Network Organization Paradigm:

Synergistic Effect on the Productivity of a Collaborative Organization *

Saad Alqithami
Department of Computer Science, Southern Illinois University
Carbondale, IL USA
alqithami@gmail.com

ABSTRACT

Human organizations that have begun to rely on networks for collaboration are already prolific. Networked collaboration is highly beneficial in many group activity including mixed teams of humans and agents. The prospect of understanding complex interactions on network organizations has prompted us to develop a paradigm serving as a reference model for organizations of networked individuals. In this paper we present a few salient components suggested to comprise network organizations. Network properties are central for incorporating a spectrum of collaboration styles that is outlined in our paradigm. We have introduced synergy as a specific network effect that embodies collaboration, which in turn has the potential to enhance performance at various levels of an organization as well as the overall productivity of it.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Management

Keywords

Agents Paradigm, Computational Models, Network Organization

1. INTRODUCTION

When the agents dwell inside an organization, they form repeated patterns of interactions that in result shape the structure of their network. There are many existing patterns to describe interactions within organizations, which affect their performance features. Horling and Lesser [13] described arrangements and interaction protocols that characterize working relationships among a group of individuals and termed them as paradigms. This included hierarchies, holarchies, coalitions, teams, etc. Instead, we consider those as features or patterns of interactions that can describe operating parts of an organization. For us, a paradigm is a term that capitulates representational power of a more ubiquitous perspective over its modifier. It is possible for an organization to exhibit specific features yet not be characterized by them. Even though it is rare to find a single paradigm that is the most likely to best describe an organization through its life cycle, the most fitted paradigm (i.e., the style that best describes an organization) guides us to understand an organization and appreciate its possibilities. However, agents in an open multi-agent system are self-governed by their own belief systems and have unmanaged and rational behaviors. In a previous recent work [5, 4], we explored applications that account for spontaneous exigencies in the agents' actions to benefit and shape an organization. We found that traditional organizational paradigms (i.e. hierarchical and market) lack the representational power in modeling such spontaneous structure that is formed from frameless actions and connections. The agents in that case seem to collectively form some sort of an organization based their connections over the networks they occupy. For that, we called such formation a network organization, informally described in Definition 1.

Definition 1. Network Organization *(NO) are large, semi-autonomous, ad-hoc networked individual entities with the aim of automating command and control of distributed complex tasks.*

We aspire to generalize the concept of NO and introduce a novel paradigm that is the best fit to model agents' actions in an NO that we call Network Organization Paradigm (NOP) [2]. NOP is one that manifests a network perspective over all aspects of an organization. Although at times an NO may exhibit hierarchic feature, it is not characterized by it. NOP guides us to model organizations of large firms working on complex, in scope or impact, problems [18]. A significant advancement was established in the network-centric warfare that allowed oversight and control of operations from any location on the network. Network-centricity stimulates self-organization and self-integrating coordination. The US Department of Defense embraced network centricity paradigm early on to accommodate collaboration and information resource sharing among distributed military assets and work units [1]. Location ignorance is extended in NOP to permit temporal freedom; therefore, operations can be controlled at any time; i.e., asynchronously. Another extension for NOP is to allow any credentialed network member node to exert influence on operations. In sum, NOP provides a more ubiquitously open model. This openness feature may include transparent entry and exit to the organization.

Evolving in the last thirty years, network organizations have produced significant impacts on formation and functioning of human organizations. Recent advances in social

*This paper is an extended version of the papers presented in [2, 7] networking media have accelerated impromptu formation and adaptations in human populated network organizations with benefits from collective pool of human knowledge and skills. Furthermore, cohesion in human NO is due to common human social traits such as trust and beneficence. We have embarked on modeling artificial, agent based network organizations that no doubt will possess features inspired by human NOs [17]. Although our modeling endeavor aspires to endow NO with qualities that are human centric there will remain profound differences. As erected to address specific problems, our artificial NO may lack long-term temporal history; whereas, human NO often benefit from their collective memories. Even dynamic human NO will possess temporal resilience that is not readily available in agent networks.

Earlier studies that focus on the traditional form of organizations was moved by a homologous structure formed from continuous cooperative interactions among different organizational entities [9]. In order to address the frequently changing social and economic landscape they operate on, network as part of the intra-organizational structure was introduced. But the impact of networks were not fully considered. On the other hand, the wide use of an inter-organizational structure common among many human NOs is relatively neutral and applicable to many real world applications [20]. Networks strengthen the social communication of an organization to access critical resources with other organizations on the network [11] as well as to agilely adapt to environmental changes [14]. Such properties allow the NO to plastically transform its internal structure to cope with outside social and information demands which in turn influence behaviors of its agents [19]. To this end, we anchor this article on the intra-organizational structure of NO that is formed among heterogeneous agents.

Since the NO is affected by the structure of its network, one possible effect of the network of interest in this paper is synergy among agents. Synergy is instrumental in increasing agents' efficiency on different tasks by allowing them to collaborate with each other in an NO. Network effects on the performance of a group have been demonstrated in several recent works. Liemhetcharat and Veloso [15] have studied synergy among agents using a social network framework. They built a task-based synergy graph to create an ad-hoc team that is efficient in comparison to others without interfering with existing team structure. The value of synergy is determined through agents' capabilities and distances on the graph where similar agents have similar capabilities. Parker, et. al. [16] have also used synergy inside different type of teams in order to improve the efficiency of tasks achievements. From this, inclusion of the synergy in this paper is deployed to improve agents' performances as well as their network structure.

The remainder of the paper is organized as follows. In Section 2, we give a brief introduction to the NOP and focus mainly on one of its key concepts, which is the problem profile, and describe the parameters that fall within it. Section 3 introduces one of the important properties that are inherited from the network and affects agents' behaviors called synergy. Section 4 describes the process of an NOP and how the problem profile plays an important role in navigating among agents when assigning tasks. Finally, we conclude this paper and describe some of the future possibilities of this work in Section 5.

2. UNDERSTANDING AN NO PARADIGM

There are many actual groups that rely on networks to organize their activity. Arab Spring and Science Teams are two examples. The modeling at a more generalized level cuts across domains to extricate the model from limited requirements of specific domains. A perspective that would model a generic network organization came to be considered as a paradigm. NOP can model many NO operations that are applied to open multi-agent systems. Examples are systems of river dam control, factory cells, electrical power grids, organized labor unions, and traffic control on land, sea, and space. As a paradigm, it does not functionally alter the operations to which it is applied. The paradigm can be understood in terms of the ways it permits arrangement of command and control regimes. Invariably, NO relies on the network in which it dwells. Thus, a profile of an NOP network residence is essential. NO member-nodes (i.e., agents) are critical constituents and will be delineated in separate profiles. Target problems (i.e., operations) modeled are important and will be separately profiled. For simplicity, we would care about flow of data, control, and coordination. The organizations may represent one or more parent institutions that govern its normative patterns of behavior and we will include distinct profiles for them. Broadly speaking, functioning of an NOP can be objective- (i.e., charter-) driven or pattern driven. Charter-based organizations seek to achieve specific goal(s) such as solving specific problems whereas pattern oriented organizations seek to maintain a state such as a flight formation pattern. Either of these organization types could be captured in the governance component/profile of the NOP. At this very high level, we summarize an NOP in Definition 2 followed by subsequent description of each component.

DEFINITION 2. *An NOP is a conceptualized tuple consisting of ⟨ networks-profiles, agents-profiles, problems-profiles, governance-profiles, institutions-profiles ⟩.*

Profiles in Definition 2 are key concepts in characterizing the NOP–i.e., the paradigm defines specific NO as profiles change [2]. Those parameters will be introduced in detail here as informal definitions in order to keep them intuitive because symbolism would have created brevity but needlessly obscured the ideas. We emphasize, in this paper, on describing one important parameter of an NOP: problem profile. The process where this profile plays an important role of an NO will be described in a later section.

The network profile is a graph of nodes (i.e., individuals) and links among them. The number of links will change as a result of not complete graph. The links might richly or thinly capture ties among individuals because they are most likely to be assessed when a mutual event occurs.

DEFINITION 3. *A* network profile *is presented in a tuple* $\langle \mathcal{N},\ R_{esource},\ \mathcal{P} \rangle$, *where*

- $\mathcal{N}$ *is a set of agents' profiles who are members of an NO.*
- $R_{esource}$ *is the available resources that an NO provides to the agents in order to achieve an organizational charter that is* $\mathcal{C}$*.*
- $\mathcal{P}$ *is a set of protocols to govern the activity of an NO that includes norms, rules, and roles.*

Since the entire network profile might be far larger than an NO, members of an NO are required to possess profiles. Each agent will have a public profile that contains all pertinent agent attributes including their allegiances with respect to an NO, capabilities, fitness etc. to be compared with other agents. This agent's profile is presented in Definition 4.

DEFINITION 4. *Each* agent profile*, $i \in \{\mathcal{N}\}$, is a tuple of* $\langle \vec{A_i},\ \vec{S^i_{kill}},\ R^i_{elation},\ \vec{f^i_{it}}, \vec{P^i_{reference}},\ A^i_{autonomy} \rangle$.

- *The agent i allegiance to all things it cares about is presented in A.*
- *S_{kill} is a set of skills that agent i has. It includes the capacity of the agent to handle tasks.*
- *$R_{elation}$ is the agent i's relations with other agents or organizations.*
- *f_{it} is the set of initial fitness values for different types of tasks based on previous experiences.*
- *$P_{reference}$ is a set of agent i's preferences for certain activities.*
- *$A_{autonomy}$ is the agent's autonomy-level at which it can perform tasks independent from other agents.*

There are many reasons that compel agents to connect with each other. The most pertinent reason for our formulation is to gather in an NO in order to solve a common problem. The problem can be large or small based on the goal that agents aim to achieve. Each distinct goal will correspond to a distinct associated problem profile that is used in selecting best-fit agents to perform certain tasks. A problem profile must contain task decomposition detail that provide task precedence and coordination requirements. With enough problem details, a plan can be retrieved from storage of prior plans. If no plans match, a new plan is conceived. Most often, problems will have corresponding plans that will be retrieved from a case history. When assuming that we have x set of problems and $i \in \{x\}$, problem i will have its own problem profile presented in Definition 5.

DEFINITION 5. *A* problem profile*, $i \in \{x\}$, is considered a tuple of $\langle C_{ontrol}, \vec{C_{oordination}}, G_i, P_{recedence}, I_{ndependence} \rangle$, where*

- *C_{ontrol} stands for controlling participants and available positions (i.e., roles).*
- *$C_{oordination}$ is a set of coordination rules for each agent or an agent group based on an agent profile for a possible assignment.*
- *G_i is the goal that the problem profile i exists to point out, which includes a set of tasks and set of plans that should be followed to achieve this goal. More details about G are presented in an upcoming definition.*
- *$P_{recedence}$ is the precedence of the problem domain comparing with others (i.e., the priority level of this problem to be addressed next, must be lesser or equal to 1, where 1 is the highest priority.)*
- *$I_{ndependence}$ stands for the independence of G_i in the problem-profile from other competing goals that can be executed at the same time.*

The goal G in the problem profile is generated through the governance profile of an NOP. Each goal generated will have different parameters presented in Definition 6

DEFINITION 6. *For Every goal $G_i \in \{G\} \rightarrow C$ where $i \in \{x\}$, there is a tuple: $\langle P_{lan}, IE, EE, \zeta, \vec{\theta}, \vec{\theta_{perf}} \rangle$, where*

- *P_{lan} is a set of plan(s) needed for the G_i to be achieved. It will be described in detail later on.*
- *IE is the set of internal events that is a set of planned status to be achieved.*
- *EE is the set of external events that a giving NO generates reactions based upon in order to address certain IE.*
- *ζ is the mapping function to perceive the relevance of $\forall ee_i \rightarrow ie_j$, where ee_i is the i^{th} external event of the set EE and ie_j is the j^{th} internal event of the set IE. It helps an NO to decide on which reaction it should perform as a result of a certain outside action.*
- *θ is a set of tasks agents need to handle for executing a plan, which is a set of $\langle \theta_1, \theta_2, \ldots, \theta_m \rangle$, where m is a unique independent number of tasks. Each task will have its own profile presented next.*
- *θ_{perf} is an optimal performance threshold for each $\theta \in \vec{\theta}$. If, at a certain time, performance is lower than these expected performances, the agents can be evaluated and reassigned.*

The comparison of θ_{perf} with an actual task's performance-level is used for two purposes: (a) it allows agents to report problems that they may face as well as (b) it allows assignment and in some cases reassignment. θ_{perf} does not only depends on the type of task but also on the problem profile provided, the plan to achieve them as well as the agent's level of fitness.

DEFINITION 7. *Each task $\theta_m \in \{\theta\}$ has a tuple of* $\langle$Precedence, Independence, MinFitness, $\theta_{current} \rangle$*, where*

- Precedence *is the temporal order of this task among all other tasks in the next set of tasks to be assigned to agents.*
- Independence *is to indicate that the task can be achieved alone without any other requirement of prior tasks or in overlapping task completions.*
- MinFitness *is the minimum fitness value required from an agent for this task to be achieved. It will include minimum values from agent's skills and autonomy-level.*
- *$\theta_{current}$ is the current task performance measure to be compared with the optimal performance (i.e., θ_{perf}) presented in the goal profile.*

In general, we consider a plan to be an and-or graph of tasks. Naturally, mutually dependent tasks and tasks with overlapping durations will not be independent. There are different types of tasks that need to be specified before a task is assigned; most importantly, the task independence from other tasks as mentioned in the task profile. On the one hand, the independence of one task from others means it does not require a prior task completion in order to complete the current task as well as parallel achievement. This type of tasks is assigned immediately to agents and does not require any further classification or evaluation. On the other hand, some tasks are dependent about their completion on completion of other tasks or to be completed in parallel with others. In such a scenario where dependence matters, we check the performance of the agents continuously to make sure that they are performing tasks in the expected order. For parallel tasks assigned to three or more agents or in a diffusion of a task to more than two agents, we will constantly check for the network balance [12] using the simple balance theory equation, where the network is considered balanced when the number of balanced cycles over the total number of cycles gives a balanced percentage that is bigger than threshold. We will provide more details about task assignment and reassignment in a later section when we describe the processes within an NO.

The governance profile includes the objectives of an NO (i.e., the organizational charters) aw well as patterns of which those organizational charters can be achieved. It does not interfere with both agents and problem profiles, and it governs the network profile. Other possible control are inherited form other institutions trough possibly norms [21]. The governance and institution profiles are presented in Definitions 8 and 9 respectively.

Definition 8. *A* governance profile *is a tuple of* $\langle \mathcal{C}, P_{attern}, \mathcal{F}, A_u, O_{perf} \rangle$, *where*

- $\mathcal{C}$ *is the organizational charter adapted from the network to generate goals presented by different problem domains.*
- P_{attern} *stands for the pattern of connecting problem-profiles provided to satisfy the global charter.*
- $\mathcal{F}$ *is a set of fitness functions for the whole NO to help in evaluating its functioning over time to make sure it follows in a proper direction.*
- A_u *is the autonomy level of an NO, where with the higher level of autonomy, the more independently the NO operates. It is self-declared and not externally determined.*
- O_{perf} *is an optimal organizational performance to be compared with the current performance to measure the NO progress.*

As has been mentioned before, an NO lives on a network that is often far larger than its scope and there may exist one or more institutional profile within that network environment. The network will have its own regime and control; as well institutions will provide their specific norms, rules and roles. Common protocols will be inherited directly from the institutional profile. However, when there is a contradiction in protocols between the institutions and network, NO will most likely stay neutral or might follow the institution's protocols for the worst-case scenario. Abstract definition of institution is presented in Definition 9.

Definition 9. *A* institution profile *is a tuple of* ⟨Charter, Pattern, Regulation⟩, *where*

- Charter *is much bigger than* C *of NO to give a general idea of the institution.*
- Pattern *is the way to link different NOs.*
- Regulation *are partially inherited from the network to include a set of roles, rule, and norm that is most likely inherited by its NOs.*

An NOP is intended to be a generic, meta-model that outlines prototypical NO instantiations. As such, NOP is not a direct recipe to be applied just as a set of architectural principles does not directly yield artifacts. In a later section, we describe an NOP functions via processes that connect its components in a running NO. Section 3 will focus on studying in details one type of network effect that exists among agent living on network and helps in improving their performances and the global NO performance.

3. SYNERGY EFFECT IN NOP

In any organization of networked agents, such as an NO, there is a level of inter-agent compatibility in which the agents can work together effectively. Such a measure will affect the agents' performances and, as a result, the global output of an NO. As long as there are continual interactions between the agents inside the NO, we describe these levels as synergy [15]. When a part of these interactions are not active, their synergies will be reevaluated and it may affect the total synergy of their NO. Volatility has set synergy apart from the traditional learning styles since an agent will no longer have a synergy with other agents when its connections are lost. There exists a synergy profile for each agent as well as a synergy for the local and global network for each task that has been assigned. The synergy will change over time due to the scale of dynamism in an NO while performing a certain task.

Synergy has a huge impact on organizational performance as a whole as well as on the agents' performances. In an NO, the current synergies are derived from the network-profile and modified or controlled through the governance-profile. The network profile will provide a list of the agents' profiles that contains their relations with others inside and outside the NO. The synergy contribution of an agent is of a value of "0" when he first joins an NO; then, it is derived from his relationships with others. In order to fully understand the way we derive synergy, we will describe relations in the agent profile next.

3.1 Relations formation and contribution to synergy

When a group of agents form a small world to work on a certain problem profile, the value of their relations have a huge impact on the formation as well as the coordination in this world [10]. It, in return, affects their performances and productivities. Therefore, the agents are obliged to provide, in their profiles, a set of their relations whether inside or outside the problem domain. Those relations are not static and the agents are able to improve or diminish these relations' values while performing a task. Also, new relations may be formed from existing ones to help in improving a total performance of an agent as well as the performance of her NO. The importance of relations has led us to model the agents' relations as an important parameter in their profiles.

In order to model dynamic values of relations, we capture relations in a goal-based graph. As we described previously in the problem-profile, there are different goals $\{G\}$ provided by different problems-profiles, and each $G_i \in \{G\}$ for a problem i is equivalent to a set of tasks $\langle \theta_1, \theta_2, \ldots, \theta_m \rangle$ that need to be achieved in order for the G_i to be completed. The coordination and control of those goals are also provided by the problem profile, which is generally based on the network-profile and the agents-profiles. During task achievement, values of agent's relations ebb and flow depending on nature of interactions that forms links (i.e., edges) among them. The continual changes in inter-agent connections will be used in detailing synergies.

A sociograph, as a part of the network-profile, will be build upon the contributing agents' profiles in order to model interactions among agents in each task assigned. The agents will be presented with a node and the edges are based on their provided relations in their profiles. Other parameters in the problem-profile will have an effect on the total value and shape of the graph. By the generic assembly, the sociograph is not active. However, when agents start to interact over existing but not active edges, they form an active edge through successive interaction. There are two different types of interactions: (a) explicit affinities when two or more agents have interactions with whom they have previous experiences over an existing edge in the graph (i.e., the edges of a graph is build upon original relations provided by the agents-profiles). (b) Implicit affinities are the interactions in between two agents without any previous experience between them [22]. These edges emerge from transitivity of relations (i.e., previously un-modeled relationships) to be explained shortly.

Based on the different structural configuration of the agents' coordination, the interactions of a triad can be either mutual, directed one way, directed in reverse, or null. The classification of these interactions is based on the MAN labeling introduced in [8]. This labeling is a reduction of the 64 possible configurations of a triadic closure (i.e., 4 possibilities for 3 edges in a triadic will yield a value of $4^3 = 64$) used in structural balance [12] to 16 by classifying the classes into mutual, asymmetric and null. Such labeling has been adopted to model the interactions among agents. We drive to find the value of interactions in order to evaluate current values of edges or help in forming new ones. At this point the structural balance of an NO is not essential but will play a role in monitoring task assignments discussed in section 4.

3.2 Determination of a synergistic value of an agent

In a network environment, confluence of individual actions and decisions often yield collective and residual rewards for the network that would not exist had the individuals not been active members of the network. These rewards are post mortem markers of successful interaction in the network. Although we may not be able to quantify how well a network functions during task performance, we can observe the results from time to time whenever rewards are witnessed. The degree of successful interaction is called *synergy* [15, 16]. Although, synergy will commonly remain implicit, it is always proportional to the amount of reward observed. Here, we will elucidate different ways to exhibit synergy in an NO:

- Whereas collective reward is the group reward (i.e., utility), residual reward is the reward (i.e., utility) that belongs to specific individuals. When an individual agent i is a recipient of a reward, we call action of others (say j) as benevolent toward i. When actions can be quantified, we set the benevolence of j toward i with that amount (i.e., $Ben_{j \to i}$). When a pair of individuals reciprocate benevolence, we call that synergy between them shown in Equation 1.

$$S^{i \to j}_{ynergy} = Ben_{i \to j} + Ben_{j \to i} \quad (1)$$

where i and $j \in N$

- By the time an entire group benefits from an individual action, we call that generalized benevolence. Degree of i's contribution to group $g \in \{N\}$ is denoted by $GBen_{i \to g}$. When a group appreciates i's benevolence, we consider the proportional appreciation of benevolence to be a synergy between i and group g. Appreciation can be measured by the importance of an group g bestows to the individual i denoted by $importance_i$ and synergy is shown in Equation 2.

$$S^{i \to g}_{ynergy} = GBen_{i \to g} \times Importance_i \quad (2)$$

where i is an agent belongs to $\{g\} \subseteq \{N\}$

- An important property of collaboration is timely and beneficial contribution of actions. When an individual recognizes a specific opportunity for a timely and significant action by i for another individual agent j, we capture that in complementary collaboration denoted by $CC_{j \to i}$. Whereas benevolence is a general offering of helpful action toward another, complementary collaborative action is much more directed and appreciated by the recipient since it is a response to a specific opportunity (i.e. a need fulfilled by the recipient). Similar to benevolence, synergy is generated when it is reciprocated.

$$S^{i \to j}_{ynergy} = CC_{i \to j} + CC_{j \to i} \quad (3)$$

where i and $j \in \{N\}$

- A variation of complementary action is general complementary collaboration (denoted by $GCC_{i \to g}$) when i's action benefits a group $g \in \{N\}$. With group appreciation measured by the importance value we derive a measure of synergy captured in Equation 4.

$$S^{i \to g}_{ynergy} = GCC_{i \to g} \times Importance_i \quad (4)$$

where i is an agent belongs to $\{g\} \subseteq \{N\}$

To this end, it becomes clear that the value of synergy is proportional the contributor capability and relation toward another or toward a group. It is one of the major effects of the network in an NO that determine its performance and productivity for that the previous possibilities of measures are not exhaustive.

4. THE PROCESSES OF A PROBLEM PROFILE

After the NO parameters (i.e., profiles) have been determined, an NO will begin functioning by the processes where the NO will effectively achieve problems or produce desired patterns. We focus on synergy as a predominant form of

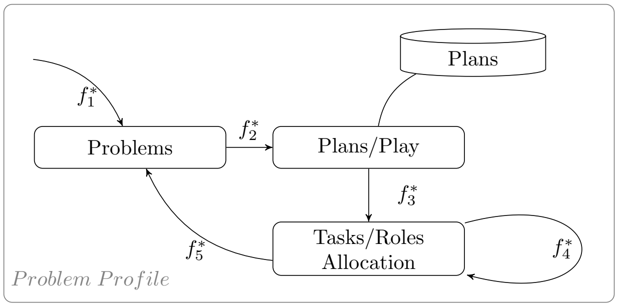

Figure 1: The flow process of a problem profile in an NOP

network effect that changes performances. This change can be at the level of individuals or groups. We will briefly outline network effects at these two levels. However, we postpone detailed discussions of processes to a latter part of this section. Figure 1 depicts a simplified sketch of flow in the problem profile, as prescribed earlier in this paper.

At the individual level, process f_4^* (see Figure 1) will continually monitor task performances and reassign tasks to each agent as needed. In part, an agent's performance is determined by its synergy with others (i.e., a network effect). Reassignments will attempt to augment synergies over a task. I.e., positive network effects will increase task performance. At the group level, process f_5^* will monitor progress on the current goal and plan in order to remedy problems with low performance on goals and plans. By initiating the process of goal re-assignment, NO will strive to increase network effect on goal performance. By initiating proper problem selection, NO will strive to fortify network effects on problems.

In an NO, the problem profiles are provided through the governance profile. Problem profiles are mainly generated to focus on the organizational charter whereas other problems are based on a perception of an external event that requires NO attention. The governance-profile will generate a set of goals. Each goal will have its own profile that shows its priority among others in the set. This set should be updated continuously in order to prioritize the set before assignment. Thus, the use of f_1^* is not only to generate a set of goals that partly satisfies the charter, it will also update this set for new generated goals, as presented in the Algorithm 1.

The governance process does not stop unless the completed goals largely satisfy the NO charter. After it generates a set of goals based on the available parameters of the NO, the problem-profile will follow the traditional steps of planning (or selecting a prior plan) for each goal. Those goals will go through the planning phase based on the priority levels assigned to them by the generator function in the governance module. In majority of cases, the problem-profile will use a case based script f_2^* to match and assign a plan or play. f_2^* may generate a plan based on the exiting agents' profiles as start up for the NO. Then, it will store them in the plan database for future reference. When a similar new goal is needed to be assigned, f_2^* will invoke similar a plan that has been assigned to similar previous goals and match the new goal with a best-fit plan.

When the agents work on a goal, they form synergy from

Data: The process of f_1 in an NO
◇ Given a $\mathcal{C}$ and P_{attern} of an NO from the governance profile;
◇ Let i be a random $G \notin \{G_n\}$;
while $\mathcal{C}$ *is not satisfied* **do**
  $\mathcal{C} \times \{ee\} \rightarrow \{G\}$
  if $\{G_n\} = null$ **then**
    Let $G_i = \{G_n\}$;
  else
    if $G_i \in \{G_n\}$ **then**
      exit;
    end
  end
  for $i : 1 \rightarrow n$ **do**
    `MergeSort` G_i based on a priority level in $\{G_n\}$;
  end
end

Algorithm 1: The process of generating and prioritize goals

the assortment of different tasks that they collaborate with each other in order to achieve. Employing those synergies will enrich the NO structure and connectively, which in turn will improve the total performance of NO. However, those synergies are not preserved and will immediately be lost by the time agents complete their current goal or depart from one goal to another. This is remedied when agents' profiles are updated continually in order to take into consideration the new formed values of synergies. As well, an NO will use the formed network of synergies to improve its performance.

After a plan has been set up for execution, f_3^* will assign tasks while taking into consideration agents' profiles. The process of f_3^* is presented in Algorithm 2. When a task has low performance, f_4^* is used to reassign tasks for other agents based on their level-of-fitness (i.e., f_{it}). The task will have low performance when the comparison of its performance (i.e., $\theta_{current}$) with expected performance presented in goal profile (i.e., θ_{perf}) is low on the case based threshold (i.e., τ). The status of an NO is reported through triggers. The reassignment of tasks/roles using f_4^* is triggered through t_1. The trigger t_1 will make sure that the condition $t_1^i : \theta^i_{current} < \theta^i_{perf}$ is satisfied before reassignment (i.e., the current performance is not less than the expected once). The performance of an NO is formed through different stages of process. This initial performance is a domain related and can be represented in a scale of "0" as a minimum to "100" for the maximum. Using those initial performances, an agent's performance at a time interval μ for a random task $m \in \{\theta\}$ is measured through Equation 5.

$$P_{erf}(\theta_m, \mu + 1) = \sum_{i,i'}^{|N|} P_{erf}(\theta_m, \mu) + \sum_{i,i'}^{|N|} S_{ynergy}(\theta_m, \mu) \quad (5)$$

where $i, i' \in \{N\}$, $\theta_m \in \{\theta\}$, and μ is a time interval.

In the case of dependent task or task assignment to more than two agents, f_4^* will use balance theory in order to examine the balance of those agents' network. The balance of the network is the percentage of the number of balanced cycles over number of existing cycles [12]. The assignment and reassignment of tasks will change over time. It will use the agents new values of synergy to update and strengthen their connections. Those synergies help in improving agents' performances, which in result change the plan for a better and faster achievement of goals.

```
Data: TaskAssignment for assigning tasks to agents
◇ Given agents' profiles that include S_kill, P_references and A_utonomy;
◇ Given a set of tasks Precedence and Independence;
◇ Let i be a random agent ∈ {N};
◇ Let θ_j be a task ∈ {θ_m} that is ready to be assigned;
for θ_j : θ_1 → θ_m do
    StateOfTask θ_j;                          ▷ Refer to Algorithm 3
    for i : 1 → |N| do
        if θ_j ∈ {P^i_reference} then
            f^i_it = Scale − of(S^i_kill + A^i_utonomy);
            if f^i_it ≥ MinFitness(θ_j) then
                Assign: θ_j → i;
            end
        end
    end
end
```

Algorithm 2: `TaskAssignment` for agents

By the time the plan is complete and tasks need to be assigned, different types of tasks have different priority and independency levels that, in result, take more time for agents to complete them. The `StateOfTask` is a simple comparison function that covers tasks' Precedence and Independence and sort them for assignment. This function is used to examine the process of assigning different types of tasks, presented in Algorithm 3.

In Algorithm 3, the "`Sort`" function applies a traditional sorting style to prioritize tasks based on their precedences. The functions "`End`" and "`Start`" are for the time intervals for each task that are used to make sure there are no overlapping in tasks achievements when assigning them. Algorithms 2 and 3 are complimentary to each other, and the functions, "`TaskAssignment`" and "`StateOfTask`" help to easily navigate between them.

The problem profile should be informed about the status of the goal assigned. When the tasks/roles have difficulties even after the reassignment, t_2 will trigger f_5^* to report the current status and ask for possible change in the current plan. In a case where the goal is taking longer than expected, f_5^* is used to update the status and to see if an extra time can be allowed for this tasks to be completed or assign a different plan. For the possibility of a goal failure, f_5^* will add the goal to $\mathcal{OLD}_{Goal}$ set, and f_2^* is required to perform the comparisons of the priorities between the two goal sets and assign the goal with the highest priority. Each goal will have a history added to its profile so that when f_2^* tries to find a plan for a previously assigned goal, it will avoid using a similar plan as assigned before and entering into an infinite loop. f_5^* will also inform the problem profile when the goal has been achieved.

Different tasks will have different performance levels. The cumulative value of those task performances present the performance values of the goal, which is also calculated through f_5^*, helps in evaluating the process of the goal assigned. The performance of each goal is determined using Equation 6.

```
Data: StateOfTask based on tasks profile
◇ Assume a level of Precedence of {0, 1.0}, where 1.0 is the optimal precedence of a task to have the highest priority among others and 0 for the complete opposite;
◇ Assume another scale of Independence of {0, 1.0}, where 1.0 for a complete independence of one task to be achieved independently from others and 0 for a total dependent on others;
if Precedence = 1.0 then
    if Independence = 1.0 then
        TasksAssignment θ_j;                  ▷ Refer to Algorithm 2
    else
        while θ_count : θ_1 → θ_j do
            if End(θ_count) ≤ Start(θ_j) then
                TasksAssignment θ_count;
            end
            θ_count ++;
        end
        TasksAssignment θ_j;
    end
else
    if Independence =1.0 then
        Sort{θ};
        TasksAssignment θ_j;
    else
        Sort{θ};
        for θ_count : θ_1 → θ_j do
            if End(θ_count) ≤ Start(θ_j) then
                TasksAssignment θ_count;
            end
            θ_count ++;
        end
        TasksAssignment θ_j;
    end
end
```

Algorithm 3: `StateOfTask` based on tasks profiles

$$G^{i}_{perf} = \frac{1}{m}\sum_{m} P_{erf}(\theta_m) \quad (6)$$

where i is the problem profile and $\forall m \in \{\theta\}$

f^*_5 will compare the current value of tasks performance with the optimal performance showing the goal profile, and report it to the problem profile. The status of completion or failure of a goal are reported to the NO through outside triggers that are out of the scope in this paper.

When the current performance passes the threshold of the minimum performance, we can consider the organization productive. Thus, the improvement in the performance will improve the productivity of an NO. Synergy helps in improving NO productivity since it improves the performance of the goals through existing network effects among its agents. Low productivity level forces an NO to adopt or plastically transform with different pattern to perform better, which may require an update to all NO profiles. The plastic transformation of an NO,addressed in [6] , can be briefly described as a group of processes that change the NO structure in order to maintain acceptable performances. Thus, it is one part of the governance profile for managing an NO shape and future directions.

5. SUMMARY AND FUTURE WORK

An NO can be a small team of two or more agents working on a common, quick goal that is possibly faster than human perceptual threshold (e.g., aerial coordination at high speeds) or a large collection of agents made up of thousands of people (i.e., possibly swarms) working on long term objectives that are possibly beyond a single human's cognitive capacity (e.g., detecting climate change). I have briefly introduced a paradigm to best model organizations dwelling on socially connected networks. This paradigm is a collection of principles, layouts, and interaction protocols that obviate the network nature of group activity as an organization. The salient properties that set an NOP apart from other organizational paradigms are: a. Openness, b. Evolving structure, c. Selfish allegiances and community social power, and d. Impromptu network topology.

Given the volatility of networks, an NOP will allow for rapid depiction and analysis of emerging and evolving networked organizations witnessed in our connected world. An NOP has introduced modular components capturing essential units to be modularly combined to define NOs. An NOP replicates many properties and features of virtual working groups. A specific salient phenomenon is how working together in networks affects their individual as well as collective productivities. Synergy is one of the main types of network effects featured in our paradigm to enhanced performance of agents and the organization.

Our plans include analyses of naturally occurring network organizations that illustrate principles indicated in our proposed paradigm as well as designs for novel applications that illustrate flexibility of our modular paradigm. We have shown by a case study that the NO paradigm is applicable for modeling real world organizations [3]. An extended work will cover more details and applications that corroborate tenets of NOs in settings such as Net-centric warfare as well as grid-based disaster responses. Of particular interest are the potential issues arising from scaling NOs to medium and large organizations, and augmenting generic NO features with features that will be required for specific domains that are unforeseen at the moment.

REFERENCES

[1] D. Alberts and R. Hayes. *Power to the Edge: Command and Control in the Information Age.* CCRP Publication Series, Washington, DC, 2003.

[2] S. Alqithami. A succinct conceptualization of the foundations for a network organization paradigm. In *29th AAAI Conference on Artificial Intelligence*, pages 4140–4141, 2015.

[3] S. Alqithami, J. Haegele, and H. Hexmoor. Conceptual modeling of networked organizations: The case of aum shinrikyo. In B. Issac and N. Israr, editors, *Case Studies in Intelligent Computing: Achievements and Trends*, pages 391–406. CRC Press, Taylor and Francis, 2014.

[4] S. Alqithami and H. Hexmoor. Spontaneous organizations: Collaborative computing model of a networked organization. In *8th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 643 – 650, 2012.

[5] S. Alqithami and H. Hexmoor. Modeling emergent network organizations. *Web Intelligence and Agent Systems*, 12(3):325–339, 2014.

[6] S. Alqithami and H. Hexmoor. Plasticity in network organizations. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 18(4):567–572, 2014.

[7] S. Alqithami and H. Hexmoor. Ubiquity of network organizations: Paradigmatic perspective and synergistic effect. In *International Conference on Collaboration Technologies and Systems*, 2015.

[8] P. Bonacich and P. Lu. *Introduction to mathematical sociology.* Princeton, NJ: Princeton University Press, 2012.

[9] S. P. Borgatti and P. C. Foster. The network paradigm in organizational research: A review and typology. *Journal of management*, 29(6):991–1013, 2003.

[10] L. Chen, G. G. Gable, and H. Hu. Communication and organizational social networks: a simulation model. *Computational and Mathematical Organization Theory*, 19(4):460–479, 2013.

[11] H. R. Ekbia and R. Kling. Network organizations: Symmetric cooperation or multivalent negotiation? *The Information Society*, 21(3):155–168, 2005.

[12] F. Heider. Attitudes and cognitive organization. *The Journal of Psychology*, 21(1):107–112, 1946.

[13] B. Horling and V. Lesser. A survey of multi-agent organizational paradigms. *The Knowledge Engineering Review*, 19(4):281–316, 2004.

[14] D. Hovorka and K. Larsen. Enabling agile adoption practices through network organizations. *European Journal of Information Systems*, 15:159–168, 2006.

[15] S. Liemhetcharat and M. Veloso. Weighted synergy graphs for effective team formation with heterogeneous ad hoc agents. *Artificial Intelligence*, 208:41 – 65, 2014.

[16] J. Parker, E. Nunes, J. Godoy, and M. Gini. Forming long term teams to exploit synergies among heterogeneous agents. Technical report, University of

Minnesota, Department of Computer Science and Engineering., 2012.

[17] S. K. Shin and W. Kook. Can knowledge be more accessible in a virtual network?: Collective dynamics of knowledge transfer in a virtual knowledge organization network. *Decision Support Systems*, 59:180 – 189, 2014.

[18] C. C. Snow and Ø. D. Fjeldstad. Network paradigm: Applications in organizational science. In J. D. Wright, editor, *International Encyclopedia of the Social and Behavioral Sciences*, pages 546 – 550. Elsevier, Oxford, second edition, 2015.

[19] S. W. Sussman and W. S. Siegal. Informational influence in organizations: An integrated approach to knowledge adoption. *Information Systems Research*, 14(1):47–65., 2003.

[20] M. van Alstyne. The state of network organization: A survey in three frameworks. *Journal of Organizational Computing and Electronic Commerce*, 7(2-3):83–151, 1997.

[21] J. Vázquez-Salceda, V. Dignum, and F. Dignum. Organizing multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 11(3):307–360, 2005.

[22] W. Zhou, W. Duan, and S. Piramuthu. A social network matrix for implicit and explicit social network plates. *Decision Support Systems*, 68:89 – 97, 2014.