coordinator can always find a converged global reconfiguration solution although it may not always satisfy all of the subsystem requests.

V. CASE STUDIES

In this section, we will demonstrate how the proposed distributed reconfiguration strategy can be adopted to systematically enhance the ship reconfiguration following multiple faults.

The reference MVDC model presented in [5] is modified and used in the case studies. This four-zone SPS includes two main generators (MTGs), three auxiliary generators (ATGs), and one backup diesel generator (BDG). They can power port and starboard bus simultaneously and independently. Mission loads and Propulsion Modules are powered through both port and starboard bus equally. Each zone has a PCM to supply the zonal loads (ZL), which are located in the integrated power node center (IPNC) and AC Load Center (ACLC). Zonal loads can be further classified into vital loads, semi-vital (SV) loads and non-vital (NV) loads based on their operation priorities and interruptibilities. In normal operations, the loads are fed in-zone. When a disturbance occurs that prevent the loads from being supplied in-zone, the loads can be fed by the adjacent zone through reconfiguration if possible.

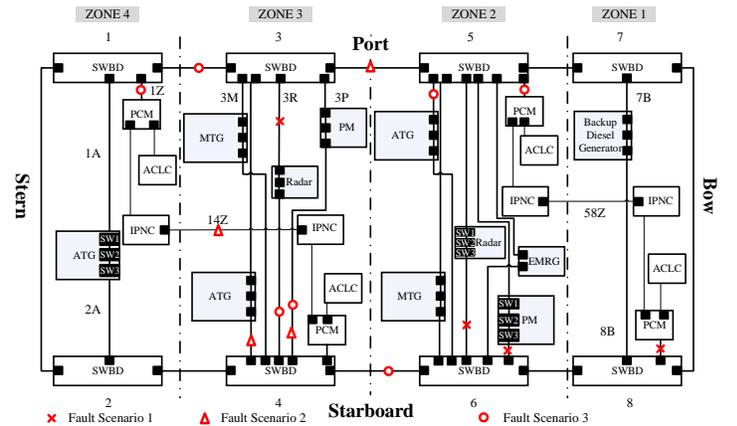


Fig. 2. Schematic diagram of the four-zone SPS

Table I displays the component specifications of the SPS under study, whose detailed parameters can be found in [25]. The initial condition of the system is that: all of the generators are online with the exception of the BDG, PMs are working at their rated power; and all of the zonal loads are fed in-zone with both radars online and the pulse load offline. All of the generators contribute

TABLE I
COMPONENT SPECIFICATIONS OF THE SPS

Component	Zone 1	Zone 2	Zone 3	Zone 4
PGM	BDG	ATG 4MW	ATG 4MW	ATG 4MW
	0.55MW	MTG 36MW	MTG 36MW	
PMM	-	PM 36MW	PM 36MW	-
Mission	-	Radar 1MW	Radar 1MW	-
Loads(ML)		EMRG 20MW		
Vital Loads	0.5MW	0.5MW	0.5MW	0.5MW
SV Loads	1MW	1MW	1MW	1MW
NV Loads	1MW	1MW	1MW	1MW

power to both starboard bus and port bus, and the loads are equally shared between them.

Under the proposed distributed reconfiguration structure, the coordinator continuously assesses the current subsystem configurations and determines the distributed reconfiguration strategies when necessary. For each zone, it is assumed that:

- (1) To accomplish the given mission, all of the required mission loads must stay online. Otherwise, the state is considered illegal.
- (2) Unless specified otherwise, PGMs, especially MTGs, always provide power to both buses at the same time to limit the step load reductions and prevent generation module shut down due to over-speed caused by significant and sudden load drops.
- (3) SV and NV loads can be shed simultaneously.
- (4) During load shedding, NV loads get shed first due to their relatively low priorities, and SV loads get shed next.
- (5) Requests from NV loads are considered non-critical. Requests from SV and vital loads are considered critical.
- (6) Zone 2 and Zone 3 are considered vital in this example due to the mission loads located within them.
- (7) The switching actions of breakers are treated as discrete events. It should be also noted that to simplify the description, it is assumed that breakers can be switched on/off within a reasonably short period of time. Therefore, their switching operations can be carried out together in one event.

To validate the performance of the proposed reconfiguration method, four scenarios with random faults have been evaluated in the following discussion. Two common reconfiguration approaches are performed for comparison purposes: a distributed MAS-based reconfiguration strategy as presented in [15] (hereinafter referred to as MAS) and a central GA-based reconfiguration strategy (hereinafter referred to as GA). The following results are obtained based on implementations in the Matlab environment on a computer with an Intel i7-9750H processor and 16GB of memory. To implement the GA-based reconfiguration, the population size and generation size are set as 60 and 100, respectively, and the mutation probability is set to 0.01.

A. Fault scenario 1

The first scenario considers four faults on the power cables connecting Zone 1 PCM to starboard, Zone 2 radar to starboard, Zone 2 PM to starboard, and Zone 3 radar to port (marked with crosses in Fig.2). It is obvious that Zone 1, 2, and 3 enter their faulty configurations due to these faults and thus require reconfiguration.

First, we analyze the subsystem reconfigurability of Zone 1, which has a BDG and three types of ZL inside. To represent the current configuration of Zone 1, we first use “0”, “1”, and “2” to indicate the connection status of ZL, with “0” indicating no connection, “1” indicating the connection to starboard bus,

and “2” indicating the connection to Zone 2. Then, for each type of ZL, indicators of “0” and “1” can be used to represent their online status, with “0” indicating offline, and “1” indicating online. Combining the connection indicator and online indicators, we can use a string of four indicators to capture the discrete states of Zone 1. For example, when the discrete state of Zone 1 is “1111”, the first “1” means the ZL is connected to starboard, the second “1” means the vital loads are online, the third “1” means the SV loads are online, and the last “1” means the NV loads are online. After the faults, Zone 1 enters the faulty configuration with the initial state “0000”, and its states can be reduced to 5 after removing the illegal states below:

$$Q_{il,1}^{\uparrow} = \left\{ \begin{array}{l} 1111, 1110, 1100, 1000, 1101, 1011, 1010, \\ 1001, 0111, 0110, 0100, 0001, 0101, 0011, \\ 0010, 2101, 2011, 2101, 2001 \end{array} \right\}$$

Note that the aforementioned illegal states are identified based on the general operation principles as well as the assumptions made above for the system under study. Specifically, in this example, assumption (4) asserts that the NV loads should be shed before the SV loads. Therefore, it is easy to justify that any Zone 1 state containing the last three digits “001”, “010”, “011”, and “101” are illegal. Similarly, it is obvious that no zonal load can be online when there is no connection, which means all of the states beginning with “0” and have non-zero ZL operation status are considered illegal. Following the faults, all the connections to the faulted bus (the starboard bus in this case) in Zone 1 are lost due to the fault, which suggests that all of the states beginning with “1” are illegal. Illegal states for the other zones can be identified and removed in a similar fashion during the initialization process of the proposed approach, all before the set of system states is generated. Due to the potential size of the configuration sets, this identification and removal process is conducted automatically.

The initial state of Zone 1 is $Q_{0,1} = 0000$. Since $Q_{0,1} \cap Q_{il,1}^{\uparrow} = \emptyset$, the first condition of Theorem 1 is satisfied. Therefore, there exist the following controls for the legal states:

$$\begin{aligned} \Phi(0000) &= \{ZL \text{ to Zone 2}\} \\ \Phi(2000) &= \{ZL \text{ no connection, VL online}\} \\ \Phi(2100) &= \{Shed VL, SV load online\} \\ \Phi(2110) &= \{Shed SV load, NV load online\} \\ \Phi(2111) &= \{Shed NV load\} \end{aligned}$$

and Zone 1 is capable of returning to the normal operation configuration when it enters $Q_{re,j} = 0000$, which suggests:

$$Q_{re,j}^{\uparrow} = \{0000, 2000, 2100, 2110, 2111\}$$

Because $Q_{0,j} \cap Q_{re,j}^{\uparrow} \neq \emptyset$, the second condition of Theorem 1 is also satisfied. Therefore, it can be concluded that Zone 1 can be reconfigured to return to the normal operation configuration. Algorithm 1 can then be applied to determine an optimal reconfiguration event sequence to achieve the best performance. The optimal event sequence can be obtained as $\{ZL \text{ to Zone 2, VL online, SV load online, NV load online}\}$ which can lead the subsystem to the state of “2111”, and also requests a global reconfiguration event of $\{Zone 2 ATG to port\}$. Note that while two possible events $\{Zone 2 ATG to port\}$, $\{Zone 2 MTG to port\}$ exist for zonal controller 1, according to Assumption 2, it is more desirable for the MTG to supply both port and starboard buses simultaneously. This assumption indicates that in the reconfiguration algorithm, the MTG can be configured to supply only the port or only the starboard if there are no other alternatives. Therefore, the zonal controller for Zone 1 would choose $\{Zone 2 ATG to port\}$ as the

more favorable global reconfiguration event.

The process to check the subsystem reconfigurability of Zone 2 is similar to that of Zone 1. We can first use “0”, “1”, “2”, and “3” to represent the connection status of PGM where “0” means offline, “1” means connection to starboard, “2” means connection to port, and “3” means connection to starboard and port. Similar to Zone 1, “0” and “1” are used to represent the online status of each component. The status of PMM and Radar can be represented by the close/open status of the breakers inside. In this way, the state of Zone 2 can be represented by a string of ten indicators. After the faults occurred, Zone 2 enters faulty configuration with the initial state of “3310101111”, and its states can be reduced to 512 after removing the illegal states. After applying Algorithm 1, the optimal subsystem reconfiguration event sequence can be obtained as: $\{PM \text{ to port (close SW2 in PM), Radar to port (close SW2 in Radar)}\}$, and it also requests a global event of $\{Zone 3 \text{ PM to starboard}\}$. It should be noted that in this scenario, while it is possible to feed maximum loads in-zone by only connecting MTG to port, the subsystem prefers to request the global reconfiguration according to assumption (2). Lastly, the subsystem reconfigurability of Zone 3 is straightforward. The optimal event sequence can be obtained as $\{Radar \text{ to starboard}\}$.

Once the zonal reconfiguration is complete, the global reconfigurability can be analyzed. The coordinator receives two separate requests from Zone 1 and 2. Since the request from Zone 2 does not influence the loads served in Zone 3, this request can be accepted. However, the request from Zone 1 can only be partially accepted due to the power capacity constraints of PGM in Zone 2, which is considered a vital subsystem. Therefore, Zone 1 enters the state of “2110” with the NV loads shed offline.

The optimal reconfiguration event sequence for scenario 1 is shown in Table II, as well as the loads remaining online upon the completion of the reconfiguration. The results obtained from MAS and GA are also presented. It can be seen that MAS results in over-curtailment of loads.

B. Fault scenario 2

The second scenario considers four faults occurring on power cables connecting Zone 4 IPNC to Zone 3, Zone 3 port switchboard and Zone 2, Zone 3 ATG to starboard, and Zone 3 PM to starboard (marked with triangles in Fig.2). Following the faults, Zone 2, 3, and 4 enter their faulty configurations and require reconfiguration.

The procedure to analyze the subsystem reconfigurability is similar to the first scenario, and the complete reconfiguration event sequence is given in Table III. The reconfiguration results of MAS and GA are also presented. It is noted that in this scenario the MTG in Zone 3 is configured to only feed port bus due to the objective of serving maximum loads as outlined in Algorithm 1, although this may expose the MTG to potential operation risks.

C. Fault scenario 3

In this scenario, we consider a scenario where seven random-selected faults occur simultaneously. These faults are on the power cables connecting Zone 4 ZL to port, Zone 4 starboard switchboard and Zone 3, Zone 3 radar to starboard, Zone 3 PM to starboard, Zone 3 starboard switchboard and Zone 2, Zone 2 ATG to port, and Zone 2 ZL to port (marked with circle in Fig.2). Following the faults, Zone 2, 3, and 4 enter their faulty configurations and require reconfiguration.

The reconfiguration results as well as the reconfiguration event sequences for all three approaches are given in Table IV.

TABLE II
RECONFIGURATION RESULTS FOR SCENARIO 1 (MW)

Zone	ML	PM	VL	SV	NV	Events	PGM	Method
1	-	-	0.5	1	0	ZL to Zone 2 VL online SV loads online	-	Proposed strategy
2	1	36	0.5	1	1	PM to port Radar to port ATG to port	MTG:35.5 ATG:4	
3	1	36	0.5	1	1	PM to stbd Radar to stbd	MTG:35.5 ATG:4	
4	-	-	0.5	1	1		ATG:4	
1	-	-	0.5	0	0	ZL to Zone 2 Shed NV loads Shed SV loads	PORT:41.5 STBD:21.5	MAS
2	1	18	0.5	1	0	PM to port PM half-load Radar to port Shed NV loads		
3	1	36	0.5	1	1	Radar to stbd		
4	-	-	0.5	1	1			
1	-	-	0.5	1	1	ZL to Zone 2	PORT:61.5	GA
2	1	36	0.5	1	1	PM to port Radar to port MTG to port	STBD:21.5	
3	1	36	0.5	1	1	Radar to stbd ATG to port		
4	-	-	0.5	1	0	Shed NV loads		

D. Fault scenario 4

In this scenario, we focus on examining the performance of the proposed algorithm in handling sequential faults. We assume that the SPS is first damaged with four faults as demonstrated in Fault scenario 1, then after 2 seconds, another set of four faults occur as demonstrated in Fault scenario 2. Here we assume that events

TABLE III
RECONFIGURATION RESULTS FOR SCENARIO 2 (MW)

Zone	ML	PM	VL	SV	NV	Events	PGM	Method
1	-	-	0.5	1	1			Proposed strategy
2	1	36	0.5	1	0	Shed NV load	MTG:36 ATG:4	
3	1	36	0.5	1	1	PM to port MTG to port ATG to port	MTG:35 ATG:4	
4	-	-	0.5	1	1	ATG to stbd	ATG:4	
1	-	-	0.5	1	1		PORT:41	MAS
2	1	36	0.5	1	0	Shed NV load	STBD:24	
3	1	18	0.5	1	1	PM to port PM half-load ATG to port		
4	-	-	0.5	1	1			
1	-	-	0.5	1	1		PORT:59	GA
2	1	36	0.5	1	0	Radar to stbd Shed NV load	STBD:24	
3	1	36	0.5	1	1	PM to port MTG to port ATG to port Radar to port		
4	-	-	0.5	1	1	ATG to stbd		

TABLE IV
RECONFIGURATION RESULTS FOR SCENARIO 3 (MW)

Zone	ML	PM	VL	SV	NV	Events	PGM	Method
1	-	-	0.5	1	0	Shed NV loads		Proposed strategy
2	1	36	0.5	1	1	ATG to stbd ZL to Zone 1 Radar to port	MTG:36 ATG:4	
3	1	36	0.5	1	1	PM to port MTG to port Radar to port	MTG:36 ATG:4	
4	-	-	0.5	1	1	ZL to Zone 3 ATG to stbd	ATG:3	
1	-	-	0.5	1	0	Shed NV loads	PORT:37.5	MAS
2	1	36	0.5	1	0	ATG to stbd ZL to Zone 1 Shed NV loads	STBD:26.5	
3	1	18	0.5	1	1	PM to port PM half-load Radar to port		
4	-	-	0.5	1	1	ZL to Zone 3		
1	-	-	0.5	1	1		PORT:56	GA
2	1	36	0.5	1	0	ATG to stbd ZL to Zone 1 Radar to port Shed NV loads	STBD:27	
3	1	36	0.5	1	1	PM to port MTG to port Radar to port		
4	-	-	0.5	1	1	ZL to Zone 3 ATG to stbd		

are executed every 0.3s, and different local controllers can execute events simultaneously. It is evident that the same reconfiguration procedures as given in Table II will be executed within 2s to handle the first set of 4 faults. Following the second set of faults, the second part of the reconfiguration process begins, as shown in Table V. Note that in this scenario, GA becomes infeasible due to its relatively long calculation time and the short time interval between two sets of sequential faults.

TABLE V
RECONFIGURATION RESULTS FOR SCENARIO 4 (MW)(FOLLOWING SCENARIO 1)

Zone	ML	PM	VL	SV	NV	Events	PGM	Method
1	-	-	0.5	1	0	Shed NV loads		Proposed strategy
2	1	36	0.5	1	0	MTG to port Shed NV loads	MTG:36 ATG: 4	
3	1	36	0.5	1	1	PM to port MTG to port ATG to port	MTG:34.5 ATG:4	
4	-	-	0.5	1	1	ATG to stbd	ATG:3.5	
1	-	-	0.5	0	0	Shed SV loads	PORT:40.5	MAS
2	1	18	0.5	0	0	Shed SV loads	STBD:3.5	
3	1	18	0.5	1	1	PM to port PM half-load ATG to port		
4	-	-	0.5	1	1			

E. Comparison and discussion

In the following discussion, the performance of three reconfiguration strategies for the four above fault scenarios are compared

from three different aspects: calculation time, execution step, and power served. The results are shown in Table VI. Here, the calculation time measures the time used to calculate the reconfiguration solution. For the execution step, we assume that each subsystem can be executed in parallel. It should be noted that due to the centralized nature of GA, its execution mechanism is essentially different. Therefore, the execution time of GA is not included in the comparison. However, it is sufficient to point out that the calculation time of GA is significantly longer than the other two reconfiguration strategies as shown in Table VI.

TABLE VI
COMPARISON OF RECONFIGURATION STRATEGIES

	Scenario	Proposed strategy	MAS	GA
Calculation time	1	0.03344s	0.02871s	1.90026s
	2	0.03446s	0.02199s	1.74817s
	3	0.03512s	0.02353s	1.30192s
	4	0.03247s	0.02285s	-
Execution Step	1	3	3	-
	2	3	2	-
	3	3	2	-
	4	3	2	-
Power served	1	83MW	63MW	83MW
	2	83MW	65MW	83MW
	3	83MW	64MW	83MW
	4	82WM	44MW	-

From Table VI, it can be observed that the proposed distributed reconfiguration strategy requires more steps and takes more time than MAS. In Fault scenario 1, both methods need three steps to complete the reconfiguration process, while MAS only needs two steps in the other scenarios. This is because MAS always checks other available feeders at the first step for support, and then performs load shedding if the capacity of these feeders is insufficient. However, compared with MAS, the proposed reconfiguration strategy is capable of serving more loads in all four fault scenarios studied. This is because MAS-based strategy only checks whether there exists any available supporting feeder after faults. If such a feeder exists and its capacity permits, it will be connected for restoration. Otherwise, load shedding will occur. This mechanism, although convenient, leads to the lack of ability for MAS to comprehensively reroute the inter-zone and intra-zone power flow, causing more load shedding. On the other hand, as a centralized approach, GA takes the longest time to calculate and serves the same amount of load as the proposed method. This comparison suggests that, with all the aspects taken into consideration, the proposed method has shown its superior efficiency and effectiveness for SPS reconfiguration than the conventional strategies present in the literature.

Furthermore, we have conducted a statistical analysis that is performed based on 15 studies that consist of 3 groups of randomly-selected fault scenarios: 1 group with 4 faults, 1 group with 5 faults, and 1 group with 6 faults. In each group, 5 test scenarios with randomly selected fault locations are conducted. The average calculation time, average execution step, and average power served for each group are listed in Table VII. These results further confirm and verify our previous observations that with all the performance aspects taken into consideration, the proposed approach has shown its superior efficiency and effectiveness for

TABLE VII
COMPARISON OF DIFFERENT METHODS

	Number of Faults	Proposed strategy	MAS	GA
Average calculation time	4 5 6	0.03405s 0.03549s 0.03875s	0.02542s 0.02328s 0.02418s	1.6811s 1.7675s 1.6416s
Average execution step	4 5 6	3.2 3 3.6	2.2 2.4 2.4	- - -
Average power served	4 5 6	83.6MW 82.4MW 81.2MW	71.8MW 74MW 69MW	83.6MW 82.4MW 81.2MW

SPS distributed reconfiguration in comparison to the conventional strategies presented in the literature.

It is also worth mentioning that if a zonal controller fails to function, then it can no longer calculate, execute, or respond to any reconfiguration commands. The impact of such a failure is two-fold: on the zonal level, the zonal configuration would remain unchanged for this zone following the fault(s); on the global level, this particular zone can no longer participate in the global reconfiguration process, which indicates that the other zones cannot request any global reconfiguration event involving this zone. In summary, a zone with a failed zonal controller would be “excluded” from both the zonal and global reconfiguration processes.

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

This paper investigates the distributed reconfiguration of a zonal SPS. Compared with the current body of literature, we propose a rigorous approach to incorporate both continuous component dynamics and discrete control events to account for the hybrid nature of the SPS reconfiguration. Following disturbances, the proposed reconfiguration criteria and algorithms can effectively check the subsystem and global reconfigurability, respectively, and determine the optimal sequence of control actions to help the SPS return to steady-state under different operation conditions. Through the results illustrated in the case studies, it is demonstrated that the proposed strategy is capable of performing distributed reconfiguration for the SPS in an effective and computationally efficient way compared to notable existing research efforts in the face of random faults. We envision that the proposed reconfiguration strategy can be integrated inherently into the existing distributed energy management and control framework of SPS for seamless service restoration and damage mitigation.

B. Future Work

As an extension to our work, one can address the reconfiguration of other types of distributed SPS, such as the hybrid AC/DC SPS as studied in [26] and [27]. The effect of cyber failures of the onboard control and communication systems can also be systematically investigated, as a fault/failure tolerant distributed control system has the potential to further enhance the shipboard power system’s resilience.

REFERENCES

[1] N. Doerry, “Naval power systems: Integrated power systems for the continuity of the electrical power supply.” *IEEE Electrification Magazine*, vol. 3, no. 2, pp. 12–21, 2015.

[2] J. V. Amy, “Considerations in the design of naval electric power systems,” in *IEEE Power Engineering Society Summer Meeting 2002*, pp. 331–335.

[3] K. C. Nagaraj, J. Carroll, T. Rosenwinkel, and et al. “Perspectives on power system reconfiguration for shipboard applications,” in *2007 IEEE Electric Ship Technologies Symposium*. IEEE, 2007, pp. 188–195.

[4] K. Butler-Purry, “Multi-agent technology for self-healing shipboard power systems,” in *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*. IEEE, 2005, p. 5.

[5] N. Doerry and J. Amy, “Design considerations for a reference mvdc power system,” in *SNAME Maritime Conv.*, 2016.

[6] N. Zohrabi, J. Shi, and S. Abdelwahed, “An overview of design specifications and requirements for the mvdc shipboard power system,” *International Journal of Electrical Power & Energy Systems*, vol. 104, pp. 680–693, 2019.

[7] M. A. Guimaraes, J. F. Lorenzetti, and C. A. Castro, “Reconfiguration of distribution systems for stability margin enhancement using tabu search,” in *2004 International Conference on Power System Technology, 2004. Power-Con 2004.*, vol. 2. IEEE, 2004, pp. 1556–1561.

[8] P. Mitra and G. K. Venayagamoorthy, “Implementation of an intelligent reconfiguration algorithm for an electric ship’s power system,” *IEEE Transactions on Industry Applications*, vol. 47, no. 5, pp. 2292–2300, 2011.

[9] S. Bose, S. Pal, B. Natarajan, and et al. “Analysis of optimal reconfiguration of shipboard power systems,” *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 189–197, 2011.

[10] M. Nelson and P. E. Jordan, “Automatic reconfiguration of a ship’s power system using graph theory principles,” *IEEE Transactions on industry applications*, vol. 51, no. 3, pp. 2651–2656, 2014.

[11] K. R. Padamati, N. N. Schulz, and A. K. Srivastava, “Application of genetic algorithm for reconfiguration of shipboard power system,” in *2007 39th North American Power Symposium*. IEEE, 2007, pp. 159–163.

[12] S. Das, S. Bose, S. Pal, and et al. “Dynamic reconfiguration of shipboard power systems using reinforcement learning,” *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 669–676, 2012.

[13] X. Feng, K. L. Butler-Purry, and T. Zourntos, “Multi-agent system-based real-time load management for all-electric ship power systems in dc zone level,” *IEEE Transactions on Power Systems*, vol. 27, pp. 1719–1728, 2012.

[14] K. Huang, D. A. Cartes, and S. K. Srivastava, “A multiagent based algorithm for ring-structured shipboard power system reconfiguration,” in *2005 IEEE International Conference on Systems, Man and Cybernetics*, pp. 530–535.

[15] C. L. Su, C. K. Lan, T. C. Chou, and C. J. Chen, “Performance evaluation of multiagent systems for navy shipboard power system restoration,” *IEEE Transactions on Industry Applications*, vol. 51, no. 4, pp. 2769–2779, 2015.

[16] K. Huang, S. K. Srivastava, D. A. Cartes, and L.-H. Sun, “Market-based multiagent system for reconfiguration of shipboard power systems,” *Electric Power Systems Research*, vol. 79, no. 4, pp. 550–556, 2009.

[17] L. Agnello, M. Cossentino, G. De Simone, and L. Sabatucci, “Shipboard power systems reconfiguration: a compared analysis of state-of-the-art approaches,” *Smart Ships Technology*, pp. 1–9, 2017.

[18] H. G. Kwatny, E. Mensah, D. Niebur, and et al., “Optimal shipboard power system management via mixed integer dynamic programming,” in *IEEE Electric Ship Technologies Symposium, 2005*. IEEE, 2005, pp. 55–62.

[19] E. Mensah and et al., “Models for optimal dynamic reconfiguration and simulation of ship power systems in simulink with stateflow,” in *2007 IEEE Electric Ship Technologies Symposium*, pp. 175–179.

[20] M. Babaei, J. Shi, N. Zohrabi, and S. Abdelwahed, “Development of a hybrid model for shipboard power systems,” in *2015 IEEE Electric Ship Technologies Symposium (ESTS)*. IEEE, 2015, pp. 145–149.

[21] M. Yasar, A. Beytin, and et al., “Integrated electric power system supervision for reconfiguration and damage mitigation,” in *2009 IEEE Electric Ship Technologies Symposium*. IEEE, 2009, pp. 345–352.

[22] S. Lahiri, D. Niebur, H. Kwatny, and et al., “A software tool for automated management and supervisory control of shipboard integrated power systems,” in *2012 IEEE Power and Energy Society General Meeting*, pp. 1–7.

[23] H. Zhao, Z. Mi, and H. Ren, “Modeling and analysis of power system events,” in *2006 IEEE Power Engineering Society General Meeting*, pp. 1–6.

[24] S. Shu and F. Lin, “Fault-tolerant control for safety of discrete-event systems,” *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 1, pp. 78–89, 2013.

[25] W. Zhu, J. Shi, and S. Abdelwahed, “End-to-end system level modeling and simulation for medium-voltage dc electric ship power systems,” *International Journal of Naval Architecture and Ocean Engineering*, vol. 10, no. 1, pp. 37–47, 2018.

[26] L. He, Y. Li, Z. Shuai, J. M. Guerrero, Y. Cao, M. Wen, W. Wang, and J. Shi, “A flexible power control strategy for hybrid ac/dc zones of shipboard power system with distributed energy storages,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5496–5508, 2018.

[27] X. Zhaoxia, Z. Tianli, L. Huaimin, J. M. Guerrero, C.-L. Su, and J. C. Vásquez, “Coordinated control of a hybrid-electric-ferry shipboard micro-grid,” *IEEE Transactions on Transportation Electrification*, vol. 5, no. 3, pp. 828–839, 2019.

APPENDIX A
PROOF OF THE THEOREM 1

First, we discuss the sufficiency of this theorem based on [24]. By the definition of $Q_{il,j}^\uparrow$, we know all the transitions from the states in $Q_j - Q_{il,j}^\uparrow$ to states in $Q_{il,j}^\uparrow$ are controllable, which means $(\forall \sigma \in \Sigma_j) (\forall q \in Q_j - Q_{il,j}^\uparrow) T_j(q, \sigma) \in Q_{il,j}^\uparrow \Rightarrow \sigma \in \Sigma_{c,j}$. Therefore, the controller can disable all the transitions from the states in $Q_j - Q_{il,j}^\uparrow$ to states in $Q_{il,j}^\uparrow$. If $Q_{0,j} \cap Q_{il,j}^\uparrow = \emptyset$, the subsystem is not in $Q_{il,j}^\uparrow$ after the occurrence of fault ft_j . Hence, the controller can disable these transitions to prevent the subsystem from entering $Q_{il,j}^\uparrow$ after the subsystem reaches faulty configuration H_j . The first condition is proved.

Similarly, from the definition of $Q_{re,j}^\uparrow$, we know all states in $Q_{re,j}^\uparrow$ can enter $Q_{re,j}$, which means $(\forall q \in Q_{re,j}^\uparrow) \exists \sigma \in \Sigma_j \Rightarrow T_j(q, \sigma) \in Q_{re,j}$. Therefore, the controller can only enable transitions from the states in $Q_{re,j}^\uparrow$ to states in $Q_{re,j}$, while uncontrollable events are always allowed. When the first condition is satisfied, it is ensured that the subsystem will never enter the illegal states set. Hence, the subsystem is reconfigurable.

Next, we prove the necessity of this theorem by contradiction. Similar to [24], suppose that the subsystem is reconfigurable when $Q_{0,j} \cap Q_{il,j}^\uparrow \neq \emptyset$ for some faulty configuration H_j , that is, there exists some $q \in Q_{0,j} \cap Q_{il,j}^\uparrow$. Therefore, the subsystem states may be inside the set $Q_{il,j}^\uparrow (q \in Q_{il,j}^\uparrow)$ after the occurrence of fault ft_j . According to the definition of $Q_{il,j}^\uparrow$, once the subsystem is inside the set $Q_{il,j}^\uparrow$, it would uncontrollably enter a certain illegal state in $Q_{il,j}$. Hence, the subsystem is not reconfigurable. The first condition is proved.

Similarly, suppose the subsystem is reconfigurable when $Q_{0,j} \cap Q_{re,j}^\uparrow = \emptyset$ for some faulty configuration H_j , that is, $\forall q \in Q_{0,j} \Rightarrow q \notin Q_{re,j}^\uparrow$. Therefore, the subsystem state must be outside the set $Q_{re,j}^\uparrow$ after the occurrence of fault ft_j . According to the definition of $Q_{re,j}^\uparrow$, only states inside $Q_{re,j}^\uparrow$ can reach the states in $Q_{re,j}$. Hence, the subsystem is not reconfigurable. The second condition is proved.

APPENDIX B
DERIVATION PROCESS OF (26)

From (24), we can get:

$$\begin{aligned}
 P_{S_K, S_{K-1}} &= \sum P_{gen, S_K} - \sum P_{load, S_K} \\
 P_{S_{K-1}, S_{K-2}} &= \sum P_{gen, S_{K-1}} + P_{S_K, S_{K-1}} - \sum P_{load, S_{K-1}} \\
 &\dots \\
 P_{S_k, S_{k-1}} &= \sum P_{gen, S_k} + P_{S_{k+1}, S_k} - \sum P_{load, S_k} \quad (32) \\
 &\dots \\
 P_{S_2, S_1} &= \sum P_{gen, S_2} + P_{S_3, S_2} - \sum P_{load, S_2} \\
 0 &= \sum P_{gen, S_1} + P_{S_2, S_1} - \sum P_{load, S_1}
 \end{aligned}$$

Then, we can derive that:

$$\begin{aligned}
 P_{S_k, S_{k-1}} &= \sum_{v=S_1}^{S_{k-1}} (\sum P_{load, v} - \sum P_{gen, v}) \\
 P_{S_k, S_{k-1}} &= \sum_{v=S_k}^{S_K} (\sum P_{gen, v} - \sum P_{load, v}) \quad (33)
 \end{aligned}$$

And from (25), we can get the range of $P_{S_k, S_{k-1}}$:

$$\begin{aligned}
 \sum_{v=S_1}^{S_{k-1}} (\sum P_{load, v} - \sum P_{gen, v}^{max}) \leq P_{S_k, S_{k-1}} \leq \sum_{v=S_1}^{S_{k-1}} (\sum P_{load, v} - \sum P_{gen, v}^{min}) \\
 \sum_{v=S_k}^{S_K} (\sum P_{gen, v}^{min} - \sum P_{load, v}) \leq P_{S_k, S_{k-1}} \leq \sum_{v=S_k}^{S_K} (\sum P_{gen, v}^{max} - \sum P_{load, v}) \quad (34)
 \end{aligned}$$

After combining these two inequalities, we can get (26).

Algorithm 3 Global Reconfiguration Strategy Searching Algorithm (Less Restricted)

```

1: Input: all target configurations  $H_k^{aim}, \mathcal{C}'$ 
2: for all  $C'_y$  in  $\mathcal{C}'$  do
3:    $count_j = 0$ 
4:   for all  $H_k^{aim} \in H_k^{v-v}$  do
5:     if  $H_k^y = H_k^{aim}$  or  $H_k^y = \varepsilon$  then
6:        $count_j ++$ 
7:     end if
8:   end for
9:   save  $C'_y$  which has the largest  $count_j$  into set  $\mathcal{C}_{temp-1}$ 
10: end for
11: for all  $C'_y$  in  $\mathcal{C}_{temp-1}$  do
12:    $count_j = 0$ 
13:   for all  $H_k^{aim} \in H_k^{nv-v}$  do
14:     if  $H_k^y = H_k^{aim}$  or  $H_k^y = \varepsilon$  then
15:        $count_j ++$ 
16:     end if
17:   end for
18:   save  $C'_y$  which has the largest  $count_j$  into set  $\mathcal{C}_{temp-2}$ 
19: end for
20: for all  $C'_y$  in  $\mathcal{C}_{temp-2}$  do
21:    $count_j = 0$ 
22:   for all  $H_k^{aim} \in H_k^{v-nv}$  do
23:     if  $H_k^y = H_k^{aim}$  or  $H_k^y = \varepsilon$  then
24:        $count_j ++$ 
25:     end if
26:   end for
27:   save  $C'_y$  which has the largest  $count_j$  into set  $\mathcal{C}_{temp-3}$ 
28: end for
29: for all  $C'_y$  in  $\mathcal{C}_{temp-3}$  do
30:    $count_j = 0$ 
31:   for all  $H_k^{aim} \in H_k^{nv-nv}$  do
32:     if  $H_k^y = H_k^{aim}$  or  $H_k^y = \varepsilon$  then
33:        $count_j ++$ 
34:     end if
35:   end for
36:   save  $C'_y$  which has the largest  $count_j$  into set  $\mathcal{C}_{temp-4}$ 
37: end for
38: for all the elements in  $\mathcal{C}_{temp-4}$  do
39:   choose  $\mathcal{C}_{min}$  which has the least subsystems need to change without sending requests
40: end for
41: OUTPUT:  $\mathcal{C}_{min}$ 

```

Wanlu Zhu (M'15) received her Ph.D. degree in Control Theory and Control Engineering from Harbin Engineering University in 2016. She is currently an Associate Professor in the School of Electronics and Information at Jiangsu University of Science and Technology. Her research interest includes microgrid modeling and control, energy management, and maritime transportation. Her email address is: zhuwanlu@just.edu.cn.

Jian Shi (M'15-SM'19) received his Ph.D. degree in Electrical and Computer Engineering from Mississippi State University in 2014. Presently, he is an Assistant Professor in the Engineering Technology Department at the University of Houston. His research interest includes microgrid modeling and control, transportation electrification, and cyber-physical power systems. His email address is: jshi14@uh.edu.

Pengfei Zhi (M'15) joined the School of Electronics and Information at Jiangsu University of Science and Technology in 2017 and is currently an Assistant Professor. He received his Ph.D. degree in Control Theory and Control Engineering from Harbin Engineering University. His research interest includes microgrid control, reliability evaluation, and

risk assessment. His email address is: zhipengfei@just.edu.cn.

Lei Fan (M'15-SM'20) is an Assistant Professor in the Engineering Technology Department at the University of Houston. Before this position, he worked in the electricity energy industry for several years. He received his Ph. D in operations research from Industrial and System Engineering Department at the University of Florida. His research includes optimization methods, complex system operations, power system operations and planning.

Gino J. Lim (M'13) is Chairman and Professor of Industrial Engineering, and Hari and Anjali Faculty Fellow at the University of Houston. He holds a Ph.D. in Industrial Engineering from University of Wisconsin-Madison. His research interest lies in developing optimization techniques for solving large-scale decision-making problems under uncertainty in such areas as network resiliency, homeland security, power systems, healthcare, and transportation networks. His e-mail address is: ginolim@uh.edu.