ALLOCATIVE MECHANISMS AND INFORMATION EXCHANGE IN TASK PROCESSING AND INTERACTIVE NETWORKS

by

Abbas Ehsanfar

A DISSERTATION

Submitted to the Faculty of the Stevens Institute of Technology
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

____
Abbas Ehsanfar, Candidate

ADVISORY COMMITTEE

____
Paul T. Grogan, Chairman Date

____
Mo Mansouri, Chairman Date

____
Somayeh Moazeni Date

____
Steve Y. Yang Date

____
Michael Pennock Date

STEVENS INSTITUTE OF TECHNOLOGY
Castle Point on Hudson
Hoboken, NJ, 07030
2018

Abstract

ALLOCATIVE MECHANISMS AND INFORMATION EXCHANGE IN TASK PROCESSING AND INTERACTIVE NETWORKS

This thesis investigates analysis techniques and mechanisms for exchanging and valuating resources and information in a task processing network of elements (TNE) and interactive social networks. For a federated TNE, a trusted auctioneer uses a mechanism to allocate resources to computational tasks and suggests prices for exchanging resources across a federation. An operational mechanism allocates processing, storage and communication resources to computational demands. This model finds an efficient solution to combinatorial routing with technical and financial constraints. Using mixed-integer linear programming (MILP) formulation, the operational model finds an optimal solution to processing tasks, allocating links, storing and delivering data to destinations. The auctioneer assumes a federation of self-centric and rational strategic participants/bidders/federates and simulation results show improvement in collective and expected values for participants in the proposed mechanism. For auction design in a federated TNE, this research investigates an auctioneer equipped with algorithms to drive behavior of decentralized components towards higher collective-efficient metrics in a combinatorial resource allocation. This work formulates five sealed-bid auction-based algorithms for exchanging resources in multi-hop and multi-source network routing and task scheduling: 1) linear program with binary search for prices, 2) first-price double-bid reverse auction, 3) non-linear searching for prices, 4) online algorithm with closed-form solution for prices, and 5) virtual pricing with closed-form solution for prices. Collective metrics for numerical validation include normalized bids and prices, an additive value function, and convergence rates for algorithms. Extensive simulation runs using hundreds of network topologies with different configurations of elements and federates show better computational performance and higher economic efficiency for the online algorithm with a closed-form and variation-reducing solution for prices (No. 4). This thesis also investigates incentivizing mechanisms for information exchange in interactive social networks. A user classification model and a clustering model are proposed for a micro-level model interactive behavior of users and a macro-level model of circulating viral content and discourse in a social network. Analysis investigates the nature of influence and interactions on social networks. A data-driven approach distinguishes endogenous and exogenous influences and statistical analysis confirms the effect of influence on emergence of viral content on Twitter. The clustering metrics include popularity, burstiness, relevance score, consolidation, and hierarchal and temporal similarities. Conclusions outline future work to capture behavioral metrics of users on evolution of content and discourse in the interactive social networks.

Abbas Ehsanfar

Advisors: Paul T. Grogan, Mo Mansouri

Date: October 4 2018

Department: Systems and Software Division, SSE

Degree: Doctor of Philosophy

Acknowledgements

I appreciate helps from many people to receive my PhD honorably, first and foremost, my thesis advisors: Prof. Paul Grogan for his support of my research, verbal and written comments, in-person and in-depth discussions of research topics, weekly group meetings, confidence in my endeavor, and his impact on my scientific and personal life. Prof. Mo Mansouri for his continuous and cordial support of my research, future career, personal and our collective ambitions, and for his trust and confident in me during my research transition period. I'm thankful of my thesis advisory committee, Prof. Steve Yang, Prof. Somayah Moazeni, and Prof. Michael Pennock for their personal and written comments on my manuscripts and performance in multiple public and in-person presentations.

I'm highly thankful of dean of our department, Prof. Yehia Massoud for his leadership and transformation at Systems and Software Division, School of Systems and Enterprises (SSE). I'm also thankful of all faculty and staff members at our department and Stevens Institute of Technology: Alejandra Moran, Patricia Mruk, Lissettee Hernandez at school and Lena Romano from office of graduate academics for their timely support of my PhD defense and financial aid during my research. I want to thank Prof. Jose Ramirez for his support of my transition period in changing research topic and defending my proposal in 2017. I also thank Mr. Arash Vesaghi and Mr. Gillies Haymes for querying data from Twitter API and for tuning the logistic regression model in chapter 5, respectively.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Mechanisms for exchanging resources and information in interactive and collaborative systems with decentralized components and driving collective performance of these systems towards a collective goal such as social welfare, reducing cost, and maximizing utility using incentivizing schemes has gained research attention in recent years. Allocative and auction mechanisms for distributed/federated systems and information sharing mechanisms in interactive social systems can drive the collective behavior of a system-of-systems towards pre-defined collective metrics.

Resource allocation in distributed systems aggregates scalable resources with interdependency, heterogeneity, and interactions among distributed entities [1]. This may include aggregating, allocating, economic, and social/legal mechanisms in a cloud of distributed entities and computational elements. For instance, an auction-based mechanism can improve a collective metric such as social welfare or algorithmic run-time with minimum sharing of private information among interactive entities. In interactive social systems, incentivizing mechanism can result in better collective performance such as dis-incentivizing fake news and dominance of malicious bots in developing discourse in social networks. A mechanism can affect the collective behavior of systems in a wide range of applications such as cloud platforms, peer-to-peer networks, open-source communities, distributed GPUs with blockchain technology, swarm of small satellites in space systems, and dominance of malevolent agents in exchanging information on social systems.

Allocation mechanisms on cloud systems enable end-users to access a product or service with more features than what they could afford to own independently. Collaborative community (CC) platforms enable peer-to-peer (P2P) access to resources through online services. The introduction of P2P platforms enables a *sharing economy* where users can access unused capacity of other participants. Cloud computing platforms enable end-users to access distributed resources such as online processors and storage regardless of geographical distance, transportation networks enables passengers to access a personal service with a comparable cost to public service, and another sharing platform has already helped travelers when rates skyrocket in particular cities during certain times [2]. Transportation and accommodation platforms are estimated to grow in market size from $150 billion in 2016 to $500 billion in 2020 [3]. Airbnb and Uber, platforms for accommodation and ride sharing with distributed resources and pricing mechanisms, are rated among the highest-valued private companies by 2018 [4,5]. In space systems, the emergence of small satellites such as CubeSat and FemtoSat will democratize space operations for regular users, researchers and hobbyists using a cloud of small, inexpensive low-orbit constellations versus traditional complex space systems with dedicated resources and pre-defined space missions [6]. Enumerating sharing opportunities extends to crowd funding/lending (e.g. *gofundme* and *kickstarter*), online staffing (e.g. *taskrabit* and *upwork*) , and music/video streaming [7].

Sharing resources among decentralized participants can be subject to unintended consequences and negative externalities for an economy. An unregulated implementation of peer-to-peer resource exchange can bring consequence and benefits to end users. The outlook of sharing mechanisms in a sharing economy is comparable to introduction of *unions* to an industrial economy. Accordingly, in a sharing economy, part of the market might gain unprecedented advantage over others and ignore the long term externalities of profit-centric decisions regarding sharing resources. For instance, in sharing economy, the price of sharing may betray traditional practices with respect to costs, service quality, public externalities and long-term investment in resources and participants. While *micro-*

outsourcing may pay for tasks at hand (because of low marginal cost) and create short-term access to underpriced resources, it may result in fluctuations in price, resource shortage, lack of skills, and reduced welfare (e.g. retirement and insurance) in long-term. For instance, in a ride sharing or accommodating platform, matching a driver to a rider or a vacancy to a tenant enables a customer to access an affordable riding and housing, but it can also create shortage of investment in affordable housing and reliable ride sharing such as what exists in taxi and hotel industries. As it says “going freelance is hollow freedom when the wage for labor is free” [8,9].

On the other hand, sharing resources can redefine trust in a networks with computational resources. A *blockchain* is a series of blocks in chronological order which enables distributed participants to keep track of transactions without centralized record keeping. This concept is based on the third generation of enabling platforms with “computing anywhere, immediately, and among shared communities and organizations” [10]. In blockchain technology, distributed miners add a new chronological block to a group of transactions when enough validating nodes have consensus on it. This is an inexpensive decentralized mechanism for verifying and tracking timestamp of evens and transactions. Blockchain market is estimated to grow 48% a year to more than $6 billion in 2023 [11]. Nonetheless, the indirect market of blockchain is growing disproportionately as market capitalization of the world’s first decentralized cryptocurrency exceeded $150 billion by November 2017 [12]. The blockchain technology with security, immutability, transparency, and ability of P2P connection among participants, has the potential to revolutionize industries in removing middleman between buyers and sellers, verifying transactions in banking and financial industry, and tracking ownership and timestamp of transactions on diamonds, fine arts, land registration, etc.

A federation of systems (FoS) is a framework for sharing distributed resources among active participants, authorities and owners. This concept was introduced to aggregate and share resources among distributed systems under a common mechanism. Similar to “system-of-systems” or “collaborative system”, a federated system or *federation* includes a cyber-physical standard and language among collaborative entities/federates [13]. A federated system is based on operational and managerial independence of systems and relies on an extendible and nonexclusive core that incentivizes participants/federates toward an adaptive and collective goal while holding its own structure against adversarial and selfish behavior by endogenous and exogenous systems. A federated mechanism follows two goals in a distributed system: increasing collective *capacity* and *robustness*. Collaboration and participation in a federation is a rational decision by owners under a federated mechanism for sharing resources. In other words, collaboration in a federation is not *a priori* but *a posteriori* [14]. Today, this domain of federated systems extends to cloud systems, low earth orbit (LEO) satellites, swarms of drones and unmanned aviation, robotic emergency teams, etc. [15–19].

Federated satellite systems (FSS) introduce a decentralized space architectures for sharing space resources such as processing, storage, inter-satellite communication, downlink bandwidth, and instrument time among distributed space systems. In FSS, the federates have managerial, operational and goal independence to cooperate and share resources under an agreement [20]. Implementation of FSS benefits the participants of space missions in three respects: 1) improved economics of capacity utilization for resource owners, 2) accessibility of space missions to larger pool of users by removing financial barriers, and 3) more efficient resource allocation by collaborative design elements, for instance, by supplementing expensive dedicated downlink for less expensive inter-satellite relay. The ubiquity of networked resources in combination with the technical concept of virtual satellite missions (VSM) and collaborative framework of FSS can revolutionize the accessibility and economics of space resources for commercial and scientific purposes [15,21].

In FoS, pricing resources is achieved through individual objectives by each federate. However, a pricing mechanism for inter-federate resources can incentivize federates to achieve higher individual and collective value. Discrepancy in valuation of resources among federates encourages act of malicious pricing and manipulative behavior by owners or users. In economics, *externalities* affect welfare and is a reason for a policy, regulation, or intervention in market, e.g. construction in cities, polluting industries, IT infrastructure. In a more abstract sense, a pricing mechanism reduces the potential need for market intervention by introducing a self-regulating system that adapts and responds to inputs by participants. In the ride and accommodation-sharing industries, a centralized matching mechanism can introduce a universal governing role in granting licenses to drivers, balancing demand and supply without intervening hands of local authorities. Eventually, an advanced pricing and auction mechanism can target the collective benefit for all participants and decompose a collective utility into distributed objectives without the cost of centralized planning and allocation.

A task-processing network of elements (TNE) is a model of computational resources in networked structures such as clouds, satellite, robotic teams or blockchain. A *federated network* is a federation with networked elements, computational resource owners and resource users with the possibility of *interdependency* among federates in terms of resource and information exchange [22]. In an element network with multiple federates, a value-maximizing approach without an operational and financial agreement among federates doesn't give a viable solution to resource allocation. Nonetheless, a decentralized approach executed by non-collaborative federates results in sub-optimal solutions. In other words, independent operations by federates (e.g. a platform of clouds or a constellation of satellites) is not efficient while a centralized solution is not feasible given the distributed control and (potentially) design over resources by decentralized authorities. In addition, a possible combinatorics of resource exchanges among tasks and resource owners calls for an efficient and effective mechanism for financial agreement and allocation using on-board resources. A targeted operational solution with exchanges among federates fits between a value-maximizing centralized operation and a solution with independent designs/operations. In this thesis, an operational model allocates resources in scheduling computational tasks and routing data to destinations using elements from systems with decentralized designs/operations. An agreement for allocating resources and inter-federate financial exchanges is a subject of auction-based mechanisms in a federated network. Accordingly, a potential mechanism shall consider decentralized/federated objectives, economic efficiency, adversarial security, bidding language and computational complexity within a federation.

Information flow in social networks is another example of collective result of micro-level behavior by interactive contributors. Similar to auction mechanisms, incentivizing mechanisms can affect the collective behavior of interactive agents toward a better collective metric such as dissemination of useful/truthful information across a network. Accordingly, the next section of this thesis is dedicated to statistical models for understanding micro-level behavior by users and macro-level model of circulating content in a social network. Finally, this thesis investigates how information exchange among members in interactive networks can incentivize members to participate (i.e. invest resources and time) in developing and distributing content.

Among other interactive social networks, Twitter offers a platform for expressing opinion, investing time, sharing resources, and circulating content by users. In particular, a user have the chance to *tweet* (express her opinion), *retweet* (republish a content for her followers), *quote* (express her opinion along with a quote by another user), and *reply* (leave comment on another's content). These micro-actions developed by platforms such as Twitter or Sina Weibo create a new type of communication called *microblogs* that allow exchange of links, images, and brief sentences and topics over a network [23, 24]. In this platforms, *topics* range from daily life to current events, news stories, and personal interests [25] where an individual user can simultaneously consume and produce content (see [26]). Nonetheless, limitations on producing content, e.g. on the number of characters, have incentivized contributors to use an existing language differently by applying minimal grammar, frequent abbreviations, and conciseness to it. An example of these linguistic tools was the introduction of *hashtags* to twitter in 2007 [27]. These linguistics nuances in addition to recent discoveries on spreading malicious information and accounts through using a popular language by foreign agents and bots call for more inclusive methods in the analysis of linguistic discourse in a social network [28–30].

In this thesis, models of resource sharing and information exchange in TNEs and interactive networks are introduced: 1) a trusted third-party auctioneer for allocating and pricing networked resources in a federated system, 2) a general mixed-integer linear program model of allocating and scheduling network resources across a federation, 3) one-sided and two-sided auction-based mechanisms for combinatorial resource allocation in networks, and 4) a statistical model for understanding influence and content models in interactive networks. In developing an allocative mechanism, the computational challenges involve solving combinatorial routing problems based on bidding preferences by resource owners and users and pricing resources based on those constraints and alternative solutions. Accordingly, in this work, a *linear program auction with binary search* (LPA), *first-price auction* (FPA), *sequential least-square algorithm* (SLA), *online algorithm* with closed-form prices (ONA), and a *virtual pricing* for multi-path solutions (VPA) are formulated and implemented in a simulation study. Finally, in interactive networks, a classification technique for analyzing user behavior, a content model for detecting viral topics, and a statistical model for analyzing relation among user classes and content clusters are introduced.

Chapter 2 discusses related works in allocating and pricing mechanisms, federated systems, auction-based algorithms, and interactive models in social networks. Ch. 3 introduces assumptions, notations and a linear program for operational model of routing and task scheduling in networks with technical and financial constraints. Ch. 4 formulates and illustrates five auction-based algorithms for exchanging resources in a network. Ch. 5 discusses a framework for analysis of information exchange and interaction in a social networks and Ch. 6 discusses contributions in pricing, auction-based and interactive models, compares those to existing works in literature, and enumerates possible extensions to this thesis in future works.

Chapter 2

Literature Review, Problem Statement, and Questions

Resource allocation and scheduling (RAS) is widely applied to cloud systems, wireless sensor networks, ad-hoc networks, cellular networks, space systems, and blockchain [15, 20, 31–34]. Cloud systems provide scalable, automated, and instantaneous access to online software and hardware resources. In a wireless sensor and ad-hoc networks, RAS solutions allocate relaying nodes and communication bandwidth including centralized and distributed routing mechanisms, finding shortest path, optimizing energy cost, and allocating ad-hoc resource and cost to ensure quality of service in broadcasting and communication services. In commercial space systems and constellations of satellites, sharing mechanisms target unused resources to maximize economic efficiency and collective utility of stakeholders and end-users as was the case with cloud platforms. In a blockchain, an auto-executing mechanism enables distributed resources to achieve consensus, validate transactions, create trust among decentralized entities, and increase economic efficiency by eliminating middle brokers and central authorities.

Section 2.1 discusses distributed systems of clouds, blockchain etc. Sec.2.2 reviews allocative mechanisms in networks such as scheduling tasks, finding shortest paths, and allocating cost. Sec. 2.3 explores literature for pricing and auction mechanisms e.g. combinatorial auctions, pricing path bundles, and dynamic algorithms. Sec. 2.4 introduces resource allocation and mechanism design to federated cloud systems, FSS, and reviews multiple auction-based algorithms. Sec. 2.5

reviews some interactive mechanisms for analyzing collective behavior of interactive participants toward a social benefit in networks. Finally, Sec. 2.6 addresses research problems and gaps in literature and states my research questions and methodology during this thesis.

2.1 Distributed Systems

In cloud computing systems, multiple platforms can collaborate through an inter-federated resource sharing mechanism. In multi-owner supplier and customer networks, a federated cloud is an applicable concept to achieve higher collective value for a federation of systems (i.e. cloud providers), fairness and stability for participants (owners), and scalable capacity for end-users. For a cloud provider, a federated cloud platform provides instant access to computational demands over its unused resources and vice versa [35]. In practice, cloud systems may still experience low utilization rates of 20-30% mostly because of dedicated resources to end-users with inaccurate estimation of demand and performance due to heterogeneity of systems, demand spikes, etc. [36]. Multiple methods are developed to address resource utilization in cloud systems including virtualization, task consolidation, Quality of Service (QoS) aware interfaces, federated clouds etc. [36–39].

In commercial space constellations, stakeholders seek efficient use of space resources by minimizing unused resources and maximizing system capacity similar to cloud platforms. The paradigm of distributed system design versus traditional monolithic design reduces initial cost of space systems using a dynamic network of inexpensive and modular units with lower level of complexity and redundancy in design [15]. Accordingly, researchers, environmentalists, startups, etc. have access to virtual, inexpensive, instant space resources for their computational demands using real-time smart auctions and contracts. Blockchain technology is a potential solution for e-auctions and instant contracts in networks for inter-federate resource sharing and pricing [40–42].

A centralized solution (CS) is referred to the optimum RAS in a system of distributed resources. However, this solution has two issues: *scalability* for problems with higher complexity and *applicabil-*

ity to distributed authorities with decentralize missions and objective functions. First, the centralized optimization limits scalability of CS in finding solutions for growing systems in resources and size. For instance, a centralized LP formulation of multi-source combinatorial path finding problem is NP-hard and exponential in time [43, 44]. In addition, a centralized mechanism cannot realistically assume access to private information such as utility functions, available missions and resources in a network with distributed owners and authorities. On the other hand, decentralized and independent solution (IS) with no resource exchange among components is drastically inefficient for resource providers and end-users. Instead, scalable and dynamic mechanisms are developed for agreement and exchanging resources among decentralized entities. These algorithms include dynamic algorithms, consensus-based algorithms, federated mechanisms, etc. [45–47].

A federation of systems (FoS) intends to enhance the collective capacity and robustness across distributed systems, reduce investment barrier for participants, and enable collaboration among distributed systems with highly distributed level of authority, autonomy, and management. The FoS was first introduced for the architecture of cloud computing platforms [48–50]. In a federation of cloud computing systems, multiple platforms can collaborate through an inter-federate mechanism for sharing and pricing resources. In multi-owner clouds and user networks, a mechanism can offer higher collective value and stability to cloud providers and scalable computational capacity to end-users [35, 51–53]. In space systems, a federated satellite systems (FSS) architecture combines distributed systems with multiple stakeholders and decentralized designs. A distributed space system can be modeled as a task processing network of elements (TNE) of satellites and ground stations where computational missions are assigned to satellites and the resulting data are received through inter-satellite and downlink communication [14, 15, 54, 55].

A *pricing mechanism* is aimed to facilitate exchanging resources among federates with distributed resources, to increase fairness, stability, and social welfare in a network or federation. An auction-based mechanism is an approach to discover resource valuations among multiple buyers and sellers in a network. In a TNE, a federate with financial incentive is a potential participant or bidder in an auction-based mechanism. An efficient auction mechanism can successfully decompose a global objective across a FoS into local objectives for decentralized operation and decision making by participants with local and private information. Mechanism design for exchanging resources in a network has recently attained interest from different communities.

2.2 Resource Allocation

The scope of this section is resource allocation in networked systems such as cloud systems, satellite constellations, computational task assignment, wireless communication networks, blockchain, etc. In federated cloud systems, a federate assigns tasks to virtual machines and allocates communication and storage resources to computational demands. In a satellite system, an algorithm allocates space resources such as storage, bandwidth, communication channel, downlink, etc. to computational tasks. For instance, an allocation mechanism finds the shortest path from a satellite to ground stations for delivering a processed data. In communication networks, routing algorithms are developed to find optimal bandwidth-constrained paths to destination(s), minimize time and space complexity of solutions and ensure *quality of service* (QoS) [56]. In a TNE, such as a robotic team, decentralized algorithms for scheduling tasks are developed with spatial and temporal constraints to maximize social welfare from processing tasks [57, 58].

A game-theoretical approach can reduce time and space complexity of combinatorial problems such as resource allocation on clouds, routing and bandwidth allocation. In economics and game theory, *Stackelberg* is a strategic game where one player/leader moves first and other players follow. In a multi-cast routing application, this model reduces the complexity of the problem in time when a leading source optimizes a solution followed by other sources [59–61]. A collaborative mechanism in a network is aimed at maximizing a collective value or utility, called *social welfare*. An aggregated utility can be distributed or translated to individual ones using models such as Shapley, Banzhaf, nucleolus function, proportional distribution, etc. to ensure stability and fairness for participants [62–64]. In most networks, an allocation mechanism assumes a central agent and individual utility functions known to the agent and the economically-efficient mechanism maximizes social welfare.

An effective cost allocation mechanism ensures willing participation by all users, namely *core of coalition game* with all players. The existence of a core is not granted and depends on the properties of a coalition's value function in terms of super additivity and convexity. Existence of a core ensures the existence of a coalition, however, due to other desirable parameters such as *fairness*, Shapley and Banzhaf indices are introduced to measure relative effect of each agent in the *grand coalition* [65–67]. Heuristic methods are studied to reduce the computational cost of finding these indices. One method finds top influential nodes in a network and uses sampled set of permutations solvable in polynomial time. Other methods estimate the Shapley index using marginal or proportional contribution by a player [68, 69]. The results for Shapley value and proportional sharing are compared based on customer size, resource size, and the convexity of characteristic function in [35]. In a more complex method, a dynamic *nucleolus function* maximizes the minimum gain in a coalition [70]. For instance, a *satisfactory core* ensures the value from a coalition is higher than the value from leaving the coalition.

2.3 Mechanism Design

Auction design is the least intrusive solution for allocating resources and payoff among participants because it doesn't assume known utility functions and a priori private information. An auction *mechanism* includes a form of submission by participants, outcome evaluation, and winner selection. A collective value or social welfare reflects a utility for a group of participants (if definable). A participant's utility can be defined as the difference between valuation and clearing price for a winner (buyer or seller) [71]. A mechanism is designed to: allocate resources efficiently to maximize value for winners, maximize revenue for the auction designer, decrease participation cost for bidder and bid-taker (e.g. overhead communication or auction time), and ensure incentivizing metrics for participants, e.g. fairness [72]. An efficient auction is defined by being *incentive compatible*, *individual rational* and *Pareto-optimal* [73]. In an efficient auction, a resource is shared by the willing seller with least valuation for that resource and allocated to the willing buyer with most valuation for it [74,75].

Auction-based pricing mechanisms are applicable to problems in cloud computing, task assignment, bandwidth allocation, satellite systems, etc. [76–80]. Auctions are introduced to RAS and bandwidth allocation in device-to-device communication, wireless sensor, cellular, and wireless mesh networks [71,81–83], scheduling resources in distributed systems [18,77], sharing cloud resources [84], satellite systems [85], and robot exploration of spatial targets [86]. In crowdsourcing, employing a biased contest-based pricing mechanism incentivizes heterogeneous crowd-workers in a social networks to execute micro-tasks [87]. In cloud manufacturing, multiple algorithms are investigated to compose multiple tasks among cloud services [88]. In networked systems such as cloud and wireless networks, auctions are designed to reflect the preferences of heterogeneous participants, achieve an auctioneer-level goal and discourage users from adversarial behavior [84].

Combinatorial auctions are defined when bidders could place bids on multiple distinct items. These auctions are applied to various problems such as resource scheduling, online advertisement, network routing, telecommunication spectrum allocation, cloud systems, etc. [77–79,89]. Some challenges in combinatorial auctions include: complexity of winner selection in time, evaluating auction performance, bidding language, cooperation among participants, and communication overhead [72]. A submission language must reduce the information overhead and auction time and winner selection must be efficient and transparent to achieve a bidder's trust in a mechanism.

Auction mechanisms applicable to combinatorial items include single-round sealed-bid auction (e.g. first-price or reverse-price), Vickery-Clarke-Groves (VCG) mechanisms, *market-clearing* price, and *iterative* auctions [71]. Accordingly, in a single round sealed-bid auction, the bids are collected before a deadline and a revenue maximizing mechanism allocates resources and determines winners. In sealed-bid *reverse* auctions, sellers compete for buyers and in double auctions both sellers and buyers bid simultaneously. Lazar introduced *progressive second price auction* (PSP) to achieve: a) minimum communication among users and b) minimum centralized computation by the auctioneer. This mechanism consists of players that submit bids (quantity and price) and an auctioneer that allocates resources and offers new prices to bidders. The intuition behind PSP pricing and allocation mechanism is that a resource price should reflect the social opportunity cost of the allocated quantity to other participants. With certain assumptions on demand elasticity such as *concave valuation* or *complete information*, the allocation rule is stable (has ϵ-Nash equilibrium) and the players will be truthful in bidding on marginal valuation of resources (*incentive-compatibility*[1]) [83].

In an auction mechanism, adversarial behavior by participants or an auctioneer negatively affects the functionality, efficiency and fairness of the auction. Nonetheless, an auctioneer shall assume selfish participants that use private knowledge to achieve individual advantage through a mechanism. A *truthful auction* incentivizes bidders to disclose their value function and behave toward a global optimum for all participants. For instance, *collusion* is a potentially adversarial behavior by bidders in which colluding participants manipulate the auctioneer regarding their value function. VCG is proved to be vulnerable to collusion affecting auction value [90]. In addition to participants, an auctioneer (or seller) might manipulate auction winners with higher price or less resources. A solution to an adversarial auctioneer involves encryption methods to submit bids and announce allocated resources [91, 92].

2.4 Federated Systems

The FoS intends to enhance the collective value and robustness across distributed systems, reduce investment barrier for participants, and enable collaboration among distributed systems with

[1] incentive compatibility means every player can achieve the best outcome by following its true preferences, i.e. are rationally truthful in his action

distributed level of authority, autonomy and management. Similar to a coalition game, the *mechanism core* exists when the resulting value for a federate is at least equal to its power, i.e., the opportunity cost of staying in the federation [93]. Otherwise, the grand coalition is not stable and federates shall form alternative coalitions. In [67], using the concept of coalition games, a mechanism for dynamic federation formation includes two functions and their corresponding rules: *merge*, and *split*. A *merge* happens when two or multiple federates prefer to merge and create a bigger federation and achieve higher value and a *split* happens when some participants opt to split to sub-federations.

In the context of a coalition game, assume $\mathcal{N}$ is the grand coalition (coalition of all players) and $\mathcal{S}$ is a coalition of a subset of players, and $\mathcal{V}(\mathcal{S})$ shows the value of coalition $\mathcal{S}$. For a cardinality $N = |\mathcal{N}|$ of participants, the vector $\mathbf{v} = \{v_1, ..., v_N\}$ is value of a coalition for players. Then, the *game core* defines the set of solutions that disincentives players from departing a coalition and forming smaller ones [35] as:

$$C = \{\mathbf{v} : \sum v_i = \mathcal{V}(\mathcal{N}), \sum v_i \geq \mathcal{V}(S), \forall S \subseteq \mathcal{N}\}$$

A popular FoS architectures is a federated clouds. The National Institute of Standard and Technology (NIST) defines the cloud as “enabling ubiquitous, convenient, and on-demand access to shared pool of configurable resources such as bandwidth, processor, storage, applications and services” [94]. Cloud systems provide instant access to computational resources (hardware and software) with scalability, energy optimization, increased monitoring, and automation in a dynamic environment [48–50]. In a federated cloud architecture, an allocative mechanism facilitates *insourcing* and *outsourcing* services among cloud providers and end-users. These concepts denote the inter-federate direction of computational requests: task inflow and task outflow respectively. RAS in this context involves maximizing utility for end-users or revenue for cloud providers. In a federated cloud architecture, a provider can receive computational demands from end-users (demand) and other providers (insourcing) or submit computational requests to other cloud providers (outsourcing). Then, exchanging resources among providers is enabled but not assumed as a priori and a global function defines a global utility. For instance, the global utility function may reflect a higher value than sum of utilities by federates [35]. Federated satellite system (FSS) is an architecture in space domain. FSS is distinct from constellations and swarms of *homogeneous* satellites and is designed for collaborative missions and distributed operations by assuming interdependence among heterogenous space systems, constellations, and federates. In literature, FSS is associated with multiple research problems: 1) resource allocation in distributed systems, e.g. allocating downlink bandwidth, storage, processors and sensors, finding shortest path and scheduling tasks. 2) design compatibility among heterogenous systems and 3) economic agreement among stakeholders such as pricing resources for inter-federate communication [14, 15].

2.5 Information Exchange in Interactive Networks

Information exchange is critical for distributed components in an interactive platform. In the early years of social networks, Jones et al. integrated social mechanisms with principles of transaction cost economics (TCE) to discuss network governance as "mechanism for exchange" aimed at adapting, coordinating, and safeguarding exchanges in a network [95]. In a US Patent, Bergh et al. proposed using distributed user and content profiles for creating a collective social recommender system [96]. For the first time, a decade after the invention of wireless networks and world wide web, humans are able to communicate through *many-to-many* social platforms versus one-to-one and one-to-many platforms, e.g. messengers and websites. In social networks, social interactions and ties and individual characteristics and values are mutual and causal: "Analysis of interaction patterns can identify structural holes in the network as well as cliques of densely connected subgroups, where distinct cultures and norms may flourish" [97].

Multiple studies have explored models to understand the networked interdependencies among social actors and communities in interactive networks [98–102]. A dynamic model of censorship between a ruler and an observer is introduced in [103] when the ruler controls the flow of information. Weng et. al. in [104] argued that social structure and competition for limited user attention results in popularity of different memes and broad diversity among them. By their model, the authors assume no intrinsic appeal towards memes, and homogenous users with different audience size (influence) when information is passed along a social structure with an epidemic nature. *Influence maximization* is a well-known problem in social networks. Singer in [105] develops allocation and payment mechanisms to elicit true information from users in social networks.

Similar to auction-based mechanisms, interactive and agent-based models of participants are developed to investigate the effect of individual actions on collective behavior of participants. For instance, an individual action driven by attitude, emotions, perception and sentiments can be associated with an emergent content in a social network [23, 106–108]. Temporal interactive models can reveal and predict community structure and participating behavior of users [109] and to profile users by behavioral characteristics including that of bots on social networks [110, 111, 111–114].

In this thesis, a topic or a discourse in social networks is “language as a practice” that is a common term in critical discourse analysis (CDA) [115]. Accordingly, each topic has a distribution of words and solely or in combination with other topics create a document (e.g. a tweet) in a corpus [116]. Network models of users, topics, and documents have been employed to understand the dynamics of community and content development in a social network [117]. In [118] and [119], models of influential users and concepts are used to calculate the effectiveness of WikiProjects in online content development and the structure of knowledge among computer science venues. Network statistics such as centrality, closeness, betweenness, and entropy are employed to explain interconnectedness of communities and concepts [120–122]. Analyzing online interaction among users and developed topics in a network have given insight into development of applications such as discovering brand reputation and political orientation [123, 124].

2.6 Problem Statement, Questions, and Methodology

This thesis studies mechanisms for resource and information exchange in federated and social networks from system-of-systems perspective.

2.6.1 Problem Statement

In system-of-systems and collaborative systems with distributed components, economic-efficient operation cannot be assumed a priori. Resource allocation in a network with decentralized decision-making authorities and agents may result in a sub-optimal solution versus a solution by a centralized planner. Equivalently, individual behavior of agents in a social structure can result in collective behavior detrimental to the global utility of participants in a network. The goal in this thesis is to propose allocative mechanisms to manage and control resource/information exchange in networks in the following areas:

1. Develop and demonstrate allocative mechanisms for computational resources in networks with technical and financial constraints by distributed components, particularly, in combinatorial problems of multi-task scheduling and multi-hop data routing.

2. Develop and demonstrate effective auction-based mechanisms for resource exchange among decentralized and rational strategic participants/federates with multiple objective functions considering economic efficiency, computational cost, algorithm's runtime, and other collective metrics.

3. Analyze and model micro-level behavior of users in terms of interactions and micro-level output of networks in terms of discourse and content. Then explore the feasibility of learning an interactive model that raise the collective performance of networked systems with a so-

cial structure and interactive components, e.g. reducing propagation of untruthful content or adversarial behaviors in a network.

2.6.2 Research Questions

Allocative mechanism can bring those components and entities under an allocative umbrella, and financial agreement for exchanging resources. Real-world applications of sharing mechanisms include vehicles in transportation, unused space in temporary accommodation, bandwidth in cellular networks, computational resources in cloud systems, imaging resources in a satellite constellation, and sensors and spatial resources in robotic missions. A devised mechanism for sharing resources in decentralized systems shall be extendible to encompass heterogenous components and scalable for systems with growing size and number of components. For instance, in the case a satellite swarm, a mechanism shall consider heterogenous space system designs, distributed components, multiple resource owners, and ever growing size of resources in near future.

In this thesis, first, a federated system is considered for a realistic representation of decentralized space systems and a combinatorial problem of scheduling tasks and routing data is addressed in a multi-source and multi-hop network. Decentralized components and entities, i.e. federates, encapsulate distributed resources and an auctioneer simulates a mechanism for allocating and pricing resources. The first research question is:

1. How to formulate a pricing and allocative mechanism that incentivizes self-centric components and improve the collective performance of a federated engineering systems? (Chapter 3)

For resource allocation in networked systems, an auction can achieve a higher value for participants and a lower cost for the auctioneer using an individual rational, truthful, and Pareto-optimal mechanism. In context of combinatorial auctions, an auction-based algorithm involves a language for communication among bidders and the auctioneer (i.e. bidding and pricing), an operational model for winner selection, and a pricing model. In a federated topology of task processing elements, an auction-based algorithm can flexibly improve the collective metrics for participants and the auctioneer:

2. How to formulate auction-based algorithms to incentivize inter-federate exchange of resources and drive decentralized components toward better collective metrics such as higher value and lower computational cost? (Chapter 4)

Exchange mechanisms among interactive participants can also affect multiple collective metrics in social networks. In recent years, two phenomena has driven multiple researches in these networks: introducing bots as autonomous and influential agents in circulation of information and producing content based on preferences by users to manipulate public opinion in social networks. For instance, the circulation of fake news in recent years was assisted by the former bots and has resulted in public consequences. In this thesis, the last research question is:

3. How can exchange mechanisms for human resources and information contribute to better collective metrics in interactive and social networks? (Chapter 5)

2.6.3 Research Methodology

The research methodology in this thesis includes simulation study and mathematical models (and proofs). For validating allocative model of scheduling tasks and routing in networks, an Orbital Federated Task Scheduling (OFTS) application is developed. The application creates a task processing network of satellites with periodic topology for elements and an auctioneer that communicates with federates at each time step, receives preferences, and find the economic-efficient solution to technical and financial constraints at each time step. This application validates the performance of an allocative mechanism for exchanging resources in FSS. For the operational solution to scheduling tasks and routing data, mathematical models provides insights and proofs for the mechanism's performance.

A Federated Network Auction-based Routing (FNAR) application is also developed to simulate and validate auction-based algorithms using in a federated network. The application creates a network topology, communicates to federates for their bids and resources, solves MILP to find best solution given in each time step (bids), and proposes prices for exchanging resources based on five developed algorithms. The time-series and data-logs are separately stored and run for each algorithm in a file system (object-oriented pickles in Python) and each federate is trained separately for bidding in each algorithm. For the MILP model and algorithms, mathematical formulations are proofs are provided.

For influence-based model in interactive social networks, a classification model of users and a clustering model of topics and content are developed. A data-driven approach is used to validate the results for the clustering model on Twitter (by tracking and comparing real-time news) and an agent-based social systems (ABSS) simulation framework is proposed based on the statistical observations on Twitter.

Chapter 3

Mechanism Design in Federated Networks

This chapter introduces a mechanism for pricing and exchanging resources in federated task processing network of elements (TNE). An operational model is developed to allocate processing, storage and communication resources to computational demands. This model finds an efficient and stable solution to combinatorial routing and allocating resources among networked elements with technical constraints. Using mixed-integer linear programming (MILP) formulation, I find optimal solution to processing tasks, allocating links, storing and delivering data to destination. A trusted auctioneer uses a mechanism to allocate resources to computational tasks and suggests prices for exchanging resources across a federation. The proposed mechanism maximizes the collective value for a federation and ensures an expected value for each federate. The auctioneer doesn't have access to utility functions and private information on resources a priori while assumes a federation with self-centric and rational participants. An application of federated satellite systems (FSS) is developed with endogenous components such as adaptive bidding and opportunity cost of using resources. Numerical results show that the proposed mechanism improves the collective and expected values in a federation with strategic federates.

3.1 Introduction

Resource allocation and scheduling (RAS) in distributed systems with high scale of interdependency and heterogeneity necessitates allocative schemes with economic and social/legal mechanisms for exchanging computational elements among providers [1]. The collective behavior of decentralized systems is based on these mechanisms. Computational mechanisms cover a wide range of applications in clouds, peer-to-peer platforms, open-source operating systems, blockchain, space systems, and unmanned teams of autonomous or semi-autonomous vehicles and drones.

Cloud systems enable end-users to access software and hardware products or services that they cannot otherwise afford to own or maintain. Collaborative community (CC) platforms, peer-to-peer (P2P) systems, and *sharing economy* are introduced within access-based philosophy of using and owning resources through online services. For instance, transportation and accommodation platforms such as Airbnb and Uber, as applications of a sharing economy and highest-valued private companies by 2018, are estimated to grow in market size from $85 billion in 2014 to $500 billion in 2020 [3–5]. As an another example, the emergence of small satellites such as CubeSat and FemtoSat will revolutionize space operations for commercial users, researchers, and entertainers by using small, inexpensive low-orbit constellations versus traditional complex space systems with dedicated resources and pre-defined space mission [6].

For operation of a collaborative system of distributed resources and decentralized systems with technical and financial constraints, a mechanism allocates resources to computational demands/tasks. In this work, an *operational run* is a cycle of RAS and financial transactions in a federated network. In a distributed system of computational elements with tasks, a solution to the operational model combines decision variables for processing tasks, storing and transmitting data, and resolving transactions. In this chapter, three approaches are used to propose an operational model. First, a value-maximizing trusted third-party entity with knowledge of available resources and demands. Second, a decentralized approach with multiple objective functions and a *consensus* mechanism for avoiding and resolving conflicts among decision makers. Third, a distributed operation of participants as independent and monolithic systems with dedicated resources to internal missions. In the first scheme, an optimizer aggregates shared resources, allocates them to tasks, and maximizes an assumed/defined global utility function for all participants. The second approach is not explored in this research and a risk associated with the third approach is failure of entities to deliver tasks which causes redundancy in components, unused capacity, and higher design cost.

The other necessity for operating a distributed and collaborative system is a payment or pricing mechanism for resource exchange among decentralized systems. A centralized and value-maximizing payment mechanism needs further considerations in terms of *stability* and *fairness*. In this regard, a federation of systems (FoS) is introduced for exchanging distributed resources among active participants, authorities and owners. In a federation, distributed systems operate and share their resources under a common mechanism to achieve higher operating *capacity* and greater mission *robustness*. In terms of operating capacity, the mechanism facilitates resource exchanges maximizing marginal values for all participants. In addition, participation in a federated mechanism is a rational decision by owners and is not implemented *a priori*, which can bring together larger set of participants in real-world applications. On the other hand, exchanging resources with some financial flexibility creates an operational redundancy in computational missions which means flexibility for alternative solutions in case of resource failure. A federated system of task processing elements with processing, storage and communication resources is called a *federated network*.

A federated system design is a solution between the two extreme cases, when centralized or monolithic design is replaced by FoS with pricing mechanism for inter-federate resource sharing. Resource sharing in a federated network can also be divided in two categories: sharing unused capacity and pricing resources. In the first case, a federate dedicates its resources to internal tasks and shares its unused capacity with other federates, keeping priority for internal missions vs sharing (RAS by individual federates). In the second approach, federates contribute to federation by pricing resources while a centralized operational mechanism allocates resources to maximize utility functions across a federation. A more advanced case of the latter solution includes bidding for resources through an auction mechanism in the federation for pricing resources. Pricing resources versus sharing unused capacity is a more inclusive approach to manage resources in a federation. For individual federates, an intuitive reason for efficacy of a pricing mechanism is the opportunity cost of sharing internal resources. In other words, aggregated cost of internal resources is reflected in the solution to operational model. Any pricing mechanism that covers and exceeds this cost can increase the federation value for a federate. However, pricing resources are trickier for each federate as it must calculate the opportunity cost of using them, which needs information on the available resources.

Federated satellite systems (FSS) introduce decentralized space architectures for sharing space resources such as processing and storage, inter-satellite and downlink communication, and instrument time among distributed space systems. In FSS, the federates have managerial, operational and goal independence to cooperate and share resources under a federated agreement [20]. Implementation of FSS benefits the participants of space missions in three respects: 1) improved economics of capacity utilization for resource owners, 2) greater accessibility of space missions to larger pool of users by removing financial barriers, and 3) more efficient RAS by collaborative design elements, for instance, by supplementing expensive dedicated downlink for less expensive inter-satellite relay. The ubiquity of networked resources in combination with technical concept of virtual satellite missions (VSM) and collaborative framework of FSS can revolutionize the accessibility and economics of space resources for commercial and scientific purposes [15, 21].

This chapter pursues research problems in scheduling and routing problems in networks with multiple actors and utility functions because: first, an assumption about a central planner with access to information on available resources and utility functions is unrealistic, second, independent operation of multiple actors without sharing information and resources across a distributed system results in an inefficient solution to RAS. For instance, in a transportation platform such as Uber, although a centralized mechanism optimizes the operational model, the objective function is equivalent to maximum capacity utilization of resources for the platform owner rather than the participants. On the other hand, independent operation of transportation services, drivers, and passengers results in the existing transportation networks with a higher cost and a number of inefficacies such as idling time for drivers, waiting time for customers, and higher traffic for urban residents. For another example, in wireless networks in a city, millions of devices with immense processing, data, and communication capabilities are in hands of users with minimal capacity utilizations for each device. Nonetheless, in existing cellular networks with no distinction between access and ownership of computational resources and no central planner to allocate private and unused resources to other users, capacity utilization remains extremely low and access to resources remains most costly. In cloud systems, multiple cloud platforms propose computational services to users with an expected cost of scheduling tasks or acquiring a computational resource. Nonetheless, capacity utilization of resources is still low because no dynamic pricing mechanism for insourcing and outsourcing of tasks exist among platforms and users solely rely on their expectation on computational resources rather than existing practices.

In a network of tasks processing elements, e.g. satellite systems, the aforementioned problem exists in allocating processing, communication, and storage resources across the network and among decentralized entities. Then, assuming no centralized solution for scheduling tasks and routing data in a network with multiple actors, independent operations and objective functions in the network results in inefficient allocation of distributed resources, reduced capacity utilization, and higher risk of operation for independent components. In the next section, I review existing works in literature on allocative mechanisms in networks with decentralized actors and objective functions, e.g. FoS, FSS, and federated cloud systems.

Sec. 3.2 discusses more detailed insight regarding literature on tasks scheduling and routing mechanisms and enumerates detailed research questions. Sec. 3.3 introduces the mathematical notations for structural, functional, and behavioral functions with binary decisions on processing tasks, task storage, link transmission, and task resolution. Sec. 3.4 formulates objective function and operational mechanism for scheduling tasks and finding paths (routing). A mixed-integer linear programming (MILP) model is applied to solve the combinatorial problem of routing in a network (multi-source and multi-hop). Sec. 3.6 introduces components in the FSS model and Sec. 3.7 explains the simulation study and design. Finally, Sec. 3.8 discuss the numerical results, research contributions, and conclusions from this chapter.

3.2 Literature and Questions

In an element network, such as cloud providers, space systems, and robotic team or task processing, the collective value of network is driven by the value of processed tasks. In the general problem of task assignment, the best matching of tasks and agents is demanded. An optimization objective function maximizes the value of processed tasks with spatial and temporal constraints for resource capacity, network topology and temporal parameters such as task expiration or discount rate. A solution to such an objective function in task processing assignment is *conflict-free* when no task is assigned to more than one agent. The general solution to this problem schedules tasks for every agent. Nonetheless, in a general problem of task assignment with interdependent tasks, a task value depends on ordered list of processed tasks by each agent (i.e. experience) where the complexity of the problem is exponential in time as the number of decision variables increases by task permutations. The task-assignment problem can be formulated using mixed integer linear program (MILP) which can be NP-hard and computationally intractable. A hybrid approach of MILP and constraint programming (CP) relaxes the search space [47] and a decentralized auction mechanism can decompose the objective function among distributed agents and reduce complexity of centralized problem [125, 126]. A centralized auction mechanism is useful for tightly coupled task processing agents (i.e. high connectivity) [126].

In a cloud federation, a centralized solution (CS) maximizes a global utility objective function by scheduling tasks and allocating onboard resources, insourcing and outsourcing to tasks [54, 55, 127–129]. In a federated solution (FS), a mechanism proposes inter-federate pricing and resource allocation that satisfies local objective functions or constraints. For instance, a dynamic pricing model is proposed for insourcing, outsourcing, and end-user requests. For a cloud provider, the schematic objective function for a pricing mechanism is [130]:

$$\max_{\{p_i, p_o, p_e\}} V_i = \text{service value} - \text{outsourcing cost} + \text{insourcing value} - \text{service cost} \tag{3.1}$$

where V_i is the value of federate i, p_i, p_o, and p_e are resource prices for insourcing, outsourcing and end-user, *service value* is financial value of cloud service for end-users, *outsourcing* is bought service from other federates, *insourcing value* is the value of provided service to other providers, and *service cost* is combined cost of all computational tasks performed by a federate: end-users plus insourcing minus outsourcing.

Routing and scheduling algorithms in multi-hop and multi-source networks are studied in [43, 56, 131–133]. A QoS-aware algorithm for broadcasting (finding shortest paths) in a communication network may consist of bandwidth calculation, time slot reservation, rerouting, and QoS path construction [56]. A network may consist of ad-hoc links (e.g. internet of things) and a solution may use centralized, distributed, or hybrid mechanisms to allocate resources [131–134]. A centralized solution usually use linear program (LP) and needs a central agent with full knowledge of and real-time access to resources, has high communication overhead, and is combinatorial to solve. On the other hand, a distributed algorithm updates information on links using periodic request and replies (RREQ and RREP) with local routing memory on nodes [43]. The latter method reduces the complexity of routing in time with extra storage space, local updates and minimum communication. In transportation networks of manned and unmanned vehicles, dynamic algorithms are used to schedule tasks and find the shortest path [135, 136].

In terms of payment mechanisms in a communication network, the cost of processing and transmitting data shall be distributed among networked elements. The general objective function is to assign cost or distribute payoff among users in a way that incentivizes optimal and stable behavior by each participant. In this regard, a well-studied method is *proportional allocation* in communication and transportation networks in terms of *cost of stability* and equilibriums [62, 137, 138]. In addition, in a network, a mechanism may allocate resources, e.g. find the shortest path, and allocate cost of using resources to users simultaneously [63, 135, 139]. While finding energy-efficient and dynamic techniques are immensely useful in communication networks, cost allocation mechanisms are well-studied for pricing paths in transportation systems when a centralized planner with an aggregated cost exists.

In a cloud system federation, a centralized solution (CS) maximizes a global utility objective function by scheduling tasks and allocating onboard resources, insourcing and outsourcing to tasks [54, 55, 127–129]. In a federated solution (FS), a mechanism proposes inter-federate pricing and resource allocation that satisfies local objective functions or constraints. For instance, a dynamic pricing model is proposed for insourcing, outsourcing, and end-user requests. For a cloud provider, the schematic objective function for this pricing mechanism is [130]:

$$\max_{\{p_i, p_o, p_e\}} V_i = \text{service value} - \text{outsourcing cost} + \text{insourcing value} - \text{service cost} \tag{3.2}$$

where V_i is the value of federate i, p_i, p_o, and p_e are resource prices for insourcing, outsourcing and end-user, *service value* is financial value of cloud service for end-users, *outsourcing* is bought service from other federates, *insourcing value* is the value of provided service to other providers, and *service cost* is combined cost of all computational tasks performed by a federate: end-users plus insourcing minus outsourcing.

A distributed space system is a computational satellite network and ground stations where the processed tasks are delivered to ground stations through inter-satellite and downlink communication [14, 15, 54, 55]. The FSS framework is a federated architecture of distributed space systems with multiple stakeholders and a distributed system design. The federated scheme in space systems is defined versus a traditional paradigm of space system design with a centralized goal and pre-defined missions, called *monolithic* space systems. In FSS design, interaction and resource sharing among federates is enabled but not assumed to achieve a collective goal.

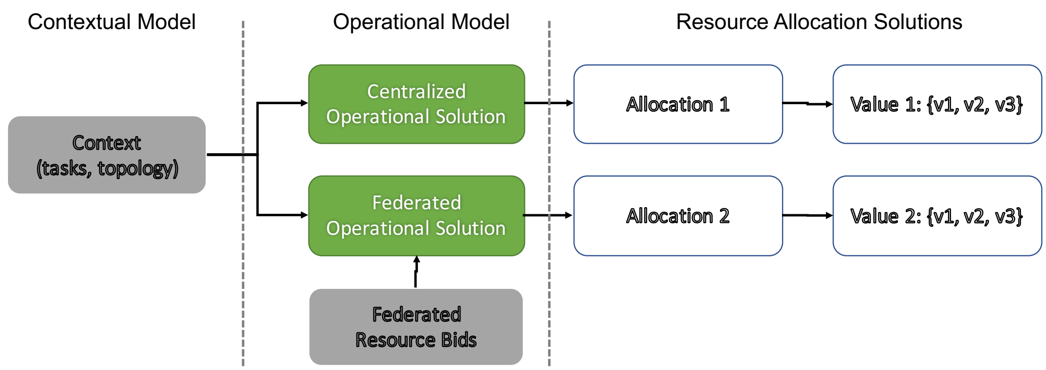


Fig. 3.1: An operational mechanism for resource allocation with centralized and federated cases.

3.2.1 Research Assumptions and Questions

In this chapter, a simulation application on FSS is developed and used to explore an allocative mechanism for scheduling tasks and routing data in a federated task processing network of elements (TNE). For RAS in a federated network of task-processing satellites, this chapter assumes:

A1 a topology of networked computational satellites with limited communication capacity among elements and dynamic and periodic topology

A2 multiple federates with internal resources and private knowledge for independent RAS

A3 an auctioneer with access to value functions of processing computational tasks in space

A4 a self-centric federate with private preferences for sharing its resources and processing tasks using its resources

A5 energy cost associated with communication resources and opportunity cost associated with storage resources

In the previous section, existing works on allocative mechanisms were discussed in federated systems, cloud platforms, wireless networks, and satellite systems. However, the proposed solutions in literature contradict assumptions in this work. The solutions to task scheduling problems in [125] and [126] assume that computational (distributed) elements are connected with limited communication resources but are managed by a centralized solver/planner. These don't stand the Asm. A2 in a federated network with decentralized decision makers. In [130], similar to some other allocative solutions to federated cloud systems, the model assume non-existing communication constraints, e.g. network topology, among cloud machines which is against Asm. A1 introduced earlier. In routing problems and finding shortest paths in wireless sensor, ad-hoc, and communication networks, e.g. in [43, 133, 136, 137], the proposed models assume constraints on communication bandwidth and links but fail to satisfy the assumptions A1 and A2 for multiple owners, dynamic and periodic designs for satellites, and network topology. The works in scheduling tasks and routing in robotic team missions and transportation systems also fail to stand by above assumptions in terms of networks topology, data routing (instant data delivery), and federated owners of resources. For mechanism design in a networked system, VCG scheme has been used in finding shortest path and pricing routing resources, however, this scheme is not well-defined for combinatorial scheduling and routing in multi-hop and multi-source networks. In addition, the latter mechanisms doesn't take into account Asm. A2 in a federated system and is vulnerable to collusion and strategic bidding by participants in Asm. A4.

The research question addressed in this chapter is: *"How to formulate a pricing and allocative mechanism that incentivizes self-centric components and improve the collective performance of a federated engineering systems?"*. This question is detailed by these sub-questions:

Q1 How to model scheduling tasks and multi-hop routing (operational model) in a federated TNE with technical and financial constraints?

Q2 How to address above problem in networks with dynamic and periodic topology (e.g. federated satellite systems)?

Q3 How a linear program applies to the above operational model in federated TNE ?

Q4 How to discover preferences by federates for sharing their internal resources, i.e. type of agents, what bidding language is used, and what information is shared by participants to an auctioneer and other participants?

Q5 What is the objective function for a pricing mechanism across the federation?

Q6 How do rational strategic participants learn (adaptive) bidding during time?

Q7 Which collective metrics can capture the performance of the auctioneer?

3.2.2 Research Methodology and Design

In this chapter, the following steps address the research question:

S1 defining structural and functional components in a federated satellite systems (FSS)

S2 formulating the operational model of a centralized RAS based on the value and cost functions of federates and resources

S3 developing an adaptive bidding model for federates for simulate strategic bidding

S4 developing a pricing mechanism to facilitate exchanging resources among federates

S5 developing an application with above components

S6 developing a simulation study to evaluate the proposed mechanism for collective metrics

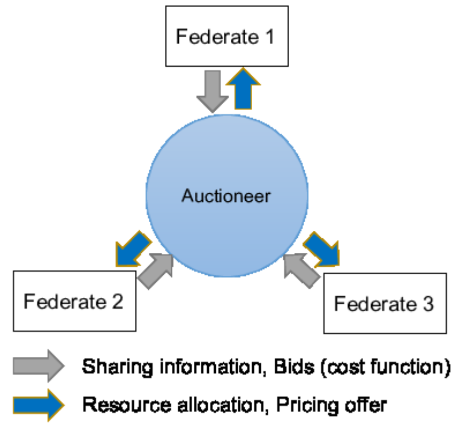
In this chapter, a general model of a federated network is introduced to increase collective value of processing and delivering tasks using distributed elements. In step S1, I employ a graph-based model for spatial topology of elements and connections, i.e. location of elements and communication links. For step S2, a global utility function is defined based on value functions of processing tasks. A trusted third-party auctioneer for pricing resources in federation is used and a one-sided reverse bidding by federates is used by resource owners to share preferences by the auctioneer. In step S3, a reinforcement learning model captures the strategic behavior of bidders. In step S4, the auctioneer uses a non-linear optimization algorithm to optimize the global utility and minimize marginal cost for each federate.

For simulation study, an application combines above modules and multiple combination of FSS configurations are used to put into test the hypothesis in this chapter. For testing the introduced method, five designs are selected that covers two and three federates and different types of satellites. Since the design cases are simple with minimum assumptions, a handful of designs can capture the performance of the proposed mechanism. Multiple collective metrics are introduced including an additive utility function for the global utility function of the federation, i.e. collective value, number of shared links among federates, bids submitted by federates, and actualized prices for exchanging resources. For results, increasing the number of exchanged resources in combination with increasing the collective value for all federates confirm the better performance of the pricing mechanism in comparison with the case with a centralized solution to bidding constraints by federates, i.e. the baseline solution. Nonetheless, a case with no financial constraint by participants (minimum bids) is global value-maximizing.

3.3 Federated Network Model

A federated network combines distributed resources for processing tasks, storing data, and communicating with other elements within a context of computational demands and monetary value

Fig. 3.2: An *auctioneer* is an independent and *trusted* entity that communicates with federates and is informed of contextual parameters and elements such as available tasks, network topology, and energy cost of communication. The auctioneer suggests solutions to operational model and prices for exchanging resources to federates.



functions for processing tasks. An *operational model* runs cycles of resource allocation and financial transactions among federates and its solution includes decision variables for assigning tasks to elements, allocating links, resolving tasks, and storing data. This section introduces and formulates an operational model run by a trusted *auctioneer* that communicates with federates and optimizes their operational value (see Fig. 3.2). Mathematical notations for structural and operational models are borrowed from models of networked systems and Infrastructure System of Systems (ISoS) introduced by Grogan and de Weck in [140].

I represent elements with nodes, feasible communication links with edges, a federation with set of federates where a federate is defined by a set of nodes, links and behaviors. An element has a federate owner and internal properties such as capacity for storage and processing tasks. A link also belongs to a federate and has properties such as source, destination and capacity for data transmission. In our general model, communication links are state-dependent, which indicates every link is associated with a state, then, every state maps to a network topology. At each time step, computational demands, or tasks, are available to one or multiple elements, for pick-up. A task has contextual issuer, element processor, and temporal value function. The element that picks up a task, agrees to its value function and a penalty amount in case of failure to deliver the task. Lastly, a federate is the decision-making authority that manages its elements and resources through behavioral functions.

In addition to physical components, functional and synthetic concepts in networked systems are introduced. A *path* is an acyclic set of feasible links to deliver data from a source to a destination element. Accordingly, a path is associated with a task which satisfies network constraints such as maximum data transmission bandwidth. At each time step, networked elements might pick up multiple tasks. In this case, multiple paths should deliver those tasks without exceeding any constraint in the network. Functional components retrieve the usage, value, and cost associated with structural model. A relational function finds the federate owner of an element or task, another finds the cumulative cash for a federate. A *data function* finds task data saved on element's internal storage, and dedicated capacity on a communication link. A *cost function* finds cost associated with network resources (communication links, paths, storage). A *value function* calculates value of resolving tasks at destination, value of an path for federates and federation value. Some *behavioral functions* describe the processes associated with processing tasks, data transmission, and resolving data or storing tasks on elements.

In the operational model, the *operational objective function* maximizes the net value of processing tasks in federation. To capture the individual rationality of federates, it minimizes cost associated with delivering tasks.

3.3.1 Structural & Functional Model

A structural model defines variables and formulates concepts in a federation. In a federated network, this model represents elements by $E = \{e_i\}$, communication links by $L = \{l_{ikt}\}$, and federates by $F = \{f_j\}$. A link l_{ikt} connects an element e_i to element e_k at time step t and has technical properties such as *capacity* for data transmission:

$$l_{ikt} = (e_i \in E, e_k \in E, t \in \mathbb{Z}^+, capacity \in \mathbb{Z}^+)$$

where an element and all links to it belong to a federate. At each time step, a set of computational tasks $\mathbf{T}_a = \{T_n\}$ appear to elements. A computational task is associated with data size, initiation time and a negative value for failure:

$$T_n = (element \in E, size \in \mathbb{Z}^+, init \in \mathbb{Z}^+, penalty \in \mathbb{R}^-)$$

A path P_{sj} is an end-to-end list of elements that connect source element e_s to a destination element e_j. The data structure associated with a path contains the associated task, communication links, cost functions, and two elements:

$$\begin{aligned}
&P_{sj} = (task \in \mathbf{T}, links = \{l_{ikt} \in L\}, \\
&\quad source = task.element \in E, last = e_j \in E, cost \in \mathbb{R}^+, \\
&\quad time = max(\{t : l_{ikt} \in links\}) \in \mathbb{Z}^+) \\
&s.t. : \\
&\quad \exists a, b : l_{sbt} \in P_{sj}.links \;\&\; l_{ajt} \in P_{sj}.links \\
&\quad \forall l_{abt} \in P_{sj}.links, b \neq j : \exists l_{bcd} \in P_{sj}.links \\
&\quad \forall l_{abt} \in P_{sj}.links : \nexists l_{cbt} \in P_{sj}.links
\end{aligned} \tag{3.3}$$

where the first two constraints ensure end-to-end connection from a source to a destination and the third constraint removes cyclic paths. The federate associated with sharing an element or a link, processing a task, or using a path is defined by functions:

$$\mathcal{F}_e : e \in E \rightarrow f \in F$$

$$\mathcal{F}_l : l_{ikt} \in L \rightarrow f = \mathcal{F}_e(e_k) \in F$$

$$\mathcal{F}_t : T \in \mathbf{T}_a \rightarrow \mathcal{F}_e(T.element) \in F$$

$$\mathcal{F}_p : P_{sj} \rightarrow \mathcal{F}_e(e_s)$$

respectively. A federate is a self-centric (strategic) entity with financial independence. The cumulative value of a federate is represented by a cash function $C(f)$ and the data size stored on an element or allocated to a link. *Data function* ($\mathcal{D}$) retrieves the stored data size on an element and the allocated data transmission on a link and its definition is modified accordingly. The data saved on internal storage of an element or transmitted by a link are retrieved by:

$$\mathcal{D}_e : e \in E \rightarrow \mathbb{Z}^+$$

$$\mathcal{D}_l : l \in L \rightarrow \mathbb{Z}^+$$

In this regard, *capacity function* is defined in similar form to $\mathcal{D}$ and finds upper constraints on $\mathcal{D}$, such as maximum storage capacity and maximum data transmission capacity per time-step on an element or a link:

$$cap_e : e \rightarrow \mathbb{Z}^+$$

$$cap_l : l \rightarrow \mathbb{Z}^+$$

For financial constraints, a *cost function* maps a link to its cost of sharing with other federates per a unit of data, i.e. $\zeta : (f \in F, l \in L) \to \mathbb{R}^{+}$:

$$\zeta : (f, l_{ikt}) \to \begin{cases} \epsilon & i \neq k \text{ and } f = \mathcal{F}(l_{ikt}) \\ c_{\mathcal{F}(l)} & i \neq k \text{ and } f \neq \mathcal{F}(l_{ikt}) \\ SP & otherwise \end{cases} \tag{3.4}$$

where ϵ is the actual marginal cost of using a link assumed to be equal for all links, c_f is the cost of sharing f's link with other federates ($\epsilon \ll \bar{c_f}$) and *SP* is storage penalty as a proxy for the opportunity cost of using scarce storage capacity. Finally, a *value function* maps a task to its value at the time of resolution at final destination $\mathcal{V}_t : (T, t \in \mathbb{Z}^{+} \geq T.init) \to \mathbb{R}$.

A path's value for a federate f is defined as a premium resulting from its associated task and inter-federate links:

$$\mathcal{V}_p : (P, f, \zeta) \to \begin{cases} \mathcal{V}_t(ts, tm) - \sum_{l \in P} \zeta(f, l) & f = \mathcal{F}_e(src) \\ \sum_{l \in P : \mathcal{F}_l(l) = f} [\zeta(f, l) - \epsilon] & otherwise \end{cases} \tag{3.5}$$

where ts (task), tm (time) and src (source) are from Eq.3.3. Finally, assume $\mathbf{P}$ as multiple paths at a time step, federated value function is defined as:

$$\mathcal{V}_{fed}(f, \zeta) = \sum_{P \in \mathbf{P}} \mathcal{V}_p(P, f, \zeta) \tag{3.6}$$

given a cost function ζ for exchanging resources.

3.3.2 Behavioral Model

The operational model, at each time step, finds the existing paths, i.e. the combinatorial problem of using multiple links to build a path. In case of existing tasks at each time step, I also have additional constraints for link capacity. To address this problem using a linear model, I need to know the longest length of a feasible acyclic path to deliver any existing tasks. The length of a path is defined as the number of its links. In this case, I define the concept of sub-step s using a time step t by dividing the latter by the maximum path length in a federated network. Assume the maximum path length m in network, for n time steps we have:

$$\mathbf{s} = \{s_{(0)}, s_{(1)} s_{(2)}, ... s_{(n*m-1)}\}$$

s.t. :

$$s_{(k*m)} \cong t_k : k \in \{0, 1, ..., n-1\}$$

and other values of s are evenly distributed between every pair of consecutive time steps. Since the value function ($\mathcal{V}$) is defined based on real time steps, function τ converts a step to its corresponding time step in a behavioral model:

$$\tau : s_{(k*m+b)} \rightarrow t_k, b < m \tag{3.7}$$

and s_t is a set of sub-steps defined after time step t_i and before its subsequent time step t_{i+1}:

$$\mathbf{s}_t : \{s \in \mathbf{s} : t = \tau(s)\}. \tag{3.8}$$

Figure 3.3 shows the sub-steps in a sample multi-task routing.

In behavioral model, an element processes a task after task pick-up and stores its data on internal storage. The operational model dedicates links and resolves the tasks. The process that

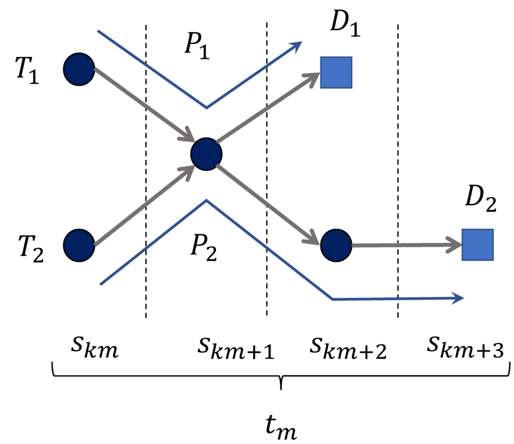


Fig. 3.3: sub-steps on delivery paths where T_i represents a computational task, D_j is a destination element, and P_i is data delivery path for T_i. A sub-step s_{km+b} of a time step t_m simulates data transition on a link. The sequence of sub-steps matches the sequence of links on a path. The number of sub-steps is equal to maximum length of the longest path(s) in a multi-path routing solution. In this example, both tasks are delivered simultaneously (t_m) but MILP considers two sub-steps for P_1 and three sub-steps for P_2.

an element follows and pick up and process a task is called *task process*:

$$
\begin{aligned}
I_{process} = I_{process}(T, s) \quad & (3.9)\\
T.init \leftarrow \tau(s) & \\
\mathcal{D}_e(T.element) \leftarrow \mathcal{D}_e(T.element) + T.size & \\
\mathbf{T} \leftarrow \mathbf{T} \cup \{T\} &
\end{aligned}
$$

where $\mathbf{T}$ is the set of processed tasks and $T.element$ is the processing element for task T, and $\mathcal{D}$ is data function.

Assume $\mathbb{D}$ is the set of destination elements when data transmission is the process of transmitting task data from source to a destination through one or multiple communication links. When a delivery path is available to an element, it can resolve a processed task by delivering it to a destination. In addition, in some circumstances such as task expiration and storage limitation, a federate can resolve a task but must pay its penalty. This *task resolving* is:

$$
\begin{aligned}
I_{resolve} = I_{resolve}(T, e, s) \\
C(\mathcal{F}_t(T)) &\leftarrow C(\mathcal{F}_t(T)) + \mathcal{V}(T, \tau(s)) \text{ if } e \in \mathbb{D} \\
C(\mathcal{F}_t(T)) &\leftarrow C(\mathcal{F}_t(T)) + T.penalty \text{ if } e \notin \mathbb{D} \\
\mathcal{D}_e(T.element) &\leftarrow \mathcal{D}_e(T.element) - T.size \\
\mathbf{T} &\leftarrow \mathbf{T} - \{T\}
\end{aligned}
\tag{3.10}
$$

And, *data transmission* function through a link is defined as:

$$
\begin{aligned}
I_{trans} = I_{trans}(T, l_{ikt}, s) \\
C(\mathcal{F}_t(T)) &\leftarrow C(\mathcal{F}_t(T)) - T.size \times \zeta(\mathcal{F}_t(T), l_{ikt}) \\
C(\mathcal{F}_e(e_k)) &\leftarrow C(\mathcal{F}_e(e_k)) + T.size \times [\zeta(\mathcal{F}_t(T), l_{ikt}) - \epsilon \times \mathbb{1}(i \neq k)] \\
\mathcal{D}_l(l_{ikt}) &\leftarrow \mathcal{D}_l(l_{ikt}) + T.size \\
I_{resolve}&(T, e, s), \text{ if } e \in \mathbb{D}
\end{aligned}
\tag{3.11}
$$

where $\mathbb{1}(condition)$ is equal to 1 when the *condition* holds and 0 otherwise. The intuition for above function is that a federate $\mathcal{F}_t(T)$ is affected by task value minus the link cost, and federate owner of the link is also affected by the link cost. However, in transmission process on storage link ($i = k$ in l_{ikt}), no internal cash or storage is affected.

In cases that an element has access to a new task, without a viable path to deliver the task at current time, it can decide on *data storage* when it stores processed data for future delivery:

$$
I_{storage} = I_{storage}(T, t) \tag{3.12}
$$

3.4 Operational Model

The operational model in a federation optimizes value and cost for its federates given a set of computational tasks and a topology for networked elements. In [39], an integer linear program (ILP)-based algorithm maximizes profit and minimizes cost in a federation of cloud resources. The operational model in this section also maximizes the collective value of processed tasks in a federation and minimizes individual cost for federates, ensuring no incentive to reroute data for a lower cost.

The MILP, at each time step, defines decision variables as:

$$x_{proc} : (T \in \mathbf{T}, e \in \mathbf{E}) \rightarrow \{0, 1\}$$

$$x_{trans} : (T \in \mathbf{T}, l \in \mathbf{L}, s \in \mathbf{s}) \rightarrow \mathbb{Z}^{+}$$

$$x_{resolve} : (T \in \mathbf{T}, e \in \mathbf{E}, s \in \mathbf{s}) \rightarrow \{0, 1\}$$

$$x_{read} : (T \in \mathbf{T}, e \in \mathbf{E}) \rightarrow \{0, 1\}$$

$$x_{store} : (T \in \mathbf{T}, e \in \mathbf{E}) \rightarrow \{0, 1\}$$

where s shows a sub-step in Eq.3.8 and we excluded time t variable for notational simplicity.

3.4.1 MILP Objective Functions

The objective function for maximizing collective value is defined as:

$$J_{value} = \textit{task value - task penalty - energy cost} \tag{3.13}$$

where *task value* denotes the value achieved from processing tasks and *penalty* compensates for failing to deliver a task before its expiration time. *Energy cost* is the marginal cost of data transmission using a link known to federates and the auctioneer. The value-maximizing problem is defined as:

$$\begin{aligned} &\text{find:} && x_{process},\ x_{trans},\ x_{resolve},\ x_{store},\ x_{read} \\ &\text{maximize:} && J_{value} \end{aligned} \tag{3.14}$$

subject to several technical constraints: 1) capacity of link for data transmission, 2) capacity of data storage on elements, 3) end-to-end connectivity of paths, and 4) expiration time for tasks on storage. The solution to the above optimization function finds $x_{process}$ for processing a task, x_{store} for storing a task, x_{read} for reading a task from storage, x_{trans} for transmitting task data through an inter-element link, and $x_{resolve}$ for resolving a task on an element. Appendix B gives detailed definitions of target variables and Appendix B.1 defines detailed equations and constraints for Eq.3.14.

This model maximizes the value of processed tasks and minimizes their delivery path cost. A *path* is a sequential set of elements (and links) that connects a source to a destination and is discussed later (see Sec 3.5). Maximizing the value of tasks ensures that federates can't pick-up other available tasks outside this operational mechanism (untruthful behavior) for a higher value as no additional task with feasible delivery path and cost exists. Minimizing path cost ensures stable solution as federates don't have incentive to reroute their delivery path for a lower cost.

$$\begin{aligned} &\mathbf{T}^* \leftarrow \arg\max_{tasks} \text{(federation value)} \\ &\mathbf{P}^* \leftarrow \arg\min_{links} \text{(path cost)} \end{aligned} \tag{3.15}$$

where $\mathbf{T}^*$ is the set of tasks that maximizes the total value for federation, and $\mathbf{P}^*$ is the set of paths that minimizes the delivering cost for those tasks.

Maximizing collective value is desirable for an operational mechanism but does not guarantee a routing with lowest cost for all federates. Minimizing routing cost is important for two reasons: 1) the solution may not be accepted by a federate when alternative paths with lower cost exist, 2) the solution does not reflect the actual value of operational model for federates. Accordingly, a second objective function and optimization problem minimizes the cost of inter-federate communication for a given set of processed tasks:

$$\begin{aligned} &\text{find:} && x_{trans},\ x_{resolve} \\ &\text{given:} && x^{@}_{read},\ x^{@}_{store},\ x^{@}_{process} \\ &\text{minimize:} && [\textit{communication cost} + \textit{rerouting failures}] \end{aligned} \tag{3.16}$$

for which at least one solution (to Eq.3.14) is guaranteed. In the latter equation, *communication cost* expresses inter-federate cost functions and *rerouting failure* (the number of failed tasks) ensures all processed tasks are delivered in the new routing. Finding the target variables (trivially) gives a solution to the operational model which is interpreted as a set of paths to deliver tasks, namely a *path bundle*. The above problem is subject to technical constraints presented for Eq.3.14 and the detailed formulations are presented in Appendix B.2. Assuming $\mathbf{t} = \{t_n\}$ as operational time steps, an *operational run* in Algorithm **3-I** summarizes the operational model.

Algorithm 3-I: operational run at time t.

1: **sub-steps:** find the sub-steps of time t: $\mathbf{s}_t$

2: **contextual:** receive from contextual model: existing links L, available tasks $\mathbf{T}_a$

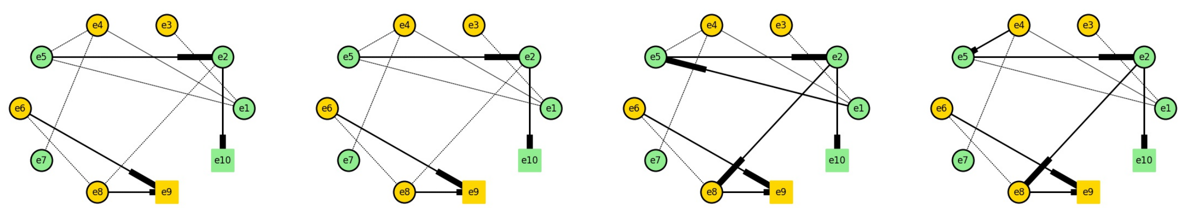
3: **federates:** receive information on elements E, data function $\mathcal{D}$, *capacity* function, destination set: $\mathbb{D}$, and cost functions ζ from federates

4: **value:** maximize federation value and find ($\mathbf{T}^* : x_{process}, x_{resolve}, x_{store}$) using Eq. 3.14

cost: minimize path cost and find ($\mathbf{P}^* : x_{trans}$) using Eq. 3.16
for $s \in \mathbf{s}_t$: **do**
  for $T \in \mathbf{T}^*$: **do**
    for $l \in L$: **do**
      if $s = \min(\mathbf{s}_t) \& x_{process}(T, t)$: **then**
        process: $I_{process}(T, e)$
      end if
      if $x_{trans}(T, l, s)$: **then**
        transmit: $I_{trans}(T, l, s)$
      end if
    end for
    for $e \in E$: **do**
      if $x_{resolve}(T, e, s)$: **then**
        resolve: $I_{resolve}(T, e, s)$
      end if
    end for
  end for
end for

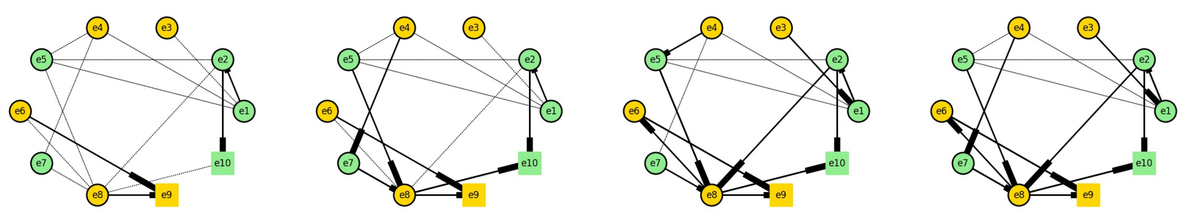
In a network with static topology or known dynamic topology of time steps (e.g. periodic topology in satellite systems), the operational model can also find paths in future time steps. Nonetheless, the network topology of L for n time steps might not be known in some networks, e.g. a robotic team or device-to-device (D2D) wireless networks. To address this issue in operational model, the networked model should summarize the value of all future topologies in a storage opportunity value (similar to *SP* in Eq. 3.4 with complementary meaning). In other words, a probable path to delivery in future is captured by an opportunistic value of storage penalty at current time in networks with unknown future topology. Then, a solution to the operational model uses x_{store} to reflect routing in later time steps.

As was mentioned in Sec. 3.3.2, the intuition for sub-steps is to create linear representation of combinatorial routing. With these binary decision variables (efficient task process, link usage, storage, and resolving), efficient set of paths to deliver a selected set of tasks (*path bundle*) is trivial and achievable in linear time (iterate over links and reconstruct paths from chained links). Nonetheless, the maximum federation value is not guarantied to deliver the tasks with the lowest available cost even for a single task delivery. The latter concern is important for two reasons: first, the solution would not be accepted by federates as they can switch to other paths with lower cost. Second, I don't realize the value of federation for a federate with a certain cost function.



(a) (b) (c) (d)

Fig. 3.4: The solutions for efficient routing in federated networks given cost function with edge density equal to 25% of possible edges (11 edges): a) $c_f > 1$: this case blocks all of the inter-federate communication since its cost greater than maximum task processing value. b) $c_f = 0.6$: a federates charge almost 60% of a task value for inter-federate communication, c) $c_f = 0.4$, d) $c_f = \epsilon$: the minimum link cost which corresponds to centralized solution with collective-efficient solution.



(a) (b) (c) (d)

Fig. 3.5: Optimized solution for federated networks given cost function with edge density equal to 33% of possible edges (15 edges): a) $c_f > 1$, b) $c_f = 0.6$, c) $c_f = 0.4$, d) $c_f = \epsilon$.

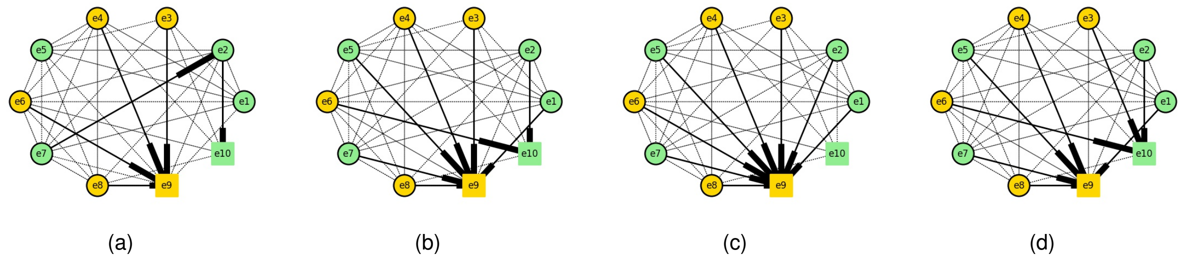
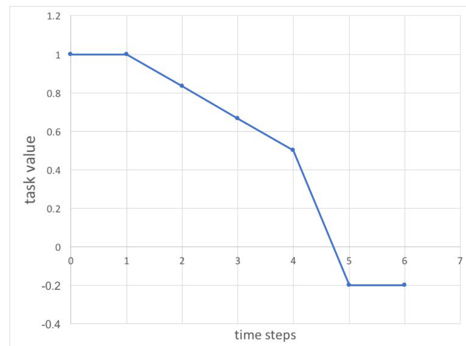


Fig. 3.6: Optimized solution for federated networks given cost function with edge density equal to 80% of possible edges (35 edges): a) $c_f > 1$, b) $c_f = 0.6$, c) $c_f = 0.4$, d) $c_f = \epsilon$.

Fig. 3.7: Task value function with highest value of 1 and diminishing value until expiration time of 5 elapsed time steps (since tasks pick- up). Penalty of 0.2 for failure to deliver applies after expiration time step.



3.4.2 Routing Case

This application case simulates a federation of task processing with ten elements: two destinations and eight potential sources. In this chapter, all prices and values are normalized between 0 and 1, relative to maximum task value. The minimum cost is assumed $\epsilon = 0.01$ and the operational model is run for different inter-federate cost functions (ϵ, 0.4, 0.6, and > 1). For instance, $c_f > \epsilon$ and $c_f > 1$ result in maximum and minimum inter-federated resource exchange respectively.

Another dimension for inter-federate exchanges is created by link density in a network. Higher density means higher probability for existing an inter-federate link but lower probability for dependance on a link for routing. Fig. 3.4, Fig. 3.5 and Fig. 3.6 show topologies with edge-densities of 25, 34 and 80%. For operational run, task values, task data size, and a link's data capacities are assumed to be equal to 1, 1, and 2. The results from the MILP model introduced by Eq. 3.14 is shown using directed arrows in the network where dotted lines show available links. The federates are distinguished by gold and green colors, source elements with available tasks are represented by circle nodes with border lines and destination elements are shown by square nodes.

Table 3.1 shows relative values by federates in one operational run in cases with various edge densities represented in Figures 3.4 to 3.6. Maximum value (last column) shows the maximum value relative to a task value collected by a federate and is associated with number(s) 1 on the same row. Table 3.2 shows the relative collective values to the maximum value collected by all federates.

Case	Federate	(a) $c_f > 1$	(b) $c_f = 0.6$	(c) $c_f = 0.4$	(d) $c_f = \epsilon$	max value
$Case_1$	F_1	0.91	0.91	1	0.9	2.15
$Case_1$	F_2	0.67	0.67	0.93	1	2.96
$Case_2$	F_1	0.45	0.76	1	0.9	4.32
$Case_2$	F_2	0.5	0.9	0.89	1	3.91
$Case_3$	F_1	0.49	0.7	0.6	1	3.97
$Case_3$	F_2	0.71	0.93	1	0.71	5.52

Table 3.1: Federated relative values for cases in Figures 3.4 to 3.6 [Fn: federate n].

3.5 Mechanism Design

This section proposes a mechanism for pricing resources among federates to increase their collective value. To simplify notations, I represent the solution to Eq.3.14 and Eq.3.16 by bundle of paths $\mathbf{P}$ and a *bundle function*:

$$\mathcal{B} : \zeta \xrightarrow{Eq.\ 3.14\ \&\ 3.16} \mathbf{P} \tag{3.17}$$

Case	(a) $c_f > 1$	(b) $c_f = 0.6$	(c) $c_f = 0.4$	(d) $c_f = \epsilon$	max value
$Case_1$	0.8	0.8	1	1	4.92
$Case_2$	0.5	0.87	1	1	7.83
$Case_3$	0.75	1	1	1	7.92

Table 3.2: Relative collective values for three cases in Table 3.1

where the input is the cost function defined for storage and communication resources. A collective-efficient solution can be achieved using the latter function by assuming a minimum cost function for links. In other words, when federates share resources with minimum cost, a operational solution is equal to the optimal solution for a federation.

Definition 3.5.1. *Minimum cost function* is defined for links and paths as:

$$\zeta_0 : l_{ikt} \rightarrow \begin{cases} \epsilon & i \neq k \\ SP & otherwise \end{cases} \tag{3.18}$$

The collective (efficient), federated (realistic), and independent (pessimist) solutions to operational model are defined as:

Definition 3.5.2. *Federate-efficient solution (FES)*: for bundle function $\mathcal{B}$ and cost function ζ is:

$$\mathbf{P}_{\zeta} = \mathcal{B}(\zeta)$$

Definition 3.5.3. *Collective-efficient solution (CES)*: for bundle function $\mathcal{B}$ and cost function ζ_0 is:

$$\mathbf{P}_{\zeta_0} = \mathcal{B}(\zeta_0)$$

.

Definition 3.5.4. *Independent-efficient solution (IES)*: for bundle function $\mathcal{B}$ and cost function ζ_I is:

$$\mathbf{P}_{\zeta_I} = \mathcal{B}(\zeta_I > taskvalue)$$

.

3.5.1 Objective Function

The proposed mechanism in this section maximizes federation value and ensures expected payoff for each federate in FES. The general form of the pricing mechanisms combines Eq.3.14, Eq.3.16, prices, and a value constraint:

$$
\begin{aligned}
&\text{find:} \quad \zeta',\ x_{trans},\ x_{resolve} \\
&\text{given:} \quad x^{@}_{read},\ x^{@}_{store},\ x^{@}_{process} \\
&\text{maximize} \quad J_{value} && (3.19) \\
&\text{subject to:} \\
&\quad \text{Constraints for Eq.3.14} \\
&\quad \mathcal{V}_{fed}(f,\zeta') \geq \mathcal{V}_{fed}(f,\zeta) \qquad \forall f \in F && (3.20)
\end{aligned}
$$

where ζ' is the alternative pricing for sharing and ζ is the current cost function provided by federates. Equation 3.20 ensures federates achieve higher value than the FES'.

The search space for data transmission on links (x_{trans}) in above equation is limited to FES and CES path bundles ($\{l : \exists P \in \mathbf{P}_{\zeta_0} \cup \mathbf{P}_{\zeta}, l \in P\}$), allowing the above formulation to be disaggregated using a non-linear method such as sequential least squares programming (SLSQP) [141] with separate operational and pricing functions. Using the latter method and parameters x_{trans} and $x_{resolve}$ from Eq.3.16, I find $\mathbf{P}_{\zeta_0}$ and $\mathbf{P}_{\zeta}$ and all possible bundles using permutations of their links to solve:

$$
\begin{aligned}
&\text{find:} \quad \zeta^{*} \\
&\text{given:} \quad x^{@}_{read},\ x^{@}_{store},\ x^{@}_{process},\ x^{@}_{trans},\ x^{@}_{resolve} \\
&\text{maximize:} \quad J_{value} && (3.21)
\end{aligned}
$$

subject to incentive constraint in Eq.3.20. Algorithm `3-II` illustrates the proposed mechanism:

Algorithm 3-II: Pricing Mechanism

1: **federated:** find FES: $\mathbf{P}_\zeta$.

2: **values:** find federated values in FES: $V_f = \mathcal{V}_{fed}(f, \zeta), \forall f \in \mathbf{F}$.

3: **collective:** find CES: $\mathbf{P}_{\zeta 0}$.

4: **path bundles:** find all path bundles using links in $\{l : \exists P \in \mathbf{P}_{\zeta_0} \cup \mathbf{P}_\zeta, l \in P\}$: $\mathbf{B} = \{\mathbf{P}_i\}$

5: **sort bundles:** sort bundles descending with collective values: $\mathbf{B} = sort_{dec:\mathcal{V}(\mathbf{B})}(\mathbf{B})$

6: **for** $\mathbf{P} \in \mathbf{B}$ **do**

7:     *new prices:* find resource prices $\mathcal{L}^*$ by solving optimization in Eq.3.21

8:     *new values:* find federated values for new pricing $V_f^* = \mathcal{V}_{fed}(f, \mathcal{L}^*) \geq V_f, \forall f \in \mathbf{F}$

9:     **if** $V_f^* \geq V_f, \forall f \in \mathbf{F}$ **then**

10:         break

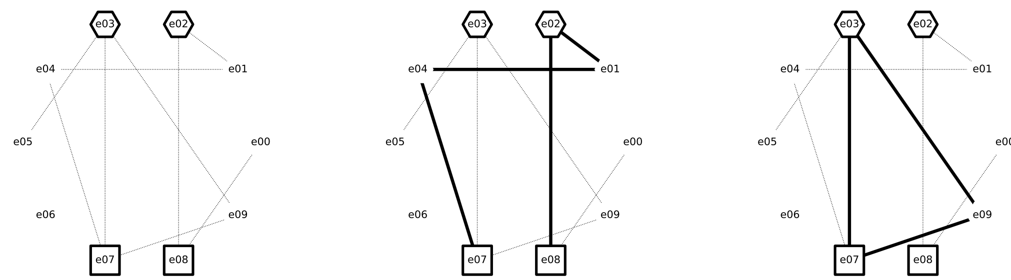
11:     **end if**

12: **end for**

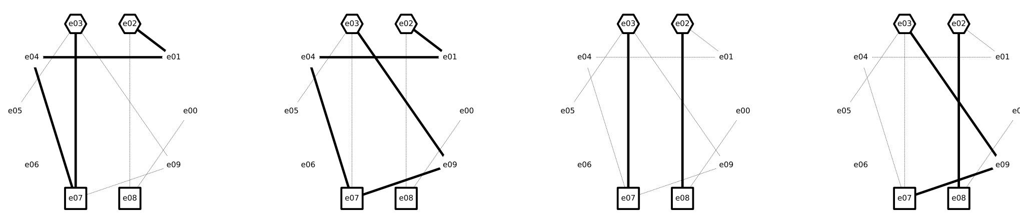
13: **suggest:** prices $\mathcal{L}^*$ to federates

3.5.2 Computational Cost and Limitations

The computational complexity of Algorithm II (A_{II}) relies on solving a MILP (lines 1 and 3 of A_{II}), sorting and searching all combinatorics of path bundles (lines 5 and 6 of A_{II}), and SLSQP optimization. However, bundles with the highest collective value likely satisfy the value constraint in Eq.3.20, implying that sorting and searching minimally contribute to computational cost. Therefore, A_{II} requires minimal iterations of MILP operational runs because it finds an applicable prices to the scheduling/routing solution with highest expected value. To analyze the computational cost associated with an operational run, we divide the routing problem into: finding paths from sources to destinations and bundling non-conflicting paths from multiple sources in a network. Figure 3.8 shows four routing bundles for a network with 10 elements and two sources. By increasing the network size, Fig. 3.9 shows the computational cost of searching paths (from sources) and searching for viable path bundles. Accordingly to the this plot, finding path bundle is significantly costlier than finding paths in larger networks.



(a) a topology (b) source 1 paths (c) source 2 paths



(d) bundle 1 (e) bundle 2 (f) bundle 3 (g) bundle 4

Fig. 3.8: A path finding and routing solution to a sample network with two sources e02 and e03 and two destinations e07 and e08: (a) shows initial topology with links between elements, (b) shows two paths from e02 to destinations e07 and e08 (c) shows two paths from e03 to destination e07 and (d)-(g) show four possible path-bundles for routing task data a destination.

The proposed mechanism is limited by some assumptions introduced in Sec. 3.2.1. First, the auctioneer is a trusted entity with knowledge of strategic information such as availability and cost of computational resources. However, federates may not disclose truthful information as a federate might allocate its resources to mission unknown to the auctioneer. Second, we don't consider corrupted behavior by an auctioneer including dual resource prices for buyers versus sellers and collusion with some federates. Last, the computational tasks are assumed known by a value func-

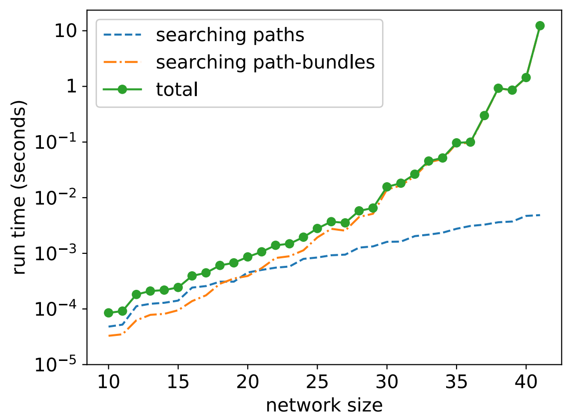


Fig. 3.9: Averaged run time associated with finding paths and path-bundles illustrated in Fig. 3.8 depending on the network size. The sources and destinations share $\%25$ of vertices combined and edge density ranges from $\%15$ to $\%6$ for 10 to 40 vertices. Results from network topologies created from 300 random seeds are used to compute averaged values for a network size.

tion although those tasks are dynamic in availability and expiration.

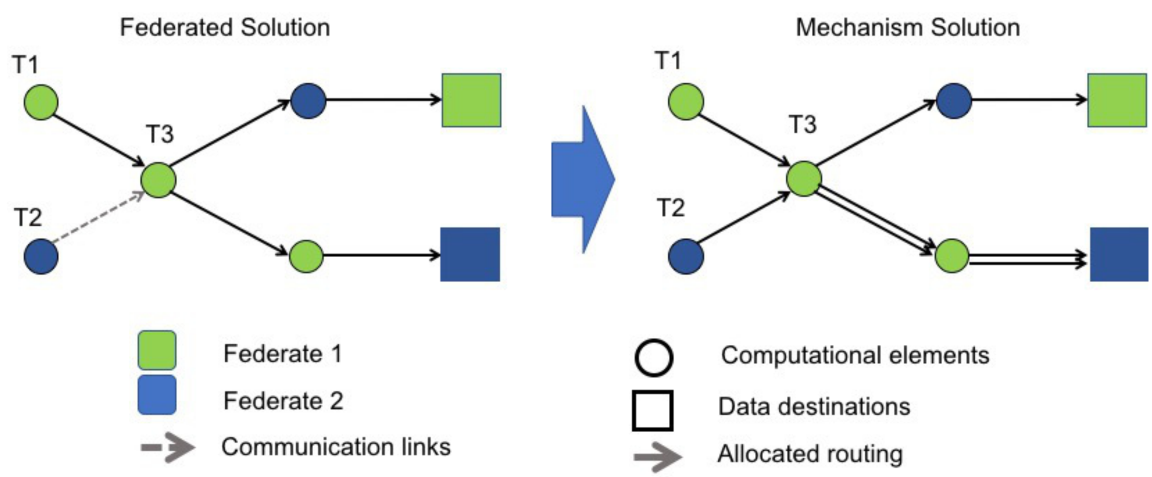


Fig. 3.10: A sample element network of computational resources including two federates where tasks are available to three elements and communication links connect all elements to destination nodes with $capacity(l) = 2 \times tasksize$. The left figure shows *federated* solution while the right figure shows the solution based on the *mechanism* introduced in Eq.3.19 with new prices, routing and values for federates. The proposed mechanism enables processing all tasks $\{T_1, T_2, T_3\}$ versus only two $\{T_1, T_3\}$ in the initial solution.

3.5.3 Rerouting Case I

The proposed mechanism in Eq.3.19 solves resource allocation to find new routing solutions and offer prices for resources. This section illustrates how the mechanism affects two federates (f_1 and f_2) using an example with a routing solution. For simplicity, all prices and values are relative to maximum task value and we assume energy cost of communication $\epsilon = 0.01$.

Figure 3.10 shows a routing solution to initial bids by f_1 and f_2 (left) and a rerouting solution by the auctioneer (right) in a simple topology. In this example, a federate processes tasks, bids to share its links (edges) with the other federate and delivers its data using available links (with feasible cost) to destination. Assuming 0.5 and 0.01 as bids by federates, tasks T_1 and T_3 can be processed and delivered to destination because the minimum cost of delivering T_2 to a destination is equal to $2 * 0.5 + 0.01$ is higher than a task value. In this case, the suggested price by the auctioneer is 0.494 for shared links by both federates. Accordingly, the auctioneer finds a new solution that includes processing T_2 by f_2, which results in value 0.970 for f_2. Assuming 0.6 and 0.4 for bids, the suggested prices increases to 0.494 and 0.884 and new values for federates are 1.170 and 1.750. For higher bids, e.g. 0.8, the auctioneer suggests 0.494 and 0.979 for prices and resulting values for federates are 0.980 and 1.940, compared to 0.370 and 1.580 without using the mechanism. In this network, the rerouting solution by the auctioneer supports different bidding conditions by federates while the prices by auctioneer ensure equal or higher values for both federates.

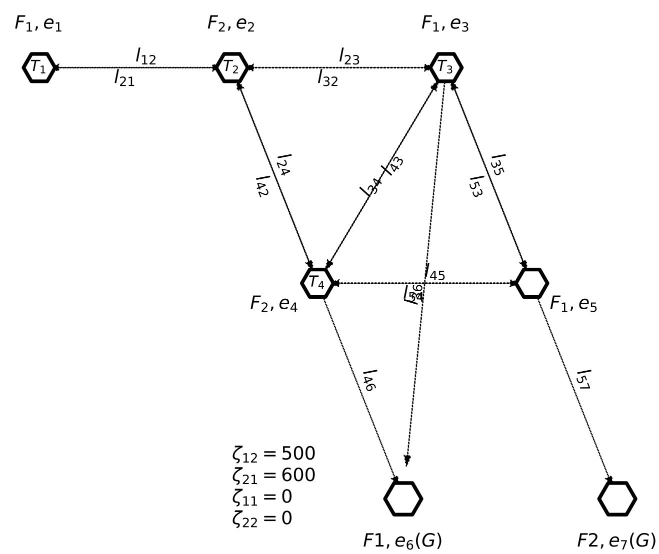


Fig. 3.11: A sample federated network with elements, tasks, federates, links, and cost function. Link costs (400 and 600) are relative to maximum task value (1000) during this chapter.

3.5.4 Rerouting Case II

Similar to *Case I*, using a different network topology with more complex routing solution, the FES for task processing fails to process and deliver all tasks processed in CES. The alternative routing is shown in last column of Table 3.3 with alternative pricing that ensures expected payoff for federates while maximizing value. The final cost function and federate payoff are compared in Table 3.4.

Task	FES paths	CES paths
T_1	None	e_1, e_2, e_4, e_5, e_7
T_2	e_2, e_4, e_6	e_2, e_4, e_6
T_3	e_3, e_6	e_3, e_6
T_4	e_4, e_6	e_4, e_6

Table 3.3: Tasks: Averaged path cost and length

Cost and Value	FES	Pricing mechanism
Link cost	F1: 0.6 F2: 0.5	0.577 0.282
Federate value	F1: 2.2 F2: 0.8	2.308 1.695
Relative Collective Value	3.003	4.004

Table 3.4: Federated Link Costs and Values

3.6 Federated Satellite Model

This section develops an FSS application, selects design configurations as application cases and proposes an adaptive cost function (in Eq.3.4). An orbital model of a federated TN of satellites is based on concepts introduced in Sec. 3.3 and a few of conceptual frameworks introduced by Grogan et. al. in [54] and [55]. In an orbital model, elements are either a satellite or a ground station and the relative location of satellites and the topology of networked elements are periodic. The location of an element is retrieved using a *propagation function* at each time step:

$$\mathcal{L} : (e \in E, t \in \mathbb{Z}^+) \rightarrow (sector \in \mathbf{S}, altitude \in \mathbf{A})$$

where $\mathbf{S} = \{1, 2, ..., 6\}$ shows two-dimensional sectors of 60-degree slices ($\frac{\pi}{3}$ radians) and $\mathbf{A} = \{0, 1, 2, 3\}$ shows possible altitudes associated with surface (SUR), low earth orbit (LEO), medium earth orbit (MEO), and geosynchronous earth orbit (GEO). For each radial sector, computational tasks such as visual imaging (VIS) and synthetic aperture radar (SAR) are available to LEO and MEO satellites above the same sector.

A communication link (see Eq. 3.4) exists between two elements depending on their type and location. A communication link is feasible for:

1. **Inter-satellite link**: satellites located at adjacent sectors or the same sector, e.g. in sectors $(1 \to 2)$, sectors $(1 \to 6)$, or sectors $(4 \to 4)$.

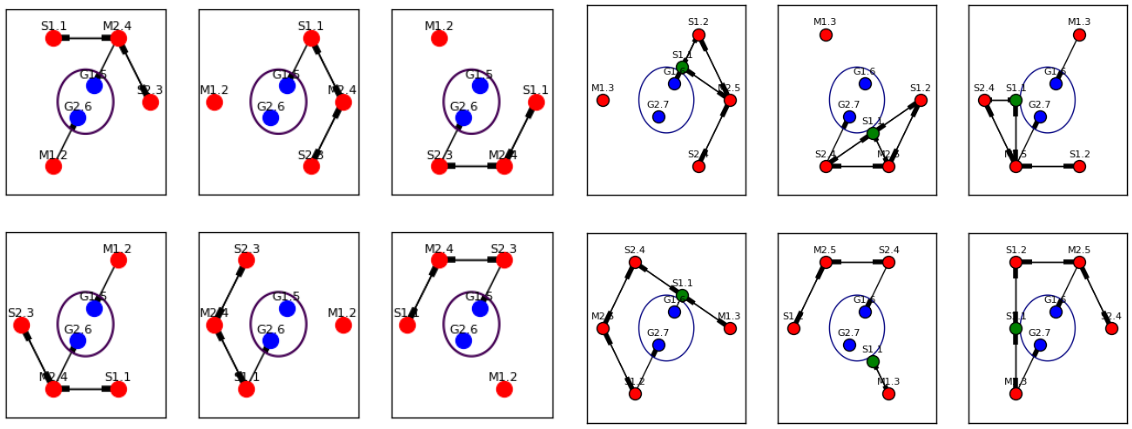
2. **Down-link and up-link**: a satellite and a ground station are located at the same sector.

In federated networks, the state function is defined by network topology ($\mathcal{S}$ in Sec. 3.3.1). From the above condition, I know location of elements determine feasible links and network topology. Then, since location of satellites are periodic, I also have periodic network topology and periodic states. In the structural model (Sec. 3.3.1), value and data functions are exogenous to the model while communication cost (c_f) and storage penalty (*SP*) (Eq. 3.4) are endogenous variables to the model. Storage penalty and link costs are different as *SP* is an opportunity cost estimated using federate's historical information and c_f is a pricing decision by a federate.

The following sections select network topologies for application cases, formulate a probabilistic model to estimate *SP*, and propose an adaptive model to offer communication cost (c_f).

3.6.1 Design Selection

A design specifies initial conditions for type and location of satellite and ground stations. Selected designs aim to: 1) develop an application for a federated network, 2) maximize the opportunities for exchanging resources across federation, and 3) explore distributed (independent) functionality for



(a) (b)

Fig. 3.12: Networked model of orbital satellite model: a) periodic orbital model with two ground stations and four medium orbit satellites (MEO), b) periodic orbital model with two ground stations, four satellites on medium altitude (MEO) and one satellite on low orbit altitude (LEO).

each federate.

For the first goal, a networked model of orbital federation at each time step is suggested. Fig. 3.12 shows the networked model of two orbital designs using the location of elements and the communication links. To achieve a maximum task processing across a federation, a balance holds for number of ground stations and the number of satellites. The cumulative values of federates in CES (e.g. summed values after 240 time steps) shows an estimation of a design's performance regarding task processing. Assuming a cost for each design, chosen designs are selected from *Pareto-optimal* line of *cost versus value*. Design cost is calculated based on the number of elements, type of each element, and number of communication links with adjacent elements. In the cost function, relative cost of LEO, MEO, GEO, and GS are 1, 2, 5, 10 and cost of each additional communication modules is assumed as 0.2 and 0.4 for LEO and MEO, then the total cost is normalized by maximum cost of 50. Figure 3.13 shows a design tradespace 472 designs computed under independent and centralized operational models. In this, the top line shows non-dominated designs to maximize CES value and minimize cost. In this figure, fewer than 29 unique designs are

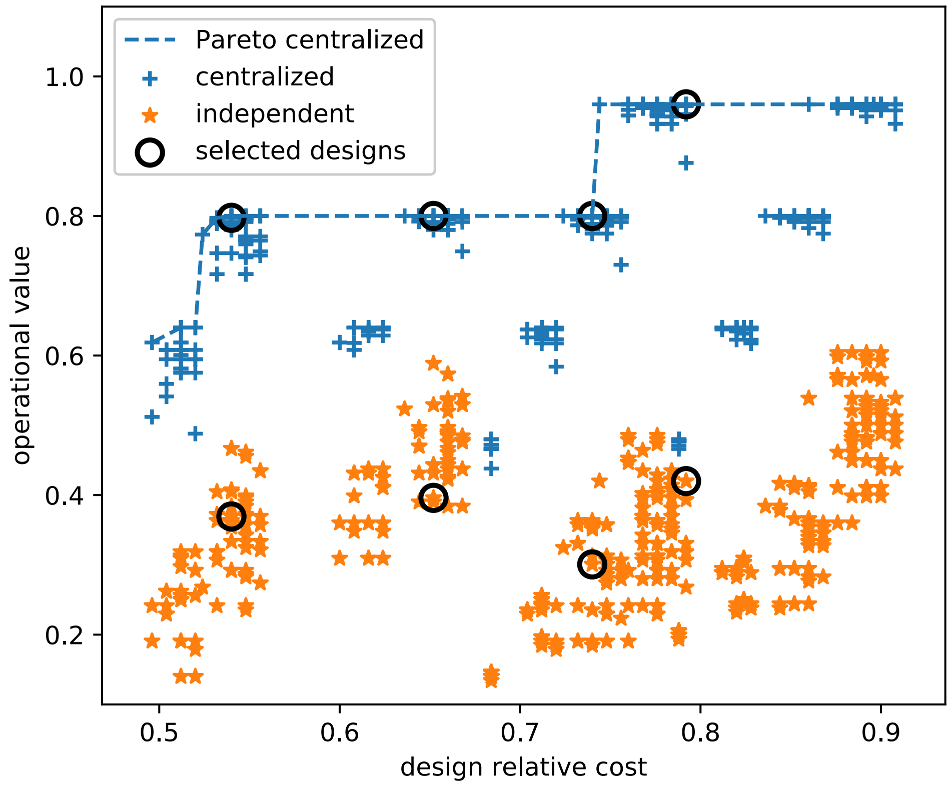


Fig. 3.13: The relative operational value of *centralized* and *independent* solutions corresponding to relative cost of design with Pareto-line of designs for 472 permutations of types and relative location of satellites and ground stations. The design cost is normalized by maximum value of 50 with 1, 2, 5, and 10 for relative costs of elements. The black circles show the four of the selected designs drawn in Fig. 3.14. The collective value is normalized by $60k$ tasks.

located on the Pareto line.

For the number of federates, two or three federates are chosen to keep results tractable. Accordingly, three or four elements are assigned to each federate and in order to keep the distributed functionality for federates, the number of ground stations are equal to the number of federates. Fig. 3.14a–3.14e select five designs from the tradespace where numbered hexagons are elements with a sector (radial) and altitude (radius) and colors show the federates. In these figures, four out of five designs are selected from the Pareto line in Fig. 3.13 and one design (`D. IV`) is created by adding a GEO satellite to `D. III` for considering its effect on balanced operation of federates and the mechanism.

Design	Federates	Stations	Satellites
D. I	F1 F2	SUR1 SUR4	MEO1, MEO4, LEO1 MEO5, LEO2
D. II	F1 F2	SUR1 SUR4	MEO1, MEO4, LEO1 GEO4, MEO5, LEO2
D. III	F1 F2 F3	SUR1 SUR3 SUR5	MEO1, LEO2 MEO3, MEO5 MEO6
D. IV	F1 F2 F3	SUR1 SUR3 SUR5	MEO1, MEO2 MEO3, MEO5 GEO5, MEO6
D. V	F1 F2 F3	SUR1 SUR3 SUR5	MEO1, LEO1 MEO2, LEO2 MEO3, LEO3

Table 3.5: The application design cases for an orbital federated satellite system. Element location: $[altitudes : SUR = 1, LEO = 2, MEO = 3, GEO = 4]$ + $[sector]$, e.g. MEO1: [MEO] + [1].

3.6.2 Storage Penalty

In FSS model, the network consists of spatial (communication) and temporal (storage) links. In the cost function introduced by Eq. 3.4, *SP* is the expected cost of storing versus delivering data of an task. Realistic estimation of *SP* is essential for a federate's decision to pick up and deliver tasks. In technical terms, a federate has to estimate *SP* to find the shortest path for tasks. For instance, biased estimation of this value toward lower cost incentivizes a federate to store data at the cost of missing better pick-up opportunities at later time steps; and, the opposite case incentivizes it to deliver data when paths with lower overall cost exist in future. This section introduces a probabilistic method to estimate storage penalty.

SP depends on: 1) task delivery value, 2) satellite state, e.g. storage capacity, location, 3) expected task pick-up opportunities, 4) network topology in future and 5) link cost function (ζ) by other federates. I estimate storage penalty by estimating expected net value of tasks being delivered (task value minus path cost). For probabilistic cost estimation, a federate tracks and uses the historical tasks of a state (network topology). The first intuition in this method is to use the net value of historical tasks for next time step to calculate expected value of *SP*. In periodic federated

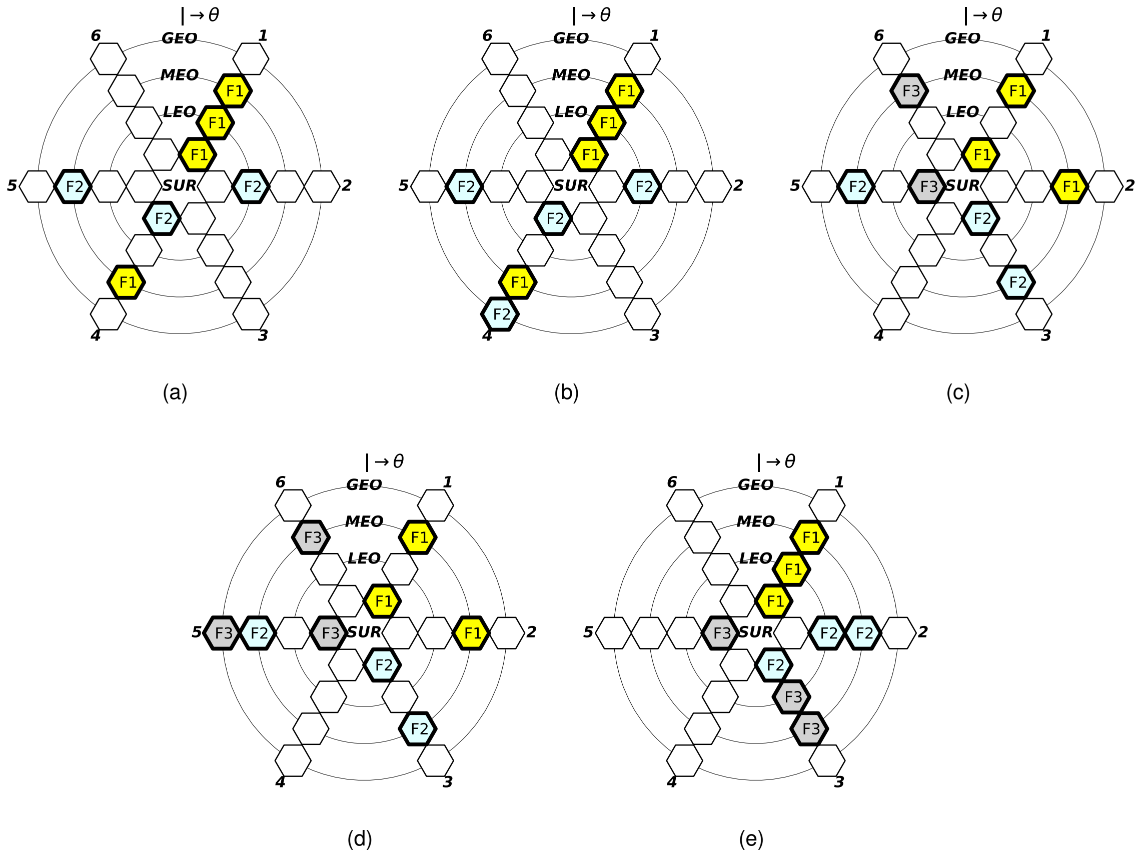


Fig. 3.14: Selected Pareto-optimal designs for orbital FSS detailed in Fig. 3.14: (a) Design I: two federates, two ground stations, five satellites, (b) Design II: two federates, two ground stations, six satellites, (c) Design III: three federates, three ground stations, five satellites, (d) Design IV: three federates, three ground stations, six satellites, (e) Design V: three federates, three ground stations, six satellites.

networks such as orbital FSS, the network topology is repetitive when a limited number of states can capture the periodic aspect of a federation. Accordingly, we define element state by its location in model, e.g. satellite's sector. In addition, I expect that network topology and historical data give most valuable information for estimating path cost. This suggests that the state-based model is appropriate to estimate *SP*. $0 \leq p \leq 1$ reflects the probability of an available task at each time step and for each satellite. In our model, I assume that computational tasks are always available (abundant) at each sector to be picked up. However, p is still useful to represent cases where no feasible path exist for some tasks, for instance, when some tasks cannot be delivered until being expired or no path with a feasible cost exist.

Internal storage is a resource available to elements and is associated with the opportunity cost of storage. For instance, maximum storage limits an element's opportunity to pick-up tasks at next time step without the current tasks being delivered. In this case, missing a delivery opportunity (storage) reflects both missing task pick up and missing task delivery. Instead, with enough storage for multiple tasks, the penalty is lower as the storage opportunity only reflects missing task delivery. To include this parameter, two options are: first, to update states with the internal storage, or, to normalize *SP* with internal storage. The second option is prefered because network topology has valuable information by being linked directly to states. I divide the estimation by its maximum available memory at next time step.

The intuition behind this division is from a simple mathematical model of a storage unit, a task pick up opportunity, and a task delivery opportunity. In this model, if a task is available and memory is full, the task is missed, only one task can be delivered at each time step, and task delivery and task availability are stochastic variables. Appendix A discusses a closed-form solution for storage penalty for memory sizes using an Markov decision process (MDP) model.

Finally, delivery time of tasks has negative effect on *SP*. A task that is delivered earlier on average, e.g. after one time step, increases the *SP* as it increases cost of missing one time step. In this model, instead of expected task benefit, I estimate storage penalty for one time step by dividing the sum of historical task values by the sum of their delivery time. This mechanism follows two objectives: the sum of net values is first divided by the number of tasks and expected delivery time.

To retrieve the states' history on tasks, a function $\mathcal{H}$ links an element to its task history at the next time step:

$$\mathcal{H} : e \xrightarrow{\mathcal{L}(e,t+1).sector} \{(T_i, P_i)\}$$

where t is current time, T is a historical task linked to next state and P is its corresponding path.

The probabilistic estimation of storage penalty for task T' is:

$$\mathcal{SP}(e,T') = \frac{p \times \sum_{(T,P)\in\mathcal{H}} [\mathcal{V}(T, P.time) - T.size \times \zeta_c(P)]}{(capacity(e) - \mathcal{D}_e(e) + T'.size) \sum_{(T,P)\in\mathcal{H}} \Delta t} \tag{3.22}$$

where :

$$\left[\sum_{(T,P)\in\mathcal{H}} [\mathcal{V}(T, P.time) - T.size \times \zeta_c(P)]\right] :$$

is sum of historical profit for picked up tasks at next state

$[(capacity(e) - \mathcal{D}_e(e) + T'.size)]$: is normalizing factor for memory (1 or 2)

$\left[\sum_{(T,P)\in\mathcal{H}} \Delta t\right]$: is normalizing factor time as sum of historical task delivery times

where e is a potential element to pick-up task T', $\Delta t = P.time - T.init + 1$ is the calibrated time difference between task pick-up and task delivery. In this chapter, the above value is called *marginal SP*.

Cumulative value of a federation over multiple time steps shows the accuracy of our estimation accuracy because right decisions on task storage and delivery depends on accurate estimation of opportunity cost. I run the operational model to validate the estimation accuracy of our estimation in Eq. 3.22. For tractable results, lets start with simple task-value and communication cost functions (c_f in Eq. 3.4). The value function for tasks varies between 1 and 0.5 for each task, decreasing linearly in time (expiration time = 5 steps, penalty = -0.2). The communication cost is assumed to be fixed ($c_f = 0.6$). For benchmark values of *SP*, I compare fixed values for *SP* on the same operational model and select the values that achieve best results in overall: $SP = 0.4$ and $SP = 0.8$. The former is best for federated models with fewer communication opportunities while the latter achieves better results on more inter-coupled elements. Intuitively, it is similar to having more delivery opportunities by increasing the opportunity cost of storage. I further apply stochastic

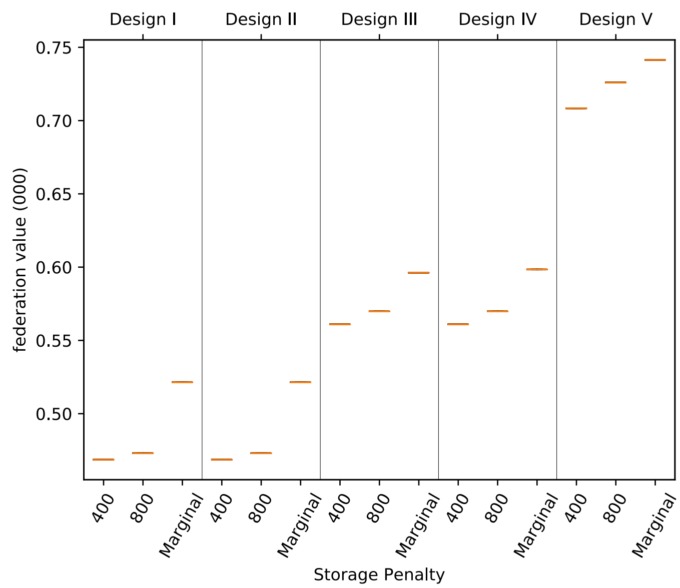


Fig. 3.15: **Deterministic**: federation value with deterministic communication cost $c_f = 0.6, f \in F$ for *marginal* case (see Eq. 3.22) vs fixed storage penalties $SP = 400(relative : 0.4)$ and $SP = 800(relative : 0.8)$ suggests higher collective value by using marginal *SP*. Designs are selected from Table 3.5. A higher value reflects more accurate estimation of opportunity cost of storage.

communication cost with similar expected value ($E[c_f] = 0.6$) to this operational model. In the stochastic model, versus the deterministic model, lower estimation $SP = 0.4$ achieves relatively better results.

Figure 3.15 draws federation value using *marginal SP* versus fixed costs. The *marginal* method achieves 4 to 10% improvement on average for the deterministic case. Fig. 3.16 shows the results for the stochastic model.

3.6.3 Communication Cost

For a link, communication cost belongs to its destination node. In the operational model, communication cost is assumed to be equal for all links shared by a federate at each time step. This section introduces an adaptive mechanism for realistic modeling of self-centric cost function.

A q-learning approach is applied to cost function where q-state represents a network topology

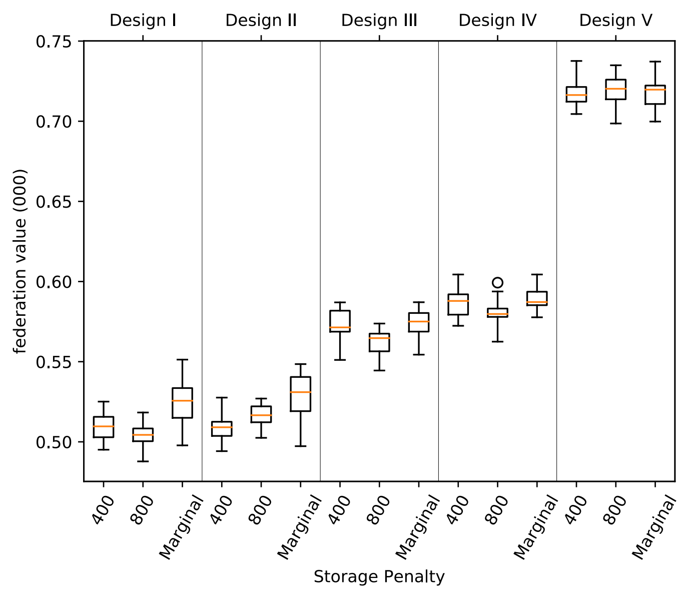
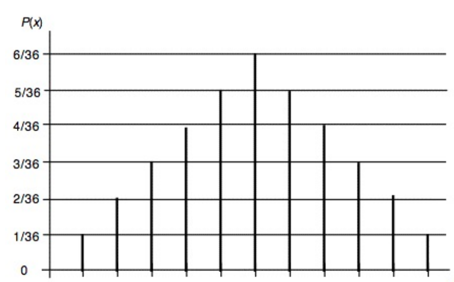


Fig. 3.16: **Stochastic**: federation value with stochastic communication cost $0 < c_f < 1.1, f \in F$ (see Fig. 3.17 and caption for Fig. 3.15)). The samples in a box include 30 seeds where the upper whisker extends to last datum less than `Q3+3*(Q3-Q1)`, lower whisker extends to last datum greater than `Q1-3*(Q3-Q1)`. Beyond the whiskers, data are considered outliers and are plotted as individual points. Standard deviation of samples range between 0.006 to 0.016.

Fig. 3.17: Communication cost random variable with triangular distribution for stochastic cost in Fig. 3.16 where lowest value (leftmost) with positive probability is $c_f = 0.1$ and highest (rightmost) value is $c_f = 1.1$.



and q-action represents the cost and general updating mechanism of:

$$Q(s_t, a_t) \leftarrow (1-\alpha)Q(s_t, a_t) + \alpha[R(t) + \gamma * \max_{\alpha} Q(s_{t+1}, a_{t+1})] \tag{3.23}$$

where s_t is the state, a_t is the action, and $R(t)$ is the reward at time step t, α is the learning factor between 0 and 1 (exploitation vs exploration), and γ is the discount rate between 0 and 1 in time which models discounted future rewards. $\alpha = 0$ means Q value is never updated and $\alpha = 0.9$ means that learning happens quickly. Also $\max_\alpha$ finds the most attainable reward in the next state following the current state. In a FSS, I select $\alpha = 0.8$ and $\gamma = 0.9$ with a random selection of actions with probability $r = 0.05$. In this model a state remembers network topology while an action represents the communication cost (c_f). More detailed explanation of the applied q-learning model is defined in Appendix. C.

For a realistic model, I need to resolve three compatibility issue between FSS and q-learning: 1) temporal distance between actions and reward in FSS (task pick up and tasks delivery) 2) interdependency between consecutive times steps in FSS, and 3) continuous cost function (action space). The first and second are resolved using discount effect on rewards and path-dependent effect (inertia) on actions. In other words, rewards are linked to closest actions with an discount rate, and a federate selects an action for multiple time step (e.g. 3) and changes it marginally (upward and downward). Both assumptions fit the realistic cases of pricing behavior by agents. For the third issue, I update Q-value using a Gaussian distance function among state-action pairs.

The cumulative values for federates are used to compare results. The operational model applies the *marginal SP* introduced in Sec. 3.6.2 in comparison to baseline cost functions: $c_f = \epsilon$, $c_f = 0.6$, $c_f > 1$, $c_f = triangular$ (see Fig. 3.17), where $c_f = \epsilon$ represents CES because zero communication cost implies that all resources are available to all federates without inter-federate cost, i.e., the federation operates as a monolithic design. $c_f = 0.6$ is fixed communication cost which implies

FES where communication cost accounts for up to 60 percent of task value on each path. Finally, $c_f > 1$ results in IES when no inter-federate communication exist due to link cost exceeding a task value. In this case, each federate works as a separate monolithic design.

Figures 3.19 to 3.22 show the collective values by federates before and after switching a cost function from baseline (CES, FES, random FES, and IES) to adaptive case of FES. With initial CES, the effect of switching to adaptive cost is positive for the federate with adaptive cost and negative for others. The intuition is the adaptive federate has access to the network's resources for free while obtain value for sharing it's own resources with other federates. With initial IES ($c_f > 1$), adaptive cost increases value for all federates as other federates have access to new resources while the adaptive federate also obtains additional value by sharing its unused resources (see Fig. 3.20). With initial FES case with fixed cost ($c_f = 0.6$), adaptive cost benefits the switching federate itself and slightly benefits other federates. Here, similar to IES, the adaptive federate finds the best cost function for sharing its resources which might also benefit other federates (see Fig. 3.21). Lastly, with initial random FES, all federates benefit from switching to adaptive cost function by one federate (see Fig. 3.22). In sum, switching to self-centric approach (aka adaptive cost function) is all but certain choice for an individual federate.

By switching cost strategy by all federates one by one, total value changes according to circles shown in Fig. 3.18. A circle shows federates with their cost strategy, a small black square shows an adaptive federate, and a red triangular shows a minimum-cost resource by a federate. The number of adaptive federates increases from left to right for each design. In IES, addition of an adaptive federate always improves the total revenue. In FES with fixed value ($c_f = 0.6$), the total revenue also increases by adding more adaptive federates. In CES, the adaptive cost function reduces the total value. In sum, self-centric behavior by federates shall reduce collective value in CES or FES with relatively high level of sharing resources.

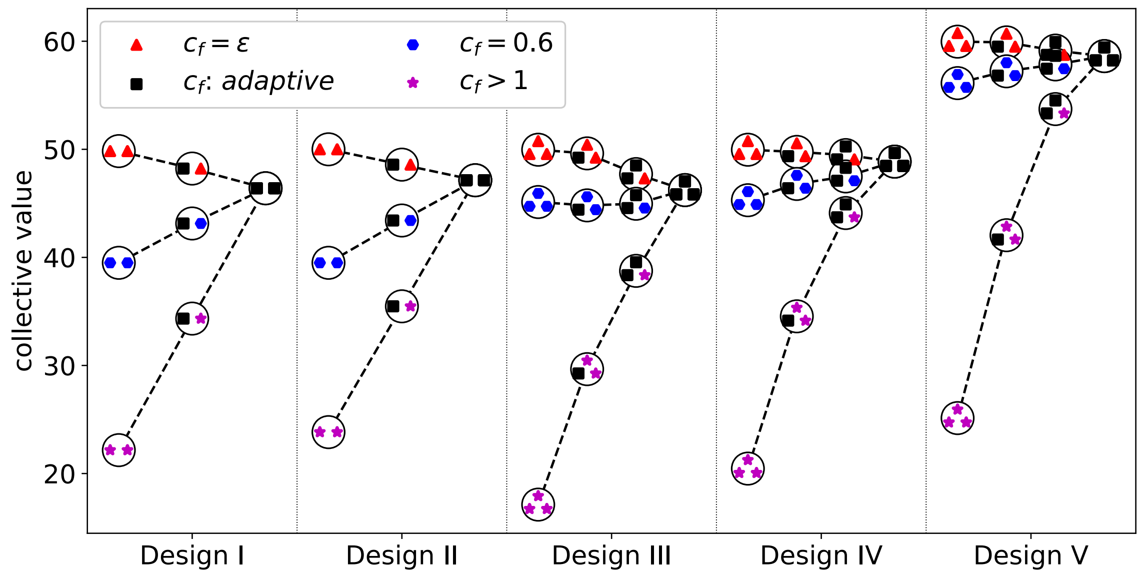


Fig. 3.18: Federation value by switching cost function to adaptive cost from left to right in each design where $c_f = \epsilon$ represents CES, $c_f = 0.6$ is FES, and $c_f > 1$ represents IES. The values are averaged values during 10000 time steps across 100 seeds for every circle.

Fig. 3.19:
Average federate value in CES ($c_f = \epsilon \approx 0$) when one federate switches to adaptive cost function: a) **adaptive federate**: value significantly increases the adaptive federate, b)**non-adaptive federate(s)**: value decreases for other federates. The values are averaged values during 10000 time steps across 100 seeds for every circle.

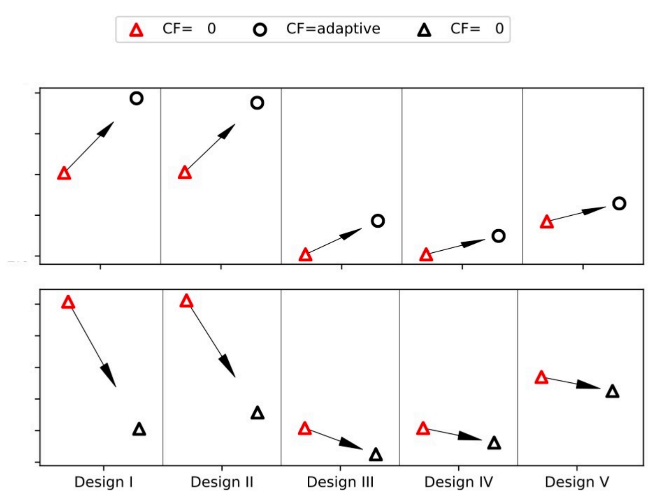


Fig. 3.20:
Average federate value in IES ($c_f > 1000\ [relative : 1]$)) when one federate switches to adaptive cost function: a) **adaptive federate**: federate value significantly increases for the federate with adaptive strategy, b)**non-adaptive federate(s)**: federate value significantly increases for other federates. The values are averaged values during 10000 time steps across 100 seeds for every circle.

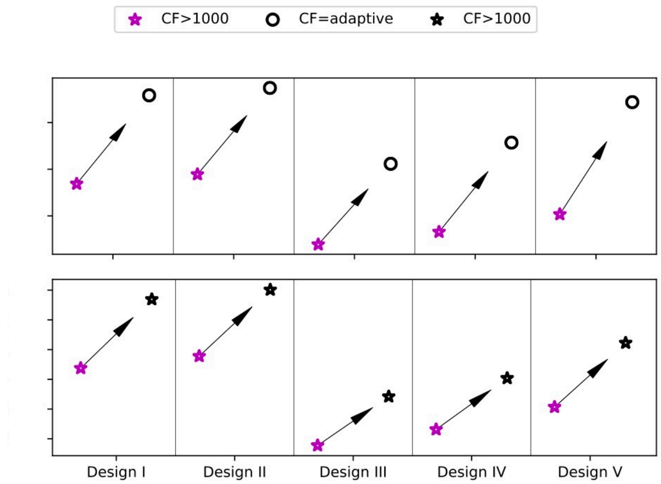


Fig. 3.21:
Average federate value in FES ($c_f = 600\ [relative : 0.6]$) when one federate switches to adaptive cost function: a) **adaptive federate**: federate value increases for the federate with adaptive strategy, b)**non-adaptive federate(s)**: federate value increases for other federates. The values are averaged values during 10000 time steps across 100 seeds for every circle.

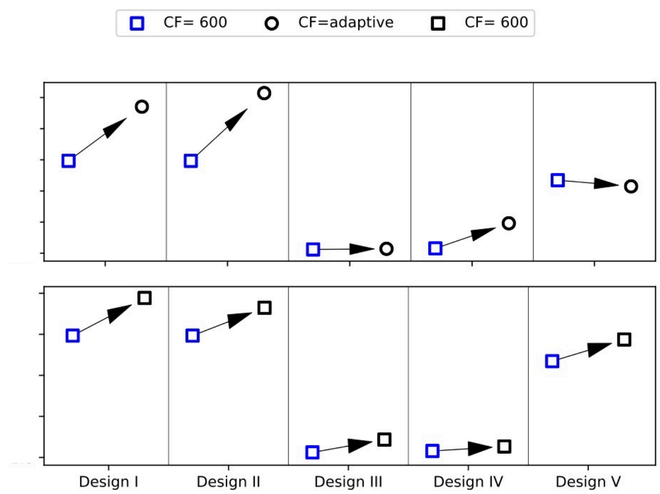
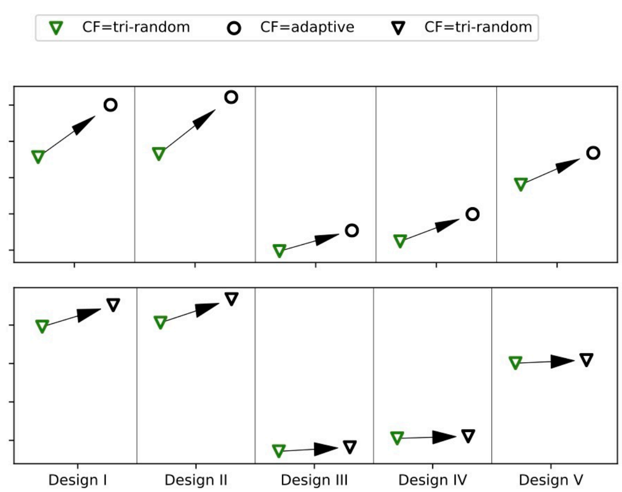
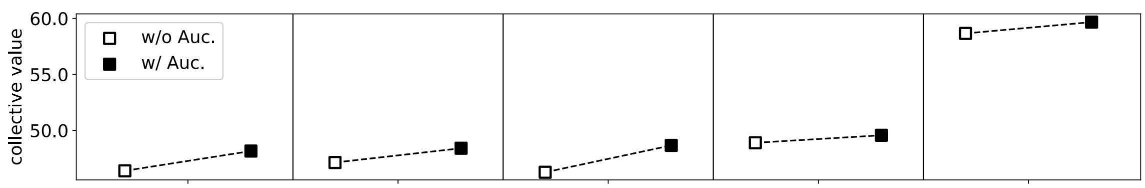


Fig. 3.22:
Average federate value in random FES ($E[c_f] = 0.6$) when one federate switches to adaptive cost function: a) **adaptive federate**: federate value increases for the federate with adaptive strategy, b)**non-adaptive federate(s)**: federate value increases for other federates. The values are averaged values during 10000 time steps across 100 seeds for every circle.

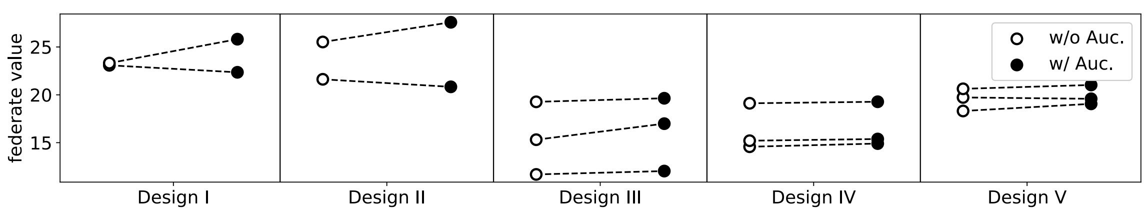


3.7 Simulation Study

This section presents and compares numerical results of the operational model and pricing mechanism in an FSS application. The application simulates the mechanisms for 10000 time steps and 300 seeds[1]. For simplicity, this model assumes equal data size and value functions for tasks, multiple destination nodes and different sources when storage size of an element and the capacity of a communication link are twice the size of a task data. Numerical results of the auctioneer are compared using five metrics: *collective value* for a federation, *individual value* for a federate, number of *shared links* among federates, averaged *relative bids* by federates, and averaged *prices* by the auctioneer.



(a)



(b)

Fig. 3.23: Effect of the pricing mechanism on cumulative collective values in five selected designs. The pricing mechanism improves: (a) *collective value* as the total value achieved by all federates and (b) *federate value:* as the individual value achieved by a federate. In a selected design, left points show the values by *adaptive* federates without pricing auctioneer (*w/o Auc.*) when right points are the values after implementing the pricing auctioneer (*w/ Auc.*). Each circle is averaged value of 300 simulation runs and y-axis is normalized by a 10^3 of maximum task values.

First, I discuss the effect of adaptive federated bids and auctioneer's prices on the collective value in a federation. In Fig. 3.18, an inclusive circle is a federation, a black square is an adaptive federate, red triangles are federates using marginal cost pricing, purple stars are independent

[1]The availability of tasks in the contextual model and random actions adopted by a learning federate plus the q-learning model depend on a random seed.

federates with no sharing of resources, and blue hexagons are federates using an arbitrary fixed cost function. Accordingly, a circle with all red triangles represent full sharing of resources following a centralized strategy or value of CES, a circle with purple stars represents the value of IES, and a circle with all black squares shows all adaptive federates in FES. Adding an adaptive federate always improves the collective value for IES and reduces collective value for CES. For FES, the effect of an adaptive federate depends on the state of exchanging resources across a federation.

The effect of the pricing mechanism on collective and individual values are shown in Fig. 3.23. Fig. 3.23a shows the collective effect of the auctioneer and the Fig. 3.23b disaggregates above values for each federate. The pricing mechanism has positive effect on collective values and most individual cases.

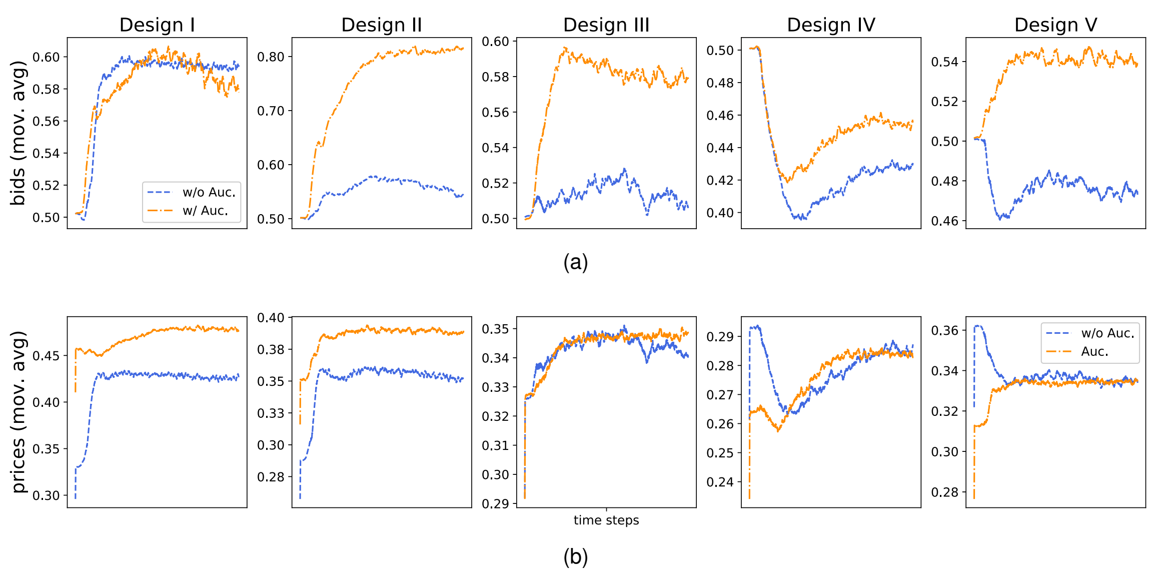


Fig. 3.24: bids and prices for exchanging communication resources with and without pricing auctioneer (*w/ Auc.* vs *w/o Auc.*): (a) *federated bids:* on upper row are submitted by *adaptive federates*, (b) *b) transaction prices:* on lower row show the actualized prices on sharing links among federates. The bids and prices are normalized by maximum task value.

Adaptive bidding depends on the expected value of state-actions and a few parameters in q-learning that contribute to the differences among values in various topologies, computational demands and states. In general, a higher bid can potentially deliver higher value but results in a lower chance of resource exchange. The plots in Fig. 3.24 distinguish some patterns of bidding and actualized transactions on exchanging resources. A network topology (i.e. design) affects the average bids by federates. In designs with scarce resources such as `D.I` (Design I) and `D.II`, bids and prices are higher compared to designs with more inter-coupled elements, path alternatives and higher competition among federates. For instance, adding a geosynchronous satellite to `D.III` (`D.IV`) reduces the level of bids and prices by more than 20%. In addition, average prices are between 18–50% percent lower than average bids. In a federation with adaptive bidding, expected bids and prices are lower than those with a pricing mechanism. The proposed mechanism increases actualized prices in designs with scarcer resources (`D.I` and `D.II`) more than other designs. Finally, introducing more federates while holding total number of elements constant increases competition and reduces the level of bidding in `D.III` compared to `D.II`. In these cases, prices remained within the same range while federates reduced bids.

In the proposed mechanism, the auctioneer ensures an expected value for a federate by exchanging resources among federates (see Sec. 3.5). In a federation of task processing elements, sharing resources results in more processed tasks and higher value for federates. The averaged number of shared resources across a federation emphasizes the effect of pricing auctioneer on collective sharing behavior across the federation. In Fig. 3.25, the upper row shows the average number of shared resources per time step. The pricing auctioneer increases sharing links by 4–16% for all designs even with higher prices per exchange of resources. The lower row of box plots shows the distribution of collective values for 300 simulation runs. The pricing auctioneer increases the collective value, reduces the standard deviation in collective gain and closes the gap between FES and CES in terms of economic efficiency.

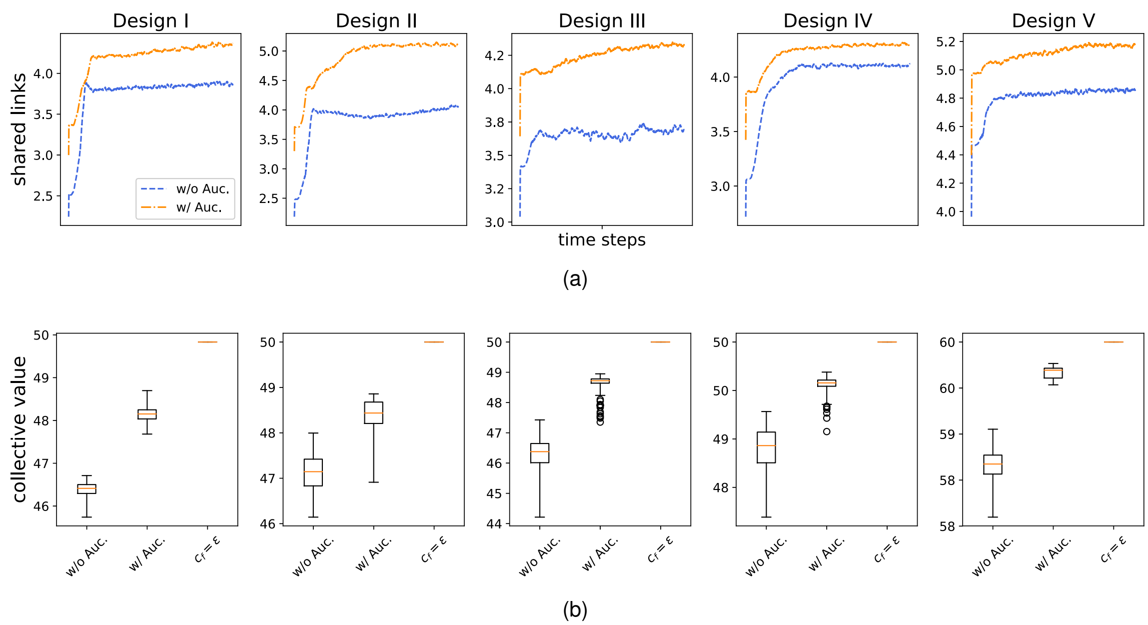


Fig. 3.25: Average number of shared resources and distribution of collective value with and without pricing auctioneer (*w/ Auc.* vs *w/o Auc.*): (a) *shared links:* on upper row show the averaged number of shared links among federates per time step, (b) *value distributions:* on lower row show the distribution of collective value in comparison to maximum possible value (centralized solution). The samples shown in box-plots include results from 300 simulation runs (seeds) where the upper whisker will extend to last datum less than `Q3+3*(Q3-Q1)`, lower whisker also extend to last datum greater than `Q1-3*(Q3-Q1)`. The collective value is normalized by 10^3 of maximum task values.

3.8 Discussion and Conclusion

This chapter developed a mechanism to suggest prices for sharing resources in a federated TNE. The network included processing, communication and storage resources, cost functions and strategic behavior by federates, and temporal demands for computational tasks. This chapter contributed: structural and behavioral models that define and formulate the objective function in a federated network, an operational model that optimizes the combinatorial problem of task scheduling and routing in a network, and a pricing mechanism that increases exchanging resources and collective value in a federation.

For research questions detailed in Sec. 3.2.1, this chapter introduced a MILP model for scheduling, storing, and retrieving tasks and routing processed data in an multi-source and multi-hop TNE with a centralized value function for the federation and dynamic topologies for satellites and communication links. The model assumes a trusted and third-party auctioneer with knowledge of available resources across the federation at each time step. The proposed model is value-maximizing in terms of a defined collective value as the objective function to find variables for processing (binary), storing (binary) and transmitting (integer) data. The operational limitations are defined by technical and financial constraints on resource capacity, payments, and costs for exchanging resources among participants. The operational solution is cost-minimizing for each federate in terms of delivering data to destination(s), accordingly, the solution is economic efficient, Nash equilibrium and Pareto-optimal as it solves a combinatorial problem using linear program. In this model, no federate has incentive to defect the routing solution by the auctioneer without utility cost. However, the computational cost of solving MILP model is exponential in time depending on number of elements in TNE. For optimization, a non-linear sequential least-square programm (SLSQP) is used to sequentially and iteratively find the maximum price for each federate while the objective function enforces financial constraints of the operational (MILP) model. Assuming rational strategic bidders, an reinforcement learning approach (q-learning) was used for discrete and adaptive bidding. For simulation model, multiple FSS designs were selected out of hundreds of possible designs from a Pareto-optimal line of cumulative value of a federation. The intuition is rational system designers will use the same logic to design their systems.

For validating the mechanism, multiple collective metrics were selected including average number of resource exchange, expected value for federates, and average prices for shared resources. Simulating the mechanism for the selected designs during thousands of time steps showed higher expected value for federates and higher prices for sharing resources. Higher prices in combination with higher resource exchange implies more efficient solution with higher incentive for sharing resources across the federation. Future researches in federated networks may investigate auctions, e.g. two-sided auctions with bidding behavior by participants (source, relay, and destinations), effects of adversary behavior by untruthful and strategic auctioneer, complexity of operational solution in time for scaling purposes, and efficient solutions for simultaneous pricing and resource allocation in a federated network. The next chapter investigates two-sided auction-based algorithms in a more general federated TNE.

Chapter 4

Auction-based Algorithms for Resource Allocation in Federated Networks

This chapter investigates auction and allocation mechanisms to drive behavior of decentralized components towards collective-efficient metrics. Multiple mechanisms are formulated for networked systems with distributed resources and entities with decentralized authority and control over resources. In particular, I investigate auction-based algorithms for exchanging resources and combinatorial routing in a federated network. A centralized and trusted auctioneer is introduced for routing and resource allocation and five auction-based algorithms are formulated: 1) linear program with binary search for prices, 2) first-price reverse-bid double auction, 3) non-linear searching for prices, 4) online algorithm with closed-form solution for prices, and 5) virtual pricing in a multi-source routing. The operational model and auction-based algorithms are implemented in a federated network with double-sided bids for link prices and path cost. The results are evaluated using extensive simulation runs in hundreds of network topologies with different configuration of elements and federates. The introduced metrics for numerical validation include normalized bids and prices, collective values, and convergence rates.

4.1 Introduction

In a "system-of-systems" or "collaborative systems", a cyber-physical standard for communication language among components exist. The operational and managerial independence of systems creates an extendible and nonexclusive core that incentivizes decentralized systems towards collective metrics. Collaboration seeks balanced access to information by every component that needs it, providing information only to those with proper authorization. In dynamic collaborative models such as a federated system with computational resources and tasks, *task-based access* grants access to resource by federates that are involved in a computational mission and *team-based access* is also applicable to dynamic coalitions and missions oriented around multiple tasks during multiple time steps. A collaborative scheme with resource allocation and an access control mechanism shall also hold against adversarial behavior by members and non-members. *Incomplete information game* frameworks are used to model the tradeoff between trust, privacy, and security against threats in networked systems [13, 142, 143]. Task processing networks of elements (TNEs) such as clouds, satellites, robotic teams or blockchain are networked structures with access control mechanisms for collaboration among multiple systems and participants.

In a network with decentralized entities and components, a global optimum and value-maximizing approach without an operational and financial agreement among participants is not a viable solution to resource allocation. Nonetheless, decentralized value-maximizing approaches executed by non-collaborative components result in sub-optimal solutions. In other words, independent operations by federates (e.g. a platform for cloud system or a constellation of satellites) is not economically efficient while a centralized solution is not feasible assuming distributed control, authority, and (potentially) design of resources. The combinatorics of resource exchanges among tasks and resource owners call for efficient and effective mechanism design, possibly, for financial agreement and collaborative resource allocation. These resources include on-board computational and communication resources in a TN. A targeted operational solution with resource exchanges among components fits between the value-maximizing global solution and the independent solution given for a set of computational elements. In this chapter, the operational model involves allocating resources for processing tasks and routing data to destinations in a network.

4.1.1 Auctions

An agreement for allocating resources and inter-federate financial exchanges is a subject of an *auction-based* mechanism in a network with decentralized components or a federation. Auctions are used to allocate bandwidth in networks, schedule tasks in distributed systems, share cloud resources among providers, execute missions by robotic teams, and in general, achieve an auctioneer-level goal (e.g. the global welfare) by discovering private preferences of heterogeneous components and willing participants and using them in resource allocation. In a coalition or federation, a participant with financial incentive to exchange its resources with others is a potential bidder in an auction-style operational model. An auction *mechanism* includes a form of submission by participants (i.e. bidding language), outcome evaluation, and winner selection. The desirability of an auction depends on general metrics of incentive compatibility, individual rationality and Pareto optimality. The *collective value* or *social welfare* reflects a global utility function that numerically assesses the auctioneer's outcome. In an auction, the individual utility function for a participant is usually defined as the difference between its valuation (e.g. a quasilinear function of a global valuation function) and clearing price for winner (buyer or seller). In sum, each resource is ought to be shared by a willing seller with least known valuation for that resource and allocated to the willing buyer with most valuation for it.

Accordingly, a designer of an auction mechanism shall consider decentralized objective functions, economic efficiency, adversarial security, bidding language and computational complexity. In a network with distributed components, computational challenges include solving combinatorial routing problems based on bidding preferences by resource owners and users and pricing resources based on those constraints and alternative solutions.

4.1.2 Federations

In *Star Trek*, Starfleet's policy to not interfere in norms of a culture abolished the apprehension and distrust between cultures and left them only to join and grow the world of federation. The post-scarcity federation doesn't rely on money but operates on individual freedom and abundance of resources (on demand in excess of basic needs) where conflicts are dealt with through distributed resource allocation, interactive consensus, and higher-level political mechanisms. In terms of laws in the federal system of Starfleet, a *governance* layer above any nation or component had authority on high-level mechanisms such as energy allocation, foreign relation, and accounting while individual planets and colonies retained their sovereignty. Laws were generally made "as close to home as possible" at individual planets and colonies by various species [144–147].

4.1.3 Research Problem and Objectives

In mechanism design for a federated network, multiple challenges exist regarding the objective, efficiency, language, and computational complexity of resource allocation. First, a centralized auctioneer simplifies resource allocation, pricing, and payment across a network but increases the communication overhead among federates and the auctioneer. Second, a submission language must reduce the auction costs in communication overhead, convergence time, and computational challenge of bidding by federates. Further, an operational model may allocate resources given literal constrain announced by bidders or maximize values with constraint relaxation and compensate affected bidders using an incentive-compatible mechanism, or use virtual pricing to avoid literal constraints in combinatorial resource allocation.

This chapter investigates and implements multiple auction-based algorithms for combinatorial resource allocation (i.e. routing) in a federated TNE. The research objectives pursue performance of auction-based algorithms in a federated network with adaptive bidding by selfish and non-cooperative participants. An application with an operational model for routing, an auction-based environment, and adaptive bidding by federates is developed then performance of the auction-based algorithms, driven from or modified based on existing models in literature, are compared using the application. This chapter is focused on performance of algorithms with respect to economic efficiency, behavioral stability, computational complexity, and auction time. I assume a networked structure of elements with pre-defined technical capacity, task value and also assumes selfish (and non-colluding) behavior by federates to assess the performance of proposed auctions. The scalability of the proposed algorithms are evaluated assuming a growing number of elements and federates. I assume a networked structure of elements with pre-defined technical capacity, task value, a trusted third-party auctioneer, and selfish (and non-colluding) federates. In addition, I introduce a trusted third-party auctioneer for pricing communication resources in a federated network to simplify and optimize routing, auction language and information exchange among participants and the auctioneer and explore in depth the effectiveness of a mechanism for coordination and efficiency in a networked system.

Sec. 4.2 reviews relevant literature in auction mechanism design in networks and combinatorial problems. Sec. 4.3 discusses assumptions, notations and an operational model of routing with technical and financial constraints. Sec. 4.4 formulates and illustrates the five auction-based algorithms for pricing and allocating resources. Sec. 4.5 introduces static and dynamic metrics in simulation study and shows analytical results for static model and simulation results of collective value, bids, and prices for dynamic model. Sec 4.6 discusses static and dynamic results and statistically analyzes those algorithms in terms of computational and performance metrics including: collective value for economic efficiency, an auction's runtime for computational cost, and convergence rates for the auction's speed.

4.2 Related Literature

Scheduling tasks and allocating computational resources in networks by an centralized auctioneer is usually a combinatorial problem that can be modeled using Linear Programming (LP) models [148, 149]. In combinatorial auctions, bidding by participants and winner selection by the auctioneer are dealt with as NP-complete or NP-hard problems. Auction mechanisms applicable to combinatorial items include sealed-bid auction (e.g. first-price or reverse-price), Vickery-Clarke-Groves (VCG) mechanisms, *market-clearing* price, online auctions, and *iterative* auctions. The iterative and online schemes usually simplify communication language between the auctioneer and bidders by gradually disclosing information and adapting resource allocation to dynamic environments. For instance, in iterative ascending-bid auctions, buyers submit bids sequentially to the auctioneer and the auction terminates once no buyer bids a higher price. Reversely, in iterative descending-bid auctions, a seller reduces its price until a buyer accepts the most recent price. In online auction-based algorithms, a seller offers multiple units of an resource while bidders appear sequentially which leads to lack of information on valuation by buyers and lack of information on the next bids by the seller [150]. These sequential steps reveals private information of participants towards a collective metric and reduces the complexity of submission and winner selection. In combinatorial auctions (CA), other challenges include submission by participants, winner determination and cooperation among participants [72]. For bidding, the submission language must reduce the information overhead and auction time. Winner determination must be *transparent* to simplify a bidder's understanding of and trust in the mechanism. In terms of computational cost, submission, pricing, and winner selection shall be scalable in most real-world applications.

In literature, two types of auctions are proposed for data-routing in wireless networks: 1) a source bids and an auctioneer allocates the path and 2) data is a bidder on time slots for access to a relaying hop [71]. For routing data, Zhang et al. proposed a reverse-auction mechanism for cooperative relaying of data when at each hop, a seller (source) node selects the next buyer (destination) [151]. Another reverse-auction model, called ABIDE, is employed for routing in peer-to-peer mobile networks when a relay-broker node is financially incentivized by other peer nodes [152]. In [153], a game-theoretic VCG mechanism is developed for efficient and truthful routing in mobile ad-hoc networks. Grosu et al investigated application of pricing mechanisms to achieve an *social optimum* in a TNE when elements reveal their processing power and a mechanism assigns payments to resource owners [154]. In distributed systems, a negotiation mechanism for suppliers and consumers is proposed for cloud systems that use a updating algorithm for prices, namely *trade-off*, that changes prices while keeps expected utilities by agents [155]. A pricing mechanisms uses mixed-integer nonlinear program (MINLP) to model optimizing task accuracy via assignment of heterogeneous resources. An inter-agent coordination protocol is proposed to maximize overall team goal of mission accuracy in task processing and routing. A contract-based multi-agent mechanism is also investigated to allocate path resources and achieve agreement among networked elements [156].

The VCG auction is a general form of second-price *Vickrey* auction to incentivize truthful behavior by bidders which operates by separating a winner's clearing price from its bid [157–159]. An iterative VCG-based mechanism with random update of prices and bids is applied for allocating resources among multiple players [160]. For pricing the shortest path in networks, assume e as the weight (cost or time) associated with a link l in network G. The *Vickery*'s payment for links on shortest path is defined as:

$$p : (x, y, l) \rightarrow \begin{cases} c(x, y; G|_{e=\infty}) - c(x, y; G|_{e=0}) & l \in \{\text{links on shortest path}\} \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

where $c(x, y, G|_{e=0})$ is the cost of data transmission from x to y using shortest path when $e = 0$ and $c(x, y; G|_{e=\infty})$ as the cost when link associated with e doesn't exist. The price is the opportunity cost of removing links on the shortest path.

In a network with known routes and an arbitrary topology, the Progressive Second Price (PSP) auction can achieve incentive compatibility in bidding for the links along a route while all users will be truthful regardless of the bids by other users [83]. An iterative auction can simplify communication language between an auctioneer and bidders by gradually disclosing information and adapting to dynamic environments [72]. Using an iterative scheme and dual optimization, an iterative mechanism updates prices and allocates resources to maximize utility function for all participants in a network [61]. In [149], two approaches are applied to a virtual machine allocation problem (VMAP). In one method, *linear program* (LP) relaxation and randomized winner selection are employed similar to the mechanism introduced in [148]. In a second approach, an approximate solution to an auction is found assuming a determined (single-minded in terms of bundle) bidder [161]. In a Bayesian approach, supply constraints are satisfied only in expectation (*ex-ante*) and the objective function is linearly separable over buyers (e.g. welfare or total revenue). In a classical Bayesian auction, expected revenue is optimized by allocating resources to self-interested participants with a known distribution of preferences [162]. In the case for multiple buyers, a general approach with no assumption on value function, type distribution, and constraints are proposed to reduce the mechanism design to single-buyer subproblem [163]. By this approach, the decisions are optimal for a bidder and also coordinated among buyers because of supply constraints.

Online auctions are introduced for dynamic resource allocation when an auctioneer collects bids at any time and can allocate resources immediately [71]. Online auctions are explored for truthful, strategy-proof, individually rational and budget-balanced sharing of spectrum channels between primary and secondary users in wireless networks [164]. An heuristic online algorithm with monotonic pricing is proposed for channel allocation and end-to-end routing in multi-hop network [165]. In terms of a behavioral model for federates in a network, *bidding* strategy and possible *adversarial* behavior affects functionality and efficiency of an auction mechanism. Bidders submit bids concurrently or iteratively. In the former, conflicting bids fail to obtain resources while in sequential bidding, a bidder adjusts its bid being informed of other bids or resource allocation at each iteration [166]. Reinforcement learning is used for modeling behavior by participants during an iterative or online auction [44, 167–169].

A *federated network* is a federation with networked elements, computational resource owners and resource users with the possibility of *interdependency* among federates in terms of exchanging resources and information [22]. With operational and managerial independence of systems, a federated system relies on an extendible and nonexclusive core that incentivizes federates towards an adaptive and collective goal while holding its own structure against adversarial and selfish behavior by federates and exogenous systems. In the 1990s, a generic *federation layer* was proposed to unify application design, reuse and automation of system design in Web-based application systems [170]. Today, the definition of federated systems extends to cloud systems, low earth orbit (LEO) satellites, swarms of drones and unmanned aviation, robotic emergency teams, etc. [15–19]. In federated networks such as a connected set of satellites, or a virtual machine network in cloud systems, auctions are designed for path finding, sharing computational resources, data transmission, etc. These networks usually face dynamic topologies, limited on-board resources and stringent constraints and delay in communication and data transmission [171]. In a federated auction, winners are determined by an operational model or *routing* algorithm which allocates tasks to elements and resources to tasks. In this regard, the most relevant auction designs are implemented in multi-hop wireless networks. In these networks, a message is distributed from a source through multiple relay nodes to a destination where source nodes are buyers with payments to relay nodes as reward for their cooperation.

In a federation, mechanism design is also a viable solution for resource allocation among distributed entities. For cloud platforms, Integer linear program (ILP) and optimization mechanisms perform allocation and pricing for *virtual machines* (VM) [39, 45]. In [39], Rebai investigates mechanisms for allocating and pricing distributed resources in a federated VM network and proposes an *exact federation* algorithm to maximize revenue and minimize cost for cloud providers. The ILP model addresses the NP-hard problem of resource allocation where an auctioneer guarantees a competitive social welfare [45, 172]. In [173], a cloud broker (CB) acts as a beneficial mediator for both cloud service providers (CSP) and end-users compared to current situation with public cloud providers (PP) such as Google and Amazon. In a similar framework, auction mechanism can achieve higher profit for cloud providers, higher utility for end users and optimal allocation of resources across federation [174]. In a FSS model, Pica and Golkar evaluate a sealed-bid reverse auction pricing scheme for exchanging underused resources among owners and third-parties where a satellite either directly connects to a ground station or seeks a data service (e.g. relaying) to overcome limited onboard data storage [85].

4.2.1 Research Assumptions and Questions

The research in this chapter follows these assumptions:

- A1 a network of computational elements (sources) with limited communication resources connecting those elements to destination(s) in a multi-hop and multi-source network
- A2 a federated architecture of elements with distributed resources for each federate (auction participant), inter-federate communication capability, and decentralized objective functions for participants
- A3 a trusted third-party auctioneer/LP-solver with knowledge of available resources at each time step
- A4 a double-sided auction mechanisms run by the auctioneer with adaptive seal-bids by participants at each time step

A5 in this chapter, the proposed auction-based algorithms may not rely on exhaustive solutions to routing in a federated network because MILP model is assumed to be computationally expensive

Multiple mechanism have been developed for scheduling and routing problem in wireless ad-hoc networks, spectrum secondary allocation, and cloud systems. In an ad-hoc peer-to-peer mobile network, a network topology of mobile peers cooperate to achieve higher value through exchanging resources [152]. The auction-based approach increases number of service providers in a network and resource exchange among peers. While the mechanism considers a set of query issuing, brokers, data provider, and relaying MPs, the proposed mechanism doesn't consider multi-functional MPs with all capabilities by Asm. A1. In addition, the model assumes that each node acts independently and not under an alternative cooperative scheme, which is in contrast to Asm. A2. In [166] and [165], online and dynamic auctions are explored for a secondary network (SN). In the former work, multiple wireless service providers (WSPs) compete for available spectrum band from a pool of spectrum. Nonetheless, this work doesn't hold the Asm. A1 in network connection of elements and communication resources. The latter work uses a topology for secondary network with multi-hop and end-to-end routing. Their method is similar to the research problem in the sense that an auction participant is a network of distributed elements by Asm. A1. Nonetheless, this work finds pricing for communication channels shared among SNs to find the optimum combination of end-to-end connection. This doesn't consider using communication links and computational elements using inter-network resources assumed by Asm. A2. In addition, the solution proposed by the authors doesn't hold for double-sided bids by Asm. A4. The VCG mechanism for pricing links in routing application, formulated in [160], is computationally expensive for our application as for every existing link a routing solution shall be computed. Instead, the chapter is looking for mechanisms that require minimum number of routing and re-routing in networks. In [39,148,149], the solutions to pricing VMs in clouds also doesn't apply to problems in this chapter as they doesn't hold against the Asm A1 for limited communication resources in a multi-hop network.

The research question addressed in this chapter is: *How to formulate auction-based algorithms to incentivize inter-federate exchange of resources and drive decentralized components toward better collective metrics such as higher value and lower computational cost?*. The research question can be disaggregated into:

Q1 How to define a two-sided auction in using an operational MILP model for combinatorial routing and scheduling problem in TNE?

Q2 How to find the most efficient prices that satisfy bidding constraints and maximizes expected values for federates (e.g. binary search)?

Q3 How to formulate well-defined auction formats (e.g. first-price and second-price sealed-bid auctions) for the combinatorial problem of routing and scheduling?

Q4 How to reduce computational cost of an auction for finding efficient prices?

Q5 What are the collective metrics for evaluating auction algorithms in a federated network?

Q6 How auction-based algorithms with lower complexity in time affect the long-term performance of the auctioneer?

Q7 How virtual pricing with balanced payment among sellers and buyers but non-equal prices for them affect the ultimate performance of the auctioneer?

4.2.2 Research Methodology and Design

In this chapter, the following steps address the research question:

S1 defining structural and functional components of a federated network of task-processing elements (TNE)

S2 formulating the operational model of a centralized RAS based on the value and cost functions of federates and resources

S3 formulating three auction-based algorithms from literature as the baseline cases

S4 developing two auction-based algorithms for pricing communication resources in a multi-hop and multi-source TNE

S5 developing simulation study, metrics, and test cases to validate the effectiveness of the proposed algorithms

In this chapter, a general model of federated networks is formulated. In step S1, a TNE is introduced with nodes as resources, multiple resource owners called federates, communication links, and technical assumptions on data and computational capacity. In step S2, scheduling and routing problem is formulated using MILP model where its solution to financial constraints is calculated. Step S3 formulates three auction-based algorithms including first-price reverse-price auction, binary search for price with iterative solutions to linear program, and non-linear algorithm for pricing resources. Step S4 develops two new algorithms with closed-form solution and minimum computational cost for the auctioneer. For a simulation study, an application combines above modules and multiple combination of TNE configurations are used to put into test the hypothesis in this chapter. For testing the introduced method, 240 designs are selected that covers two or three federates with different network sizes and topologies. Finally, in step S5, multiple collective metrics are introduced including bids submitted by federates, actualized prices for exchanging resources, convergence rates for algorithms, and an additive function for the global utility in a federation.

4.3 Operational Model

In the auction-based resource allocation, exchanging resources among federates is achieved through an *auctioneer* equipped with an operational model. A network topology represents computational tasks and destinations with nodes, communication resources with edges and federates with colors. Assume N tasks and M elements. The operational model is run at each time step when new tasks are available for processing. $\mathbf{T} = \{T_i\}$ shows set of N tasks. The value of a computational task is defined using a *value function*:

$$\mathcal{V}(T) : T \in \mathbf{T} \rightarrow V_T \in \mathbb{R}$$

A task is also associated with a data size which is retrieved by data function:

$$\mathcal{D} : T \in \mathbf{T} \rightarrow datasize \in \mathbb{Z}^+$$

and $\mathbf{E} = \{e_j\}$ represents set of elements, $\mathbf{F} = \{f_k\}$ is set of federates and $\mathbf{L}_{ij} = \{l_n\}$ is set of available links between elements e_i and e_j. A function $\mathcal{F}$ retrieves the federate owning an element, sharing a link, and processing a task:

$$\mathcal{F}_e : e \in \mathbf{E} \rightarrow f \in \mathbf{F}$$

$$\mathcal{F}_l : l \in \mathbf{L} \rightarrow f \in \mathbf{F}$$

$$\mathcal{F}_t : T \in \mathbf{T} \rightarrow f \in \mathbf{F}$$

and the owner of a link is defined as the destination element of data on the link or receiver element: $\mathcal{E}(l_{ij}) : l \in \mathbf{L} \rightarrow e_j$. At each time step and with availability of new tasks, the auctioneer runs auction A. A federate f submits bids for sharing its links with other federates and path cost for delivering its computational tasks, represented by functions:

$$\mathcal{B}_l : f \in \mathbf{F} \rightarrow (linkbid \in \mathbb{R}^+ < V_T)$$

$$\mathcal{B}_p : f \in \mathbf{F} \rightarrow (pathbid \in \mathbb{R}^+ < V_T)$$

respectively. The auctioneer, by running an auction, responds to federates with prices for links:

$$\mathcal{P} : (T, l) \rightarrow linkprice \in \mathbb{R}$$

A mixed-integer linear program (MILP) for operational model of orbital network or satellites was introduced by authors in Sec. 3.4. This section formulates a MILP model for solving multi-task processing and multi-hop routing in a federated network. The operational solution results in binary variables of *processing* computational tasks by elements (x_{proc}), integer variables of *transmitting* data by communication links between elements (x_{trans}), binary variables of *resolving* tasks by destination elements ($x_{resolve}$):

$$x_{proc} : (T \in \mathbf{T}, e \in \mathbf{E}) \rightarrow \{0, 1\}$$

$$x_{trans} : (T \in \mathbf{T}, l \in \mathbf{L}) \rightarrow \mathbb{Z}^{+}$$

$$x_{resolve} : (T \in \mathbf{T}, e \in \mathbf{E}) \rightarrow \{0, 1\}$$

We introduce a cost function for data transmission accordingly:

$$\mathcal{C}(T, l) = \begin{cases} \epsilon_l & \mathcal{F}_l(l) = \mathcal{F}_t(T) \\ P(T, l) & otherwise \end{cases}$$

where $\epsilon_l << V_T$ is cost-of-energy (see [153]) of using a communication link l and is significantly smaller than the processing value of a task.

An auctioneer actualizes financial exchange among federates by pricing resources under a mechanism. A value function finds the value of auctioneer for a federate f:

$$\begin{aligned}\mathcal{V}(f) = &\sum_{T:\mathcal{F}_t(T)=f} \mathcal{V}(T) \\ &+ \sum_{T:\mathcal{F}_t(T)=f,} \sum_{l:\mathcal{F}_l(l)\neq f} x_{trans}(T,l) \times \mathcal{C}(T,l) \\ &- \sum_{T:\mathcal{F}_t(T)\neq f} \sum_{l:\mathcal{F}_l(l)=f} x_{trans}(T,l) \times \mathcal{C}(T,l) \end{aligned} \tag{4.2}$$

which sums value of processing tasks and that of sharing resources minus payment to other federates.

For simplicity, I use *value* for *collective value* in a federation. In the operational model, the objective function maximizes value of a federation:

$$\begin{aligned}&\text{find:} x_{proc}(T,e), x_{trans}(l), x_{resolve}(T,e) \\ &\text{maximize:} \sum_{f\in\mathbf{F}} \mathcal{V}(f)\end{aligned} \tag{4.3}$$

subject to:

$$B_p(\mathcal{F}_t(T)) \geq \sum_{l\in L} x_{trans}(T,l) \times \mathcal{C}_l(T,l) \tag{4.4}$$

$$D_{in}(e) = D_{out}(e), \forall e \in E \tag{4.5}$$

where:

$$D_{in}(e) = \sum_{T} x_{proc}(T,e) + \sum_{l_{ij}:e_j=e} x_{trans}(l_{ij})$$

is sum of data inflow to or processed data on an element. In a similar fashion:

$$D_{out}(e) = \sum_{T} x_{resolve}(T, e) + \sum_{l_{ij}:e_i=e} x_{trans}(T, l_{ij})$$

is the sum of data outflow from or resolved on an element. Constraint 4.4 ensures that the cost of using an end-to-end path is smaller than the path bid by a path user, i.e. path source and task processor. R maps a cost function to the operational solution:

$$\mathcal{R} : \mathcal{C} \xrightarrow{Eq.\ 4.3 \& \mathcal{B}_p(f)} R \equiv (x_{proc}, x_{trans}, x_{resolve})$$

Using Eq. 4.2, $V_R(f)$ and V_R are federated values and the value of operational solution R. The solution to the above model ranges from no resource exchange, to fully shared resources across federation, depending on technical and financial constraints. For instance, in a federated network with no inter-federate link, no exchange of resources is possible and the above model can be solved by federates independently. On the other hand, assume that inter-federate links exist and federates share their resources with minimum value, then, above model is the value-maximizing solution. Finally, any other configuration of bids by federates results in a result between these two extremes in terms of value. These categories are called *independent*, *centralized*, and *federated* solutions and are explained in details in following sections.

4.3.1 Independent Solution (IS)

No sharing of resources among federates happens given the technical and financial constraints. The solution to the operational model and a value-maximizing solution by each federate give the same results (i.e. zero value-added by the mechanism in Eq. 4.3). For instance, submitted bids with these conditions result in an *independent* solution regardless of technical constraints: $\max_f (B_p) < \min_f (B_l)$. The current situation with design of satellite systems (e.g. constellations and swarms of satellites) resembles independent operation.

4.3.2 Centralized Solution (CS)

Under a centralized solution, federates share resources with others for the minimum (energy) cost. The operational solution results in value-maximizing sharing of resources in federation, namely *centralized* solution. This condition results in the most collective benefit for a federation but not necessarily for all federates. Under a centralized scheme, the solution is anonymous in the sense that re-configuring the federates won't change the solution, actually, the solution resembles a solution to elements belonging to one federate. For instance, these two conditions ensure centralized solution to operational model: $B_p(\mathcal{F}_t(T)) = V_T$ and $B_l(\mathcal{F}_l(l)) = \epsilon_l$.

4.3.3 Federated Solution (FS)

Under realistic assumptions and using an operational model with feasible technical constraints for inter-federate resource exchange, the solution to an operational model reflects strategic decisions (bids) by federates. This solution represents the most realistic case for a federated network as federates consider internal priorities (e.g. opportunity cost of resources) and translated to strategic bids or premium for relaying data. This strategic bid is similar to the concept of strategic premium over actual cost of energy by relaying nodes in a communication network [153]. This solution assumes distributed control and authority on resources by a federate when it informs federation of its available resources at each time step and join a limited agreement by auctioneer's mechanism resulting in the federation solution. Any reasonable condition outside IS and CS might lead to a federated solution where:

$$V_{IS} < V_{FS} < V_{CS}$$

4.4 Proposed Algorithms

The operational model introduced in Sec. 4.3 solves the efficient routing for a given set of feasible prices for links. Nonetheless, offering resource prices in a feasible boundary constrained by bids is not trivial as higher prices might alter the routing solution. In theory, offering prices and allocating resources shall be performed simultaneously. The VCG scheme for pricing shortest path suggests the highest price that won't change resource allocation, i.e. the price that is independent of bidder's declaration [153]. In this section, auction algorithms includes finding prices and searching for most efficient operational solution. I formulate five models for pricing resources for link owners. For linguistic simplicity of auctions in communication between federates and the auctioneer, I assume equal bids for all links owned by one federate.

4.4.1 LP Binary Auction (LPA)

In [149], the CA-LP model was introduced to search for the resource values for cloud users in a combinatorial auction. Algorithm 4-Iintroduces a reverse auction for a routing solution and binary-search for maximum prices applicable to the solution. This mechanism is based on solving the MILP model in Sec. 4.3 and searching for a combinatorics of prices higher than link bids that also satisfy path bids. In this algorithm, distinct prices are limited by the number of federates, i.e. one price for a federate. This assumption reduces communication overhead of running the auction [175] and the computational cost of searching for prices.

Algorithm 4-I: LPA

1: *Phase 1: Collection*

2: *Collect* computational tasks $\mathbf{T}$ on potential task-processing elements

3: *Collect* seller and buyers bid functions from federates: $\mathcal{B}_l(f)$ and $\mathcal{B}_p(f)$ from federates.

Phase 2: Routing & Price Search

find *MILP* initial solution/value $R = \mathcal{R}(B_l)$ and V_R.

find *upper* and *lower* boundary for link prices: $(pl_f, pu_f) : f \in F$.

while $pl_f \neq pu_f$ **do**:

    find mid-price $pm_f = (pl_f + pu_f)/2$

    solve MILP solution $M = \mathcal{R}(pl_f)$ value for mid-price: V_M

    if $V_M = V_R$ **then**:

        $pl_f = pm_f$

    else

        $pu_f = pm_f$

    end if

end while

suggest $\mathcal{P}(f) = pl_f$ to federates

4.4.2 First-price Sealed Auction (FPA)

Pica and Golkar evaluated performance and cost of sealed-bid reverse auctions including a first-price algorithm for reallocating resources in an application case of federated satellite systems [85]. FPA implements the first-price reverse auction for pricing resources for a value-maximizing routing solution:

$$R^* = \mathcal{R}(\mathcal{B}_l, \mathcal{B}_p)$$

$$\mathcal{P}(T, l) = \mathcal{B}_l(l)$$

where $\mathcal{B}_l$ is the proposed bids for links, R^* is the routing solution, and $\mathcal{P}(T, l)$ is the proposed prices by the auctioneer to federates.

4.4.3 Sequential Least-square Pricing (SLA)

In Section 3.5.1, I introduced an application of a fast non-linear algorithm in developing an incentive-compatible pricing mechanism in a federated network. This algorithm uses a sequential iterative method for constrained nonlinear optimization, namely sequential least squares programming (SLSQP), to maximize prices and ensure an expected value for a federate in an auction run. Accordingly, SLA maximizes inter-federate prices and use FS (Sec. 4.3.3) values as constraints for pricing:

$$\max_{\mathcal{P}} \left[\sum_{l \in \mathbf{L}, T \in \mathbf{T}} x_{trans}^{R^*}(l) \times P(T,l) \right] \tag{4.6}$$

subject to:

$$\mathcal{V}_{R^*} \geq \mathcal{V}_R \tag{4.7}$$

$$\mathcal{V}_{R^*}(f) < \mathcal{V}_R(f) \qquad \forall f \in \mathbf{F} \tag{4.8}$$

$$\mathcal{P}(T,l) \geq \mathcal{B}_l(l) - k * \epsilon \quad \forall l \in \mathbf{L} \tag{4.9}$$

where $R*$ is MILP routing solution to relaxation in bids:

$$R^* = \mathcal{R}(\mathcal{B}_l - k * \epsilon, \mathcal{B}_p)$$

and $R = \mathcal{R}(\mathcal{B}_l)$ is routing solution to literal bidding constrains. The reason for relaxing the constraints is to explore alternative solutions with minimal rerouting in operational solution. In Chapter 3, maximum flexibility was considered for pricing constraints, i.e. $\mathcal{P}(T,l) \geq \epsilon$ but in SLA, limited rerouting flexibility ($k = 3$) is assumed for closing the gap between prices and bids. However, this method is incentive-compatible by at each auction run (not necessarily in multiple runs) by Eq. 4.8 since all federates benefit from being truthful to accept the auctioneer's prices.

Algorithm `4-II` for SLA is:

Algorithm 4-II: SLA

1: **federated:** find solution with bidding constraints: $R = \mathcal{R}(\mathcal{B}_l)$.

2: **values:** find federated values: $V_R(f)$.

3: **path bundles:** find all solutions $\mathbf{S}$ with $\mathcal{P}(T,l) \geq \mathcal{B}_l(l) - k * \epsilon$ and $V_{R^*}(f) \geq V_R(f)$

4: **sort bundles:** find solution S with maximum total value V_{R^*}: S_m

5: **prices:** find and suggest prices $\mathcal{P}(S_m)$ solving SLA objective function in Eq.4.6

4.4.4 Online Closed-Form Pricing (ONA)

In multi-hop device-to-device communication networks, *online* auctions are applied to scheduling spectrum by primary users in a secondary market, routing data, and matching sellers and buyers [176, 177]. In this section, I introduce a closed form solution for pricing communication links in a multi-hop source-to-destination routing. Algorithm `4-III` gives a pricing solution for resources shared by federates and used by multiple computational tasks in a routing solution. In this algorithm, the idea is to find maximum link prices on each path that doesn't violate constraints and prioritizes increasing lowest bids on a path.

Algorithm 4-III: ONA

1: *Phase 1: Collection*

2: *Collect* computational tasks $\{T_i\}$ on potential task-processing elements

3: *Collect* seller and buyers bid functions from federates: $\mathcal{B}_l(f)$ and $\mathcal{B}_p(f)$ from federates $f \in \mathbf{F}$.

4: *Phase 2: Routing*

5: find *MILP* solution/value $R = \mathcal{R}(B_l)$ and V_R.

Phase 3: Pricing

for $T \in \mathbf{T}$ **do**

  $\mathcal{P}^*(T,l) = B_l(\mathcal{F}(l)) : \forall l \in L$

  while *True* **do**

    find minimum link price for T:

$$pr_{min} = \min_{l} \mathcal{P}^*(T,l)$$

$$l_{min} = arg \min_{l} \mathcal{P}^*(T,l) \tag{4.10}$$

    find the second lowest price for T:

$$pr_2 = \min_{l:2^{nd}\mathcal{P}^*(T,l)} \mathcal{P}^*(T,l)$$

    if

$$\sum_{l \in L,T} [x_{trans} \times \max_{l} (\mathcal{P}^*(T,l), pr_2)] > B_p(f_T)$$

(i.e. cost with pr_2 violates a path bid)

then:

      end raising link prices (break *while* loop)

    else

      update minimum price with second pr_2:

$$\mathcal{P}^*(T, l_{min}) = pr_2 : \forall l \in L_P$$

    end if

  end while

end for

find minimum price/link(s) in $\mathcal{P}^*$ using Eq. 4.10 $\rightarrow (pr_{min}, \mathbf{l}_{min})$

raise minimum price until:

$$\sum_{l \in L,T} [x_{trans}(T,l) \times \mathcal{P}^*(T,l)] = B_p(f_T)$$

suggest minimum calculated price for each federate:

$$\mathcal{P}(f) = \min_{T,l:\mathcal{F}(l)=f} P^*(T,l) \tag{4.11}$$

Proposition 1. Federation (collective) value by *ONA* is equal to the value by *LPA*.

Proof. Assume R and V_R are routing solutions and value by the *operational model* in Eq. 4.3 and assume that the $\mathcal{P}_{LPA}$ and $\mathcal{P}_{ONA}$ are suggested prices by those algorithms. From auctioneer functionality (see Algorithms I and II), I know that $\mathcal{P}_{LPA}(T,l) \geq \mathcal{B}_l(l)$ and $\mathcal{P}_{ONA}(T,l) \geq \mathcal{B}_l(l) : \forall l \in L$. Then, any solution R^* with $V_{R*} > V_R$ must be R as it satisfies bidding constraints and has higher value: $R^* = R$. □

According to this proposition, I apply *ONA* instead of *LPA* as it is faster in runtime (see Sec. 4.6.3).

4.4.5 Virtual Multi-Path Pricing (VPA)

delivering multiple tasks, multiple prices for sharing resources might be applicable for different tasks as long as payments remain balanced for federates. In this section, I retain calculated online prices $\mathcal{P}^*(T,l)$ (in Eq. 4.11) and propose balanced virtual prices for links in a routing solution:

$$\mathcal{P}(f) = \frac{\sum_{T,l:\mathcal{F}_l(l)=f} \mathcal{P}^*(T,l)}{\sum_{T,l:\mathcal{F}_l(l)=f} x_{trans}(T,l)} \tag{4.12}$$

Lemma 1. Virtual pricing in Eq. 4.12 results in a balanced payment.

Proof. Assume a pricing scheme H in which a federate pays $\mathcal{P}^*(T,l)$ for link l, the payment is balanced because the seller also receives the same amount for sharing each link. Now, for a federate f, if total payment by VPA is equal to H's, the former payment is proved to be balanced:

$$payment(f) = \sum_{T,l:\mathcal{F}_l(l)=f} x_{trans}(T,l) \times \mathcal{P}(f)$$
$$= \sum_{T,l:\mathcal{F}_l(l)=f} \mathcal{P}^*(T,l) = H(f)$$

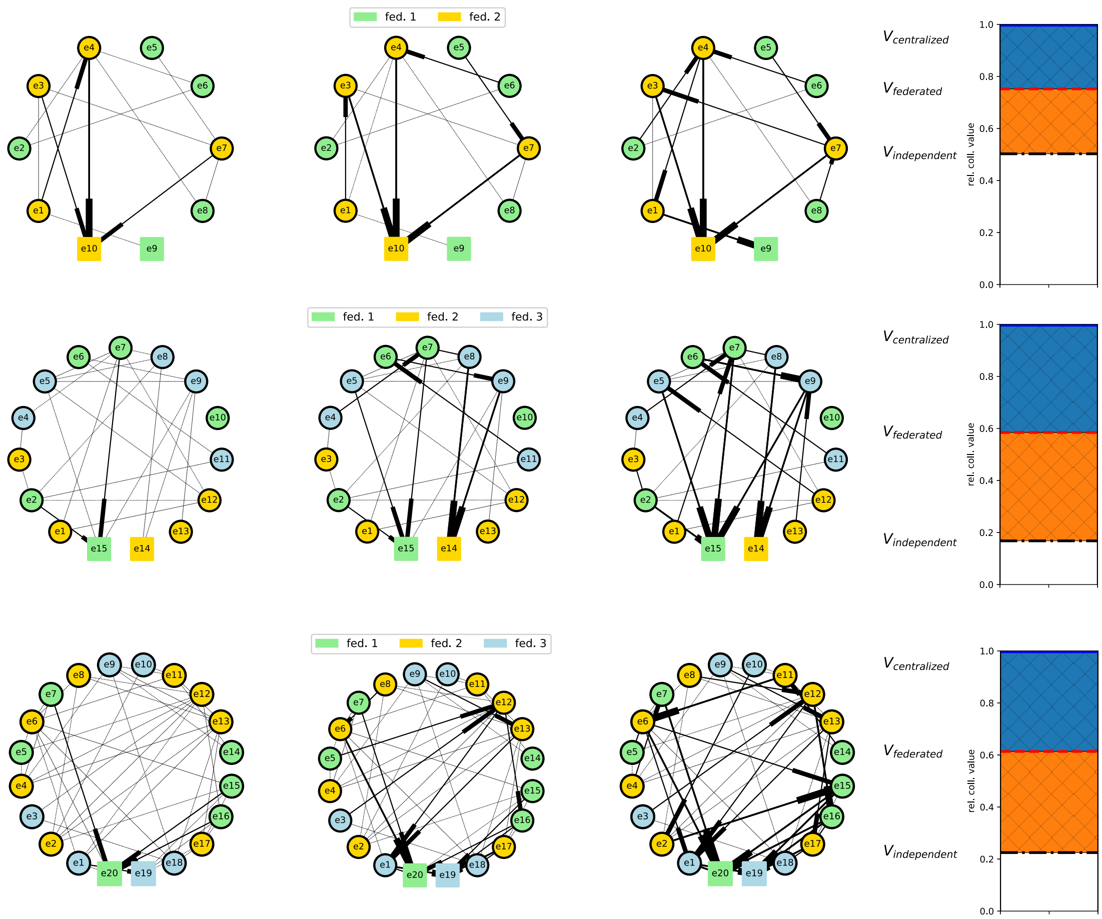
□

4.5 Simulation Study

This section proposes a federated application intending to evaluate the auctioneer's economic efficiency, behavioral effects, pricing functionality, convergence time and computational cost for the proposed algorithms in different networks. The application in this section receives bids from federates, executes an operational model, and offers prices for inter-federate resource exchanges based on the algorithms introduced in Sec. 4.4. For initial conditions, I consider a network topology consisting of elements, federates and communication links among those elements. In the following sections, static and dynamic models are introduced with assumptions and metrics for model validation. For simplicity, bids and prices are relative to the maximum task value.

4.5.1 Network Topologies

In a federated TN, a topology includes source and destination elements and communication links among those elements. A feasible solution by an operational model to a topology depends on its number of elements, federates and existing links. In this chapter, the number of edges are defined relative to its maximum number in a complete network, namely *edge density*. In addition, the number of federates affects *independent* and *federated* solutions (IS and FS). Intuitively, for a given set of elements and a fixed set of bids, more federates reduce the value of IS and FS as it creates more constraints in the operational model and higher combinatorics of prices in the pricing mechanisms. In this section, I explore the federated networks with 10, 15, and 20 elements, 2 or 3 federates and edge densities in $\{0.33, 0.22, 0.15, 0.1\}$. Two elements are assumed to be equivalent destinations of delivering data while other elements are potential sources for processing tasks. For each topology, I assume tasks are available to all elements except to the destinations and link capacities for transmitting task data is twice a task's data size, i.e. a link is usable by two separate sources. Finally, I assume data size and delivery value are equal for all tasks.



(a) independent (b) federated (c) centralized (d) collective value

Fig. 4.1: Federated TNE with 8, 13, and 18 processors and 2 destinations with optimum results for: *a) Independent (IS) b) Federated (FS)* with path-cost bid more than for one-link resource exchange among federates (i.e. $price_p > price_l >> \epsilon$), *c) Centralized (CS)* and *d) Values:* relative values of IS and FS to CS.

In building networks, a random seed shapes a topology in terms of its inter-element links and distribution of elements among federates. Since the number of available tasks only depends on the number of elements, value of CS (centralized solution) is limited by the number of paths for delivering data to destinations. Real-world networks such as satellite systems have fewer number of destination elements or hubs (e.g. ground stations in a satellite swarm) but these elements are more connected (central in a network) than others. Accordingly, I consider higher probability for connecting those nodes.

Figures 4.1 shows selected topologies with bids that produce routing solutions for CS, FS and IS. The selected network topologies with 8, 13, and 18 sources, two destinations, and three federates for each topology. For FS, link bids are equal for all federates and is more than one-link resource exchange among federates ($\mathcal{B}_p > \mathcal{B}_l \gg \epsilon$). The CS has high path bid by task processors ($\mathcal{B}_p \approx V_T$) and minimum link bids by resource sharing federates ($\mathcal{B}_l \approx \epsilon$). The right column shows is relative values of independent and federated solutions to centralized solution.

4.5.2 Static Model

Static results explore the effect of non-competitive double-bids by bidders on the auctioneer's output in terms of values and prices for federates within one time step. In static model, I assume equal bids by all federates when a federate f's link and path bids change within a range: $\epsilon \leq bid_l(f) < 1$ and $\epsilon < bid_p(f) \leq 1$. The total number of auction runs is 2.4 million cases including all permutations of double bids for four designs and all topologies. According to Sec. 4.3, the minimum bid_l and maximum bid_p show the CS and any $bid_p < bid_l$ leads to IS.

For this model, Fig. 4.2 shows averaged values versus financial constraints (link and path bids) in the operational model. The values are averaged values relative to CS across 240 topologies introduced in previous section. This model considers granularity of 100 distinct link bids and 50 path bids being assumed as equal for federates. Intuitively and analytically, both lowering link bids and raising path bids result in greater-or-equal values for a federation. With similar assumptions, the

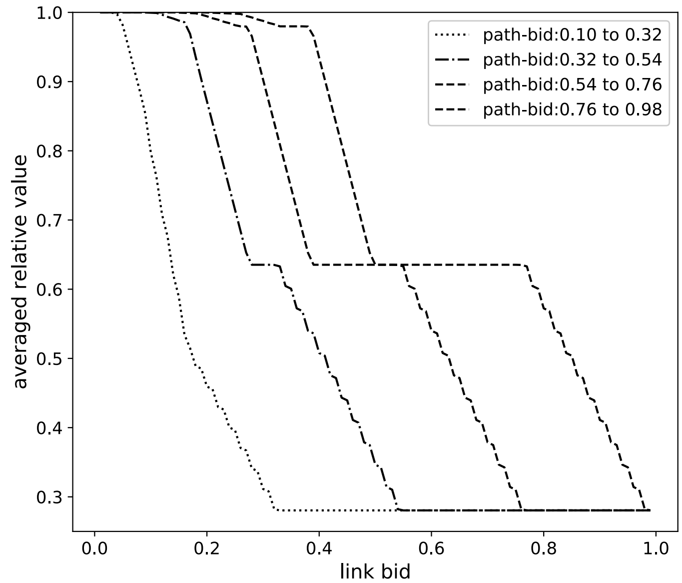


Fig. 4.2: The averaged *collective value* of federation versus link bids by link owners and path bids by path users: lower link bids and higher path bids (higher path cost cap) result in higher values for a federation. The values are averaged relative values across 240 topologies in Sec. 4.5.1. The bids are assumed to be equal for federates with granularity of 100 link bids and 50 path bids (1.2 million simulation runs). Collective values versus bids for one auction-run are equal for all mechanisms as bids are assumed with no sequential bids.

plots in Fig. 4.3 show the auctioneer's functionality in terms of pricing resources for four algorithms corresponding to link bids and ranges of path bids. In the latter figure, for no resource exchange among federates, i.e. a failed auction run, the actualized price is considered to be zero.

4.5.3 Dynamic Model

In the context of online auctions or those with sequential bidding, an auctioneer's performance depends on static parameters and dynamic ones such as bidding behavior by selfish participants. In this section, I introduce building blocks of a dynamic application and run the proposed algorithms for 6000 time steps and 100 initial seeds of random functions in q-learning. All dynamic visualizations use the moving average of values with window-size of 40 steps resulting in 150 points for a temporal plot and dynamic metrics for bids, prices, values, convergence, and computational cost are introduced.

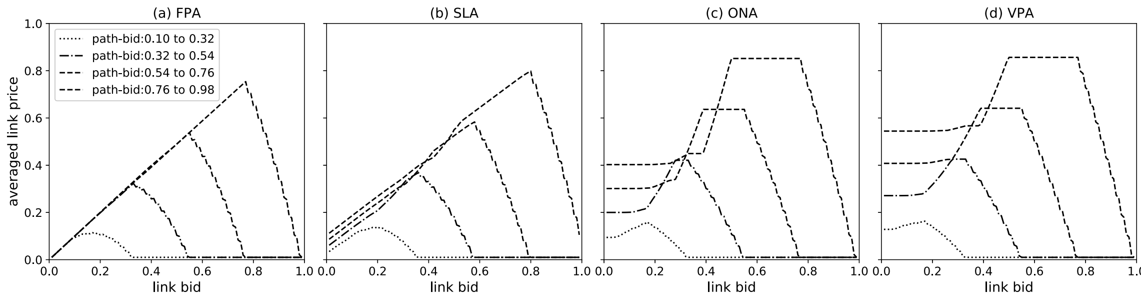


Fig. 4.3: The prices by auctioneer for *link* and *path* bids by federates: *(a) FPA:* first-price auction in which a resource owner receives its suggested bid-value from task processor after resource allocation, *(b) SLA:* sequential non-linear that maximizes the prices for resource owners using SLSQP algorithm, *(c) ONA:* online auction which is the result of closed-form calculated prices by Algorithm `4-III`, and *(d) VPA* online auction with virtual prices that calculates closed-form calculated link prices from Algorithm `4-III` and Eq. 4.12.

In [44], the authors introduce a q-learning technique with Gaussian update for Q-values Q^t_{jf} of federate f with index j at time t:

$$Q^t_{jf} \leftarrow Q(b^t_{jf}) + \alpha_{ij}[V^t_{if} + \gamma Q(b^{t+1}_{jf}) - Q(b^t_{jf})] \tag{4.13}$$

where b^t_{jf} is bid by federate f with index j at time t, α_{ij} is learning factor for closeness between actions with indices i and j, V^t_{if} is value at time t for federate f for bid with index i, i.e. Q-value of j is updated using value from bid i. In above formulation, α is a Gaussian distance function with lower α_{ij} for farther bids when $\alpha_{ii} = \max_j \alpha_{ij}$.

The above model starts from higher effect of observation on learned behavior converges to more specific bids later in time. In addition, for a topology I don't consider states and a Q-value corresponds to a bid index. Two random parameters (r_g and r_l) help with finding global-optimum bid at earlier time steps and local adjustments in later time steps:

$$\begin{aligned} r^{t+1}_g &= \beta_g r^t_g \\ r^{t+1}_l &= \beta_l r^t_l \end{aligned} \tag{4.14}$$

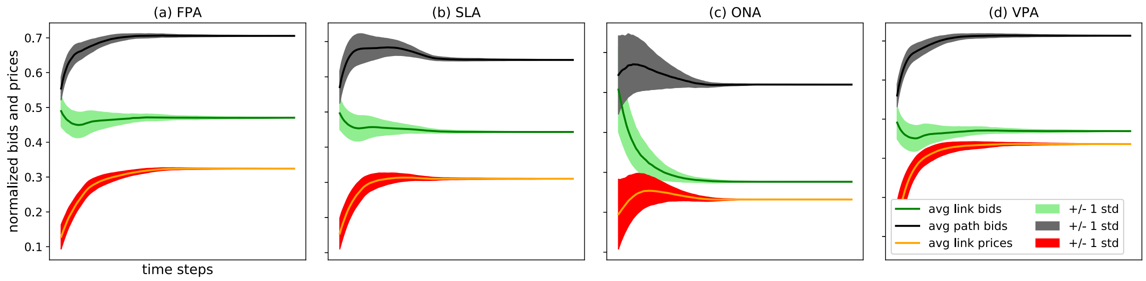


Fig. 4.4: Averaged link bids, path bids, and link prices by federates normalized by the *task value* and expected bids/prices for each topology. The colored areas are within one standard deviation. The convergence implies bids approaching toward the expected values for each federate and algorithm and bid type (link or path): *a) FPA*, *b) SLA*, *c) ONA* and *d) VPA*. ONA's link bids converge to lowest bids.

where r_g selects random bids and r_l determines stepwise movement of bids. The random parameters for q-learning in Eq. 4.14 are selected as: $r_g^0 = 1$ and $r_l^0 = 0.05$ while these parameters change at rates of $\beta_g = 0.997$ and $\beta_l = 0.998$ at each time step[1].

- ***Normalized Bids:*** A metric is introduced to normalize bids by steady-state bids in each topology and averaged bids in each algorithm. Using this technique, I can both show convergence in bids and compare relative bids across algorithms on y axis. Assume b_n^{mt} is relative bid at time step n for topology t and algorithm m. Also assume that cb^{mt} is the converged (steady-state) bid. The normalized bid for visualization is calculated by:

$$nb_n^{mt} = \frac{b_n^{mt}}{cb^{mt}} \times \sum_t cb^{mt} \tag{4.15}$$

  This function is separately but similarly applied to link and path bids and results are shown in Fig. 4.4. The colored areas show the bids within one standard deviation of mean bids within a topology.

- ***Auctioneer's Prices:*** The prices proposed by the auctioneer reflects the actualized resource exchanges between federates. The function introduced in Eq. 4.15 is also applied to normalize prices and Fig. 4.4 also shows normalized prices for actualized exchanges, i.e. won

[1] Performance of algorithms in our simulation study are consistent for different combinations of β_g and β_l

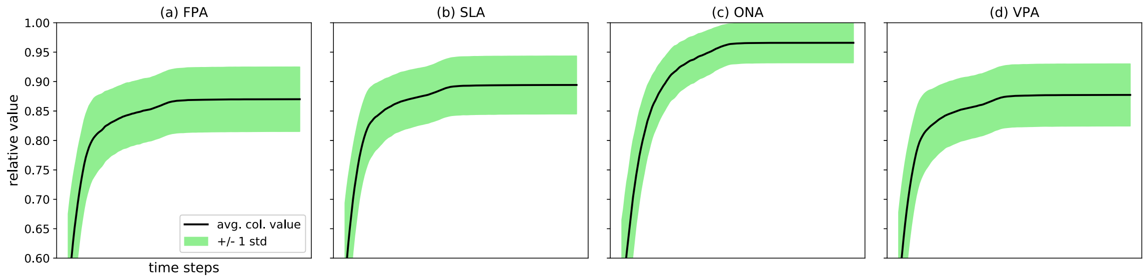


Fig. 4.5: Averaged value of federation per auction-run normalized by value of CS and the area within one standard deviation: *(a) FPA:* first rice reverse bid auction, *b) SLA:* non-linear rerouting and pricing of auction, *c) ONA:* online algorithm with closed form prices and *d) VPA:* virtual pricing for multi-task routing solution. The simulation is run for 6000 time steps each topology among 240 variations with 100 initial random seeds for *q-learning* algorithm for bidding by federates. All values represent moving average of values with window size of 80 and 40 for convolution and sampling respectively. In terms of results, ONA's prices show converging to highest value and most consistent results across 240 topologies.

auctions, assuming price equal to zero for auction runs without any resource exchange.

- ***Collective Value:*** A collective value shows the sum of values collected by federates in a time step. For each topology, a collective value is normalized by the maximum value of federation and is averaged across different seeds. While a maximum price or bid is not objectively definable for on a topology, the maximum value driven from CS. Fig. 4.5 shows the evolution of *collective value* for the algorithms across different topologies. The simulation is run for 6000 time steps each topology among 240 variations with 100 initial random seeds for *q-learning* algorithm for bidding by federates. The shown values are moving average of values with window size of 60 and 30 for convolution and sampling respectively. The colored areas show relative values within one standard deviation of the mean values across topologies.

4.6 Analysis and Discussion

In this section, I first analyze static and dynamic modes based on simulation results then introduce statistical and computational perspective to auction-based algorithms and discuss implications of this work compared to similar studies in literature.

4.6.1 Static Analysis

For the static model, results show decreasing values for increasing bids except for some flat intervals particularly for more relaxed path constraints (Fig. 4.2). The decreasing values are because of additional routing and path constraints in the operational model. The flat intervals indicate a range of bids where only one inter-federate link is feasible in a path. For instance, assuming equal path bids $bid_p = 0.9$, any link bid $0.45 \lessapprox bid_l \lessapprox 0.9$ results in the same routing solution with maximum one inter-federate link per path. The ripples in the second part of the latter plot is due to the granularity of path bids and the number of topologies. Although a value interval doesn't change a routing solution, it may hide negative correlation between resulted values and submitted bids to federates, i.e. non-decreasing reward for increasing bids.

For pricing results by the static model (see Fig. 4.3), FPA offers equal prices to bids as long as those bids satisfy path constraints. For lower ranges of path constraints, averaged prices falls earlier as the number of failed auction runs increases ($price = 0$). The suggested prices by the SLA is slightly higher than FPA's as this algorithm searches for an alternative routing solution with a relaxation in bids. The ONA offers higher prices relative to SLA while highest prices belong to VPA. In the two latter results, I observe two main steps in pricing: *1) The lower step* belonging to CS with maximum value when a range of link bids won't affect the solution and *2) The upper step* belonging to routing solution with one inter-federate links per path: $price_l \approx bid_p$. The static results show that in two algorithms FPA and SLA, suggested prices are directly proportionate to bids while in ONA and VPA, prices are stepwise and indirectly proportionate to bids. For instance, in the latter algorithms, increasing bid doesn't necessarily result in higher value for a federate, i.e. the auctions are almost *monotone*.

4.6.2 Dynamic Analysis

The results by the dynamic model show: 1) effect of selfish behavior by federates on bids, prices, and values in time, 2) effect of topologies on values, and 3) effect of algorithm on convergence in and across topologies. In the ONA, bids on links converge to the lowest ones (≈ 0.42) among algorithms and are strictly decreasing while path bids also converge to lowest bids (see Fig. 4.4). The link prices in the same figures, calculated using actualized inter-federate exchanges, are rather increasing which show higher prices by the auctioneer as a result of more practical bids by federates. Nonetheless, these prices don't reflect the number of exchanges between federates which might be higher for more efficient auctions. Then, the lower prices in ONA is because of more resource exchanges among federates. For collective values, ONA achieves highest values with minimum deviation among topologies. In Fig. 4.5, the results for FPA and VPA are similar in terms averaged value and variance while they have most difference in prices among algorithms. SLA achieves slightly higher average values than these two algorithms and lower values than ONA's.

In the dynamic model, I introduce a metric for convergence rate of values given a topology or an algorithm. Order and rate of convergence are widely used in numerical methods and defined by p and μ where an iteration is $x_{n+1} = g(x_n)$ and:

$$\frac{|x_{n+1} - x^*|}{|x_n - x^*|^p} \rightarrow \mu$$

with x^* as target value. In above equation, the convergence is linear for $p = 1$ and rate of $0 < \mu < 1$. In above equations, I distinguish *individual* and *aggregated* rates of convergence. In the former equation, target value x^* is distinct for each topology while in aggregated case the target value is averaged value across all topologies. In both equations, I consider absolute values for nominator and denominator at each time step. In particular, assume x_n^t is the averaged value of an topology t at time step n and x^{t*} is final converged value for topology t:

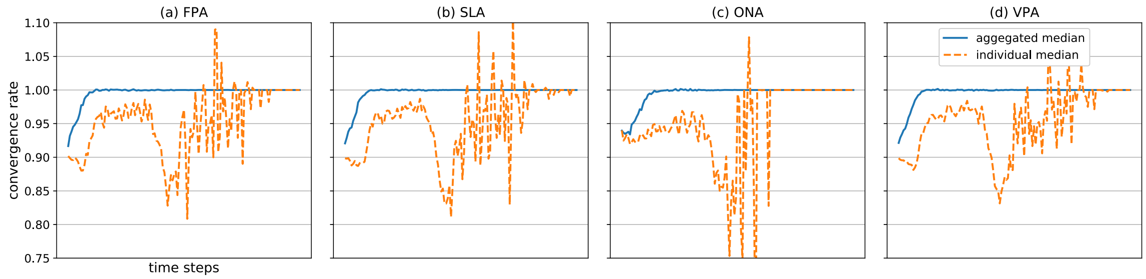


Fig. 4.6: Median *individual* and *aggregated* convergence rates for all topologies.

$$
\begin{aligned}
converate_{ind} &= \frac{|x_{n+1}^{t} - x^{t*}|}{|x_{n}^{t} - x^{t*}|} \\
converate_{agg} &= \frac{|x_{n+1}^{t} - \frac{1}{240}\sum_{t} x^{t*}|}{|x_{n}^{t} - \frac{1}{240}\sum_{t} x^{t*}|}
\end{aligned} \tag{4.16}
$$

where 240 is the number of topologies.

Fig. 4.6 shows the individual and aggregated convergence rates for collective values of all topologies. In this figure, FPA, SLA and VPA are faster converging than ONA, although toward sub-optimal and lower values. The individual and aggregated convergence rates in Fig. 4.6 show that convergence is faster (rate is lower) for SLA and VPA algorithms earlier in time. Almost all individual rates are less than one ($\mu < 1$) because the values are improving in a distinct topology. Initially, individual rates go downward for diminishing effect of random selection in bids during first steps. The rates rises when the stepwise random selection for local optimum starts and the interaction effect $\{\alpha_{ij} : i \neq j\}$ in Eq. 4.13 is still high. The median convergence rate reaches minimum value as both random adjustments and α_{ij} decrease which relatively stabilize values after half of time steps being passed. The quantile variation in individual convergence also decreases in later time steps. The individual values converge to final values for ONA faster than other algorithms as the individual rates are equal to one after about $4k$ of auction runs.

The aggregated rates start from low values, rise towards one with high variation and stays

Variables	DF	F	p-Value	Eta-sq
Topologies	239	11079	0.0	0.725555
Auctions	3	9576	0.0	0.007872
Topol.:Auct.	717	19	0.0	0.003786

Table 4.1: Two-Way ANOVA: Value vs Network Topologies and Auction Algorithms

close to one for later time steps. During initial steps, values are extremely sub-optimal as the result of random exchanges in federation, i.e. exploration in q-learning. However, convergence in aggregated rates stop while individual federates find better values because structural differences among topologies result in various caps on collective values. In sum, ONA offers convergence, not fastest but most consistent in terms of expected value, individual rates, and aggregated variation.

4.6.3 Statistical Analysis and Computational Cost

For statistical results, I shall separate the effect of an algorithm from that of a network topology. A two-way ANOVA test is applied to collective values using a linear model from *statsmodel* in Python[2]. In this model, independent variables are three sets of categorical variables for *topologies*, *algorithms* and mutual interactions among those, respectively with 239, 3 and 717 degrees of freedom.

Fig. 4.7 compares converged values of federation among algorithms from 10 latest steps across all topologies. The values are relative to value of CS for each topology. The results from a two-way ANOVA test using two categorical variables of topology and algorithm are presented in Table 4.1 where ONA achieves significantly higher values among algorithms.

For computational costs, I use random topologies different from the proposed topologies in Sec. 4.5.1 because we need to run each algorithm once (for one time step vs 6000 time steps in the dynamic model) to assess algorithmic cost and we can practically afford to assess the cost for larger networks. A fixed link $density$ equal to 0.1 in networks with 10 to 34 nodes is considered with

[2] http://www.statsmodels.org/dev/anova.html

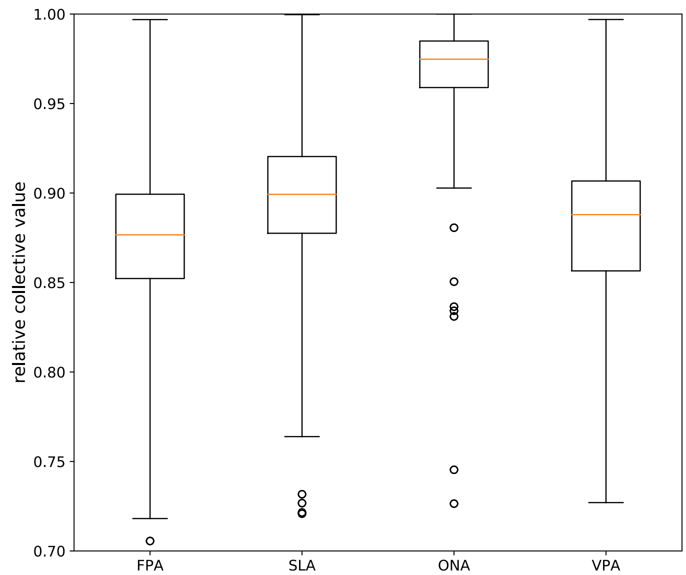


Fig. 4.7: Converged collective values of topologies for algorithms during 10 latest steps resulting in 2400 points per algorithm.

30 random seeds to build connections in topologies. Finally, I consider 2 to 10 federates for each simulation run and network topology resulting in total of 27000 simulation runs. Fig. 4.8a shows the runtime of SLA and MILP (VPA/ONA/VRA). The values are averaged across 30 seeds of distinct topologies and in networks with close to 0.1 *edge-density*. The left figure shows the logarithmic scale of averaged time based on number of nodes and the right figure shows the averaged runtime for different ranges of networks size.

FPA, ONA, and VPA are computationally equivalent to one operational run of MILP while SLA needs at least two operational runs for a federated solution in finding values and a relaxed solution for rerouting and finding prices. Although MILP is NP-hard, depending on the model and constraint, it can be solvable in polynomial time. The complexity of a sequential algorithm for finding price is independent from and significantly lower than the operational solution. Fig. 4.8a shows the relative runtime of algorithms that are linear in logarithmic plot, i.e. exponential in time. The algorithmic runtime also depends on the contextual model. Based on Fig. 4.8b, runtime rises with the number of federates at first (up to 4-5 federates) and goes down afterwards noticing the logarithmic scales

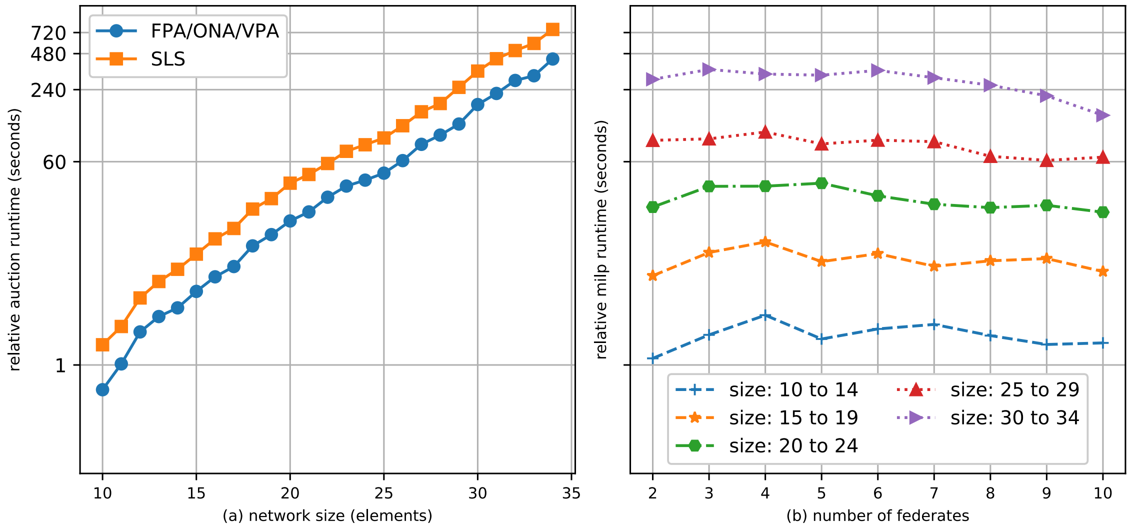


Fig. 4.8: Computational runtime of auction-based algorithms based on network size and number of federates.

in this figure. This observation is because of first increasing the complexity of satisfying constraints in operational run, second, decreasing the number of feasible exchanges in a federation. In addition to the operational cost, the computational cost of a LPA also depends on a binary search for prices. The binary search depends on the granularity of target price and the range of price search. For m possible prices for each federates, the computational cost of binary search is $log(m)$ powered by number of federates and multiplied by a runtime of MILP. Assuming a granularity of 50 ($2 * \epsilon$ precision) for prices and two federates, the LPA runs in $3.9^2 \times runtime(MILP)$. Fig. 4.9 compares the runtime of LPA to ONA's for smaller set of networks.

4.7 Discussion and Conclusion

For the research questions in Sec. 4.2.1, this chapter formulated five auction-based algorithms for the scheduling and routing problem in a network. In the context of a federated system with bidding constraints and efficient solution for resource allocation, an auction-based algorithm finds the prices for resources aimed at being economically efficient, individually rational, and truthful. In

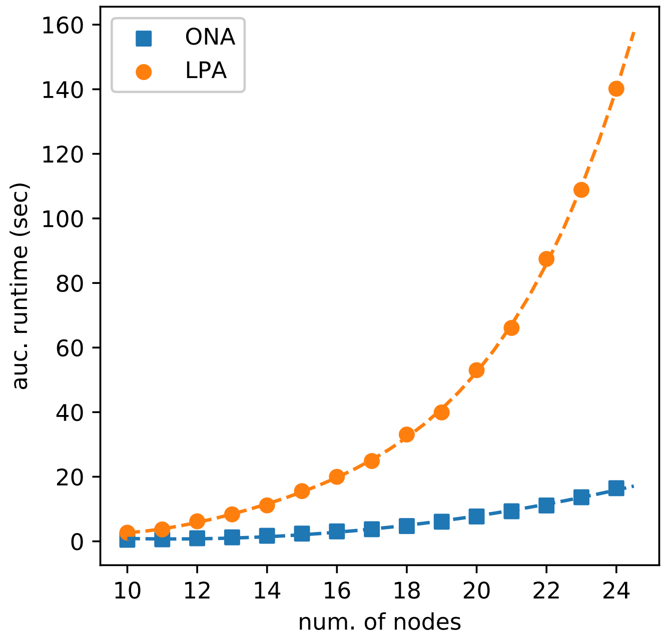


Fig. 4.9: Computational runtime of ONA and LPA based on network size.

TNE, a two-sided auction are modeled using an additional variable for each routing path (i.e. path-cost-bid) which captures the minimum value a federate expects for processing a task. For finding most efficient prices, an auctioneer should search for prices using a combination of prices with the mentioned characteristics. This results in designing an algorithm with iterative MILP solution and binary search for prices. In a routing problem, I formulated first-price sealed-bid reverse auction by assuming a link price equal to its bid. Nonetheless, defining a unified second-price auction or VCG scheme for combinatorial multi-source and multi-hop routing is challenging. Through an online algorithm and a virtual pricing, a second-price and VCG schemes is represented using increasing prices and reducing the variation of prices for resource owners on each routing path. Using the online algorithm (ONA), I suggested a closed-form solution for prices by maximizing prices on each path and reducing variation among prices and finding the minimum price among links shared by a federate across all its paths. In the case with virtual pricing (VPA), algorithms use the same technique except that it suggests average prices to each federate versus minimum price by ONA, which implies that prices for resources might be different for the seller and the buyer of a link while payments remain balanced. Finally, an auction model with non-linear sequential programming (SLA) maximizes prices for shared resource while maintains same prices for each federate.

The proposed metrics for model validation include collective value of processing tasks across the federation, actualized prices for exchanging resources in successful auctions, and convergence rate of the algorithm. For simulation study, a q-learning technique is used to model rational strategic bidders/federates. The resulting figures show that ONA algorithm with variation-reducing approach significantly increases the collective value across a federation while reduces the actualized prices and bids for sharing resources. In addition, convergence rate for the proposed ONA algorithm is faster while variation in value across different topologies is lower. The same algorithm with virtual pricing (VPA) fails to maintain the same performance level although suggested prices for sharing resources are higher.

This chapter contributed: 1) formulating a value-maximizing operational model that allocates online tasks to elements and communication resource to computational tasks given a federated network with structural topology and financial constraints, 2) formulating linear program auction with binary search for prices, 3) an online algorithm for closed-form solution to pricing links on a given routing solution, 4) a virtual pricing algorithms for routing multiple tasks, paths and federates. In this chapter, the communication cost relates to inter-federate data transmission and is formulated by constraints in an operational mechanism. The study plan involved a static model for demonstrating auctioneer's behavior in a structural topology and a dynamic model to evaluate the effect of mechanisms on collective values given selfish bidding behavior by federates. The results showed that by ONA mechanism, an auctioneer proposes equilibrium prices for sharing inter-federate resources (see Fig. 4.4) and offers higher value for exchanging resources by federates (see Fig. 4.5) and lower cost for inter-federate communications. The pricing behavior and values resulted from the developed mechanisms in this work showed monotonic growth in value and stability in bidding behavior by federates. The results compared four selected auction-based algorithms in terms of auction time, convergence, computational cost, and their adaptability to various network topologies.

Future works may develop mechanisms to address adversarial behavior by heterogenous participants such as those modeled in social networks by *sybil* attacks (i.e. fake identities and tasks) [87] and mechanisms for allocating resources to combinatorial team tasks, executed on multiple elements with shared assets among owners.

Chapter 5

Influence-based Information Exchange in Social Networks

This chapter investigates a dynamic mechanism for influence-based information exchange in interactive and social networks. The social influence is defined by being able to create content that is circulated on the network. The nature of influence is captured using network structure of content contribution, e.g. tweeting, retweeting, and replying on Twitter. For information exchange, a temporal framework for detecting and clustering emergent and viral topics on social networks is developed. For validating and visualizing clustering results for viral topics, three clustering metrics of popularity, burstiness, and relevance score are introduced when two temporal graphical models show timing, cluster size, and temporal granularity of viral topics. In this chapter, two camps of users are identified and recognized with media-driven and interaction-driven influence. The results are matched with real-world news circulation by following and searching for viral topics during the same period of data analysis (210 days). A simulation model is defined and formulated based on a real-world user network in twitter and the observed characteristics of users. Using the same clustering model for Twitter data, I detect viral simulated topics and associate the characteristics of those clusters to interactive nature of social networks.

5.1 Introduction

According to facebook on organized attempt for influencing 2016 election, $100000 was spent on 3000 ads by inauthentic accounts and pages [178]. The process of targeting and influencing a population on a social network under the current mechanism is extremely cheap and fast [179]. On the other hand, humans are more likely to be involved in the process of circulating fake news than bots due their emotional responses associated with fear, surprise, and disgusts facing those news [180]. Due to the low cost of news circulation and effectiveness of a networked structure, a dynamic mechanism involving behavioral characteristics of participants and spreading dynamics of viral topics may effectively share a perspective on circulation of content in interactive networks with social structure.

In a survey conducted by Pew Research Center in 2017, 32, 68 and 74 percent of youtube, facebook and twitter users have reported getting news on these social media platforms respectively. In the case of twitter, this number was up 15 percent from the previous year [181]. Twitter has evolved from "a toy for bored celebrities" as described by the New York Times columnist Maureen Dowd [182] towards a news source for 68 million users in the US since its creation in 2006. Twitter differs from the other two platforms in the way tweeters contribute to information exchange on the platform. It enables users to tweet, *retweet* (republishing a tweet for followers), *quote* (expressing an opinion along with another), and *reply* (introducing a destination for a tweet). In addition, since November 2017, a member can express herself in 280 characters that although is 100 percent higher than the previous limit but is lower than book-length characters a user can share on Facebook[1]. On an interactive social platform, aside from *hard* limitations applied to a content, *soft* limitations such as attention span of audience and type of audience limits the freedom of contributors in terms of using a conventional language with an arbitrary message and length (see [98, 183, 184]).

Ubiquitous use of social platforms, e.g. *Twitter* and *Sina Weibo*, by diverse contributors and their content limitations create a new type of communication called *microblogs* that allows exchange of content including links, images, and brief sentences across a network for a set of audience where a user can both consume and produce content [23, 24, 26]. Thereby, *topics* might range from daily life

[1] Most famously, 280 limit on Twitter,1300 for status update on Linkedin, and not as much applicable limit of 63,206 for a single post on Facebook

to events, news, stories, lifestyle, and personal interests [25]. Accordingly, contributors may invent and apply new linguistic tools aimed at more efficient communication with: minimal grammar, brevity in wordings, and frequent abbreviations. An example of these minimization was the introduction of *hashtags* to twitter in 2007 [27].

The latter linguistic techniques increase the number of audience who participate in a conversation, imitate using a social language, and contribute to communication in form and content. The use of social language with automated content creates enormous opportunity and threats for contributions by automated accounts (bots) on social platforms. For the instance of Twitter, around 9 to 15 percent of accounts are bots that contribute to content and advertisement revenue [185]. The complexity of using language on social networks and active involvement of automated agents and bots, seizing on spreading certain topics, political agenda, and fake news call for novel and explanatory approaches for detection and analysis of viral topics on social networks [28–30]. In this respect, formation of *echo-chambers* among users (i.e. retweeting and replying) emphasizes the effect of information source on developing content by users [186], i.e. usual users act according to perceived behavior from influential users.

This chapter explores the connection among behavioral metrics of influential users and evolution of content and discourse on a social networks. The scope of this chapter is limited to analyzing and comparing two distinguished types of influential behavior in terms of their perceived contents in a network. First, *exogenous* influence appears in sharing information by a non-interactive and external source such as a news media outlet or to some extent a journalist. In contrast, *endogenous* influence is exercised through interaction and discussion through the platform among influential users, i.e. hop, source, and lead users [187]. These observed types of behavioral influence by users are: 1) *Media-driven influence (MDI)* that is associated with exogenous influence through introduction, sharing links, and summarization of topics originally external to a platform and 2) *Interaction-driven Influence (IDI)* that is associated with interactive behaviors such as retweeting, replying, and discussing topics on the platform among others. For instance, skimming the twitter feed of CNN and Brian Sollis (@briansolis) roughly clarifies some differences between those categories: news-oriented vs interaction-oriented, one-to-many vs many-to-many communication, one-sided vs responsive, etc. Nonetheless, individuals may also be classified as *MDI* when their main activity is spreading preprocessed news from news outlets (e.g. @DavidNakamura) while some news outlet sources might be considered as *IDI* for the opposite reason (@AJCCenter).

This chapter investigates the application of mechanism design to drive micro-behavior toward a collective goal in interactive social networks. The research problem mainly comes from recent incidents on fake news and bot-driven content, propaganda, and advertisement in these networks.

To the author of this thesis, the observed problem is mainly driven from three factors: 1) social network has provided an inexpensive medium for users and other participants to contribute in spreading content based on their opinion, 2) a social network is extremely effective and influential in lives of millions or billions of people in terms of its collective results (i.e. social welfare or global utility) for all participants, and 3) users as the drivers of a global welfare don't have control or knowledge on the effects of opting a specific micro-behavior on the collective welfare affecting themselves, i.e., users might hurt themselves without indention. The first components has been the most significant results of introducing world-wide web in 21^{th} century as the a platform that enabled many-to-many connection among users and participants in terms of ability to create and distribute content. Then, creating and distributing content is as cheap as the time spending on them except that *social influence* remains expensive while the mechanism that results in higher influence is driven from a more distributed and dynamic parameters, e.g. connections to other influential users, attention-driven behavior, timing, distributing content, lifestyle, etc. Second, the social networks are proven to easily affect or dominate the news cycles in current world which ultimately affect millions of lives through politics, social movement, and economics. Nonetheless, users are usually blinded regarding the collective effects created by their own actions in a social network. In other words, users act in a certain way partly because they don't know the social cost of a naive/myopic behavior. For instance, the cost of spreading fake news in a social network is hidden from regular users while the nature of spreading is mainly because of naivety of regular users in accepting and distributing (i.e. confirming) a controversial but viral topic on a network.

Sec. 5.2 reviews related works on interactive models, agent-based social simulation, and clustering models of topics in social networks. Sec. 5.3 introduces a method to classify users in two behavioral categories of MDI and IDI based on their contribution features. Sec 5.4 introduces a novel clustering technique to discover viral topics based on a network of terms and concepts. Sec. 5.5 introduces and discuss a temporal connections among viral topics and Sec. 5.6 describes data collection, research workflow and introduces three metrics to compare topics and visualize results.

5.2 Literature and Problem

More than a decade after invention of world wide web, we started communicating through *many-to-many* and interactive online social platforms. Multiple studies have developed models to understand interaction mechanisms among social actors and communities during this era [98–102]. These interaction networks are used to discover attitude, emotions, perception and sentiment associated with a content [23,106–108]. In addition, temporal interactions could reveal and predict a community structure, membership behavior of users, and profile a user by its behavior including identifying bots or agents in spreading fake news [109–114]. In developing commercial applications, online interactions among users and analysis of topics have given insight into discovering brand reputation and political orientation [123, 124].

For analysis of topic and discourse on social networks, Davis et al. develop a ranking model for finding prevalent topics on Twitter [188]. Cigarr et al. present an approach using Formal Concept Analysis (FCA) to distinguish interest groups regarding products and brands on social networks [189]. Lipizzi et. al. use a graph-based approach using adjacency matrix of concatenation among keywords to identify real-world discourses expressed through back-channeling on social networks [190] where a similar approach can cluster users based on trending topics [191]. Xie and Mathioudakis employ the concepts of popular and *bursty* keywords to detect topics in real-time [192, 193]. Crane et al. differentiate between exogenous and endogenous topics [194] and other studies compare topics from Twitter to conventional online media such as New York Times, Google trends, and CNN base on endogenous and exogenous influence on users [195–197]. Network models of users, concepts and documents have been employed to understand the dynamics of community and content development on social network [117]. Networked influexntial users and concepts help to calculate the effectiveness of WikiProjects in online content development [118] and the structure of knowledge among computer science venues [119]. In addition, network statistics, such as centrality, closeness, betweenness, and entropy are vastly employed to explain the interconnectedness of communities and concepts in social networks [120–122].

Agent-based models are developed to facilitate theory-building, explain and analyze real-world scenarios, and obtain policy recommendations in support systems [198]. Agent-based Social Simulation (ABSS) is a popular approach to study a complex social phenomenon eluding analytical or empirical methods [199]. ABSS models must be as simple as possible and, at the same time, describe reality, i.e. “keep it descriptive stupid” (KIDS) [199]. In other words, these models allow simple rules of behavior in individual level that lead to a complex and aggregated behavior observed in a collective system. The behavioral models involve metrics for interactions of agents with environment at each time step, e.g. stochastic process with some probabilities, production rules, density functions, and machine learning [200]. In terms of applications, an agent-based model is used to analyze and maximize diffusion of commercial messages by companies on social networks [201] and machine learning models classify messages to rumors versus non-rumors based on temporal, structural, and linguistic characteristics [202].

5.2.1 Research Questions

This chapter approaches a research gap between behavioral model of users and diffusion model of content in social networks. The goal of this chapter is to better understand a statistical dynamics of users and viral topics and to propose a framework to analysis connect micro-level model of interactive users to macro-level model of content in a social network. In this research, to understand the connection between between these two models, I address three research gaps in literature:

A1 statistical relation between micro-level actions and macro-level collective observations in terms of social modeling

A2 user actions and introduce (discover) collective metrics that capture and explain those behaviors

A3 an interactive model to explain and potentially predict interactive connection above behavioral metrics

The research question addressed in this chapter is: *How can exchange mechanisms for human resources and information contribute to better collective metrics in interactive and social networks?*. This question can be disaggregated into:

Q1 How to model interaction in social networks?

Q2 How to detect and model spreading viral topics on social networks?

Q3 Which features describe social behavior of participants in interactive networks?

Q4 How micro-level behavioral model affects the macro-level content model?

5.2.2 Research Methodology

The main methodology in this chapter is data-driven and statistical analysis of user and network behavior to: first distinguish and classify users in terms of their interactions and contributions, second, analyze discourse and content in a social network. The following steps address the above questions in this chapter:

S1 analyze and define the concept of influence in social networks

S2 query and retrieve data for most influential users on Twitter

S3 develop a statistical model to classify types of influence in social networks using activity-based data

S4 develop a content and discourse model for analyzing information exchange in a network

S5 develop a clustering framework to detect viral topics

S6 validate the clustering model using real-world comparative analysis of viral topics

S7 find statistical connection among the user-model of influence and the content-model of information exchange

For the proposed researches in future, the proposed methodology is simulation study when an interactive mechanism for driving collective behavior of a network toward a collective metric may include analytical solutions and mechanism design.

5.3 User Classification

In this section, we distinguish two classes of behavior by users in terms of their interaction level and type of contribution. First, online activities of a user can be captured by her tweets, e.g. length and links, retweets, and replies. This introduces a model to classify users to two classes (*MDI* and *IDI*) and extract new accounts on twitter associated with each class.

The classification model is a Logistic Regression (LR) model with number of retweets, number of replies, number of shared links, and median text-length as predictive features:

$$g(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$$
$$\pi(x) = \frac{e^{g(x)}}{1 + e^{g(x)}} \tag{5.1}$$

In the first phase, I select 170 accounts with *MDI* and *IDI* characteristics noticing that these accounts are recognized by monitoring online activities of accounts in terms of sharing personal stories, direct communication with others, i.e. their organic interaction on social media. For instance, New York Times is labeled as 1 (MDI) and interactive users such as Bill Gates is labeled as 0 (IDI).I train the logistic regression (LR) model in Eq. 5.1 using the selected accounts (training set). In second phase, I extract accounts that have been most influential on those selected accounts (e.g. being retweeted by them the most) and select second batch of data set consisting of 170 new accounts. I label the latter data and add it to training data set, update the logistic model and repeat the same processes of *extracting*, *model updating* and *labeling* the most influential accounts until I collect more than 1750 accounts. In sum, 10% of accounts were labeled manually and the other 90% were collected and labeled iteratively (in mini-batches) using above LR model.

x_i variable	Interc. (0)	Retweets (1)	Replies (2)	Links (3)	Text median length (4)
β_i coeff.	-0.96	0.35	-1.76	2.82	0.61

Table 5.1: Coefficients of Selected Features in *LR*

Table 5.1 shows the selected variables and their calculated coefficients. The positive values for *retweets*, *shared links* and median *length* of tweets imply that these variables are more associated with MDI than IDI while number of *replies* is more associated with IDI than MDI.

Fig. 5.1 shows the probability distribution of accounts using *LR* model. The probability distribution is more skewed towards *IDI* with a sharp peak on $p = 1$. This implies higher certainty around labeling the former set, i.e. tweeting behavior of a journal is more predictable than that of an indi-

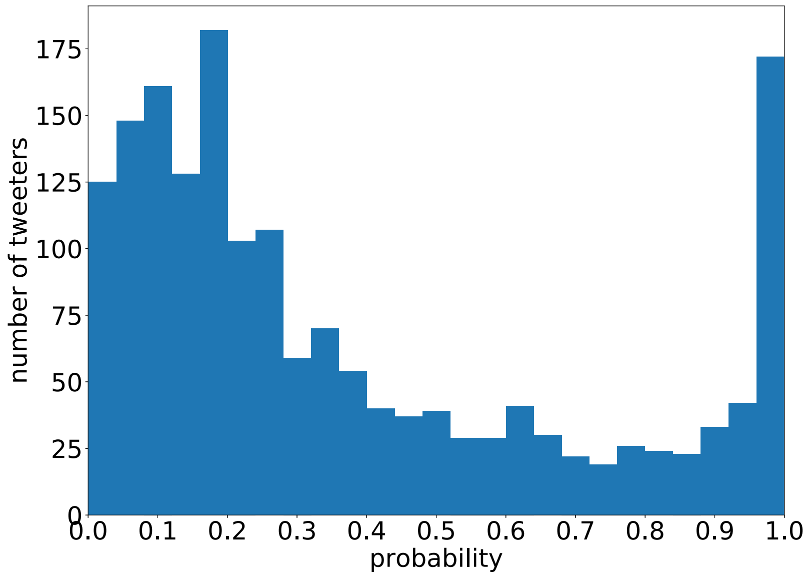


Fig. 5.1: *Logistic regression*'s probability distribution for 1580 unlabeled samples among 1742 total accounts. The right side is associated with class 1 or *MDI* and left side relates to class 0 or *IDI*. The resulted distribution is skewed towards interactive (*IDI*: left) behavior with a peak around the maximum probability. $p = 0.7$ is selected as the threshold for *LR* classification.

vidual. The minimum point of $p = 0.7$ is used to assign labels to accounts which also results in 20% of users being labeled as *MDI* and the rest being recognized as *IDI*. In sum, user accounts are labeled into 352 and 1398 sources with exogenous and endogenous influence. Although the number of former accounts are significantly lower than the latter accounts, but the aggregated activity level of two groups are equivalent and very close because *MDI* users publish and share more tweets and distribute more contents on average.

5.4 Clustering Model

A frequentist models of words are vastly used to cluster emergent topics in social networks. In this respect, topics are both popular and scarce: first, a new topic is relatively viral and under discussion across a network, second, it may not has been as much discussed in the past or under regular circumstances. In *tf-idf*, a widely used method for extracting new topics, the usage-frequency of a word in combination with the inverse-frequency of documents including the word define the relevance between a topic and a document. In social networks, a user may apply unconventional wordings, phrases, hashtags and abbreviations to efficiently communicate her message, thereby, a networked model of terms and words are applied to reconstruct grammar in analysis of discourse [190, 203]. In this chapter, I also opt to employ the networked structure of language for clustering viral topics. According to a graph-based model, a term A (e.g. word, hashtag, abbreviation, keyword or compound word) is represented as a node when the frequency of its usage in combination with the second term B in shared occasions imply the strength of connection between these terms in developing new topics across a network.

The method introduced by this work is aimed at being efficient in detecting new topics, illustrative and intuitive for analysis of those topics. To discover the strength of a link between terms using term-pairs, similar to the frequency case in *tf-idf*, I devise two metrics: frequency of a pair and inverse of expected frequency of the pair in a time frame, i. e. lower value for higher expected frequency. Feng et al. (see [204]) defined the *popularity* of an event as the normalized frequency of that event by number of tweets and and burstiness of an event as the standardized popularity by the popularity of the same event achieved during temporal time frames. In this work, I introduce similar definitions for graph edges among all used terms during a time frame.

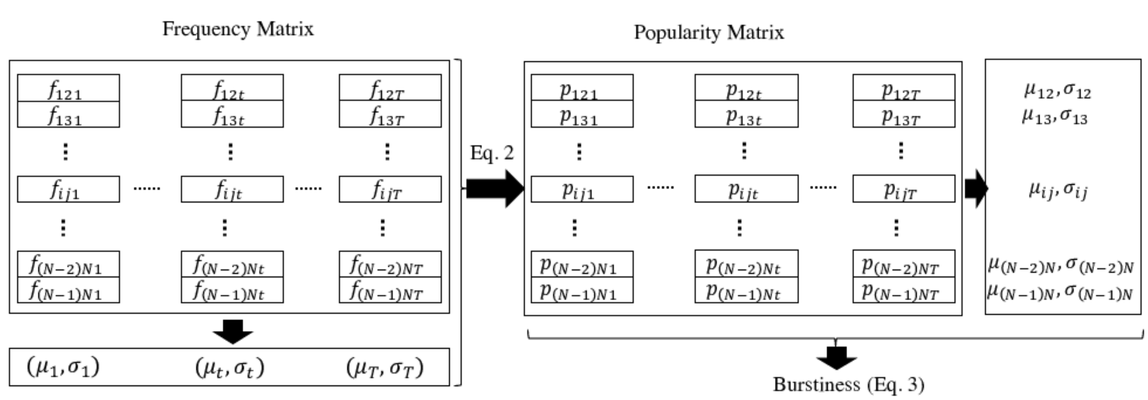


Fig. 5.2: Clustering metrics of *popularity* and *burstiness* using the frequency of links among terms, words, hashtags, abbreviations and emojis: the total number of possible connections are defined based on the number of words: $N(N-1)/2$ and shows the number of rows in these matrices. The *burstiness* for each connection is defined as the standardized popularity across all time frames.

Assume that $W_t = \{w_1, w_2, ..., w_N\}$ are the number of all terms (i.e. words, hashtags, concepts, and compound words) during time frame t, $D_t = \{d_1, d_2, ..., d_M\}$ are the documents (e.g. tweets, post updates) published during the same frame, and:

$$F_t = \{f_{ijt} : w_i, w_j \in W_t\}$$

consists of frequencies of all defined connections among those terms where the cardinality of F_t is the number of possible connections among N words: $|F_t| = N(N-1)/2$. The *popularity* of link normalizes its usage by the number of tweets during each time frame:

$$popularity : p_{ijt} = \frac{f_{ijt} - \mu_t}{\sigma_t}, i \in \{1, ..., L\} \tag{5.2}$$

where μ_t and σ_t are averaged and standard deviation of frequencies in F_t at time frame t. The latter equation which leads to an array of normalized popularity for existing links. The *burstiness* standardizes popularity by the average values and standard deviation for the same connection among all temporal frames:

$$burstiness : b_{ijt} = \frac{p_{ijt} - \mu_{ij}}{\sigma_{ij}} \tag{5.3}$$

where $\mu_{ij} = \sum_t p_{ijt}/T$ is the averaged value of popularity of link between w_i and w_j across T time frames and σ_{ij} is the standard deviation of those frequencies.

Fig. 5.2 shows the algebraic steps in calculation of *popularity* and *burstiness*. While popularity (Eq. 5.2) uses the direct columns of frequency matrix in time (each column represents frequencies for all connections at one time step), burstiness formula (Eq. 5.3) uses popularity matrix. A weighted average of popularity and burstiness for inter-word links gives the relevance *score* for every connection:

$$relevance : r_{ijt} = \alpha p_{ijt} + \beta b_{ijt} \tag{5.4}$$

A similar linear combination of above metrics in addition to *localness* was called *ranking score* in [204]. In this work, the matrix of connection scores among terms is called *relevance matrix* which is close to definition of *similarity* matrix in literature. We use a thresholds (e.g. 99^{th} percentile) to convert the *relevance* matrix to a sparse *adjacency* matrix.

$$a_{ijt} = \begin{cases} 1, & \text{if } r_{ijt} \geq threshold \\ 0, & \text{otherwise} \end{cases} \tag{5.5}$$

5.5 Network Model of Topics

This section introduces a mathematical model to distinguish the behavioral aspects of cluster formation among *MDI* and *IDI*. Assume R_i being defined as vector of score-weights between term i and others where each score shows a relevance-score:

$$\mathbf{R}_i = (r_{i1}^+, r_{i2}^+, ..., r_{iN}^+)$$

where:

$$r_{ij}^+ = \begin{cases} r_{ij}, & r_{ij} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{5.6}$$

In other words, $r_{ij}^+ = 0$ when two terms are not connected or are weakly connected at time t and $r_{ij}^+ > 0$ otherwise. For notational simplicity, I dropped time index t from r_{ijt} while each $\mathbf{R}_i$ is defined for a time frame. The *similarity* formula among documents is defined

In literature, a similarity between terms namely *text-weighing similarity* formula is proposed by Salton in [205]. This formula is also applied to compare texts, hashtags and documents in [204,206] and [119]. Accordingly, *term-similarity* is defined as:

$$similarity(\mathbf{R}_i, \mathbf{R}_i) = \frac{\mathbf{R}_i.\mathbf{R}_j}{||\mathbf{R}_i|| \times ||\mathbf{R}_j||} = \frac{\sum_{k=1}^{N} r_{ik}^{+} \times r_{jk}^{+}}{\sqrt{\sum_k (r_{ik}^{+})^2} \times \sqrt{\sum_k (r_{jk}^{+})^2}} \tag{5.7}$$

In this paper, we aggregate similarities of all links between two topics and define *topic-similarity*. Assume that topic v is defined using $\mathbf{W}_v$ and its corresponding vectors:

$$\mathbf{C}_v = \{R_i : w_i \in \mathbf{W}_v\}.$$

Topic-similarity is defined as as:

$$topicsimilarity(\mathbf{C}_v, \mathbf{C}_w) = \sum_{\mathbf{R}_i \in \mathbf{C}_v} \sum_{\mathbf{R}_i \in \mathbf{C}_w} similarity(\mathbf{R}_j, \mathbf{R}_i) \tag{5.8}$$

5.6 Empirical Results and Discussion

For empirical study, I collect and process more than 6,250,000 tweets published or retweeted by 1742 influential accounts on Twitter from August 2017 to March 2018. 355 accounts belonging to journalists, economists, scientist, news organizations, activists, etc. are selected as initial seed of influential users[2] and another 1395 accounts from the most retweeted accounts by the initial seed during 210 days of the time period are algorithmically retrieved. Extracting data from Twitter API was sequential (170 user at a time) to regularly update our list of top influential accounts as each additional set of accounts determined the next set.

For each tweet, I stripped text from frequent words and punctuations using *stopwords* reposi-

[2]using websites such as *time*, *politico*, *sciencemag*, etc.

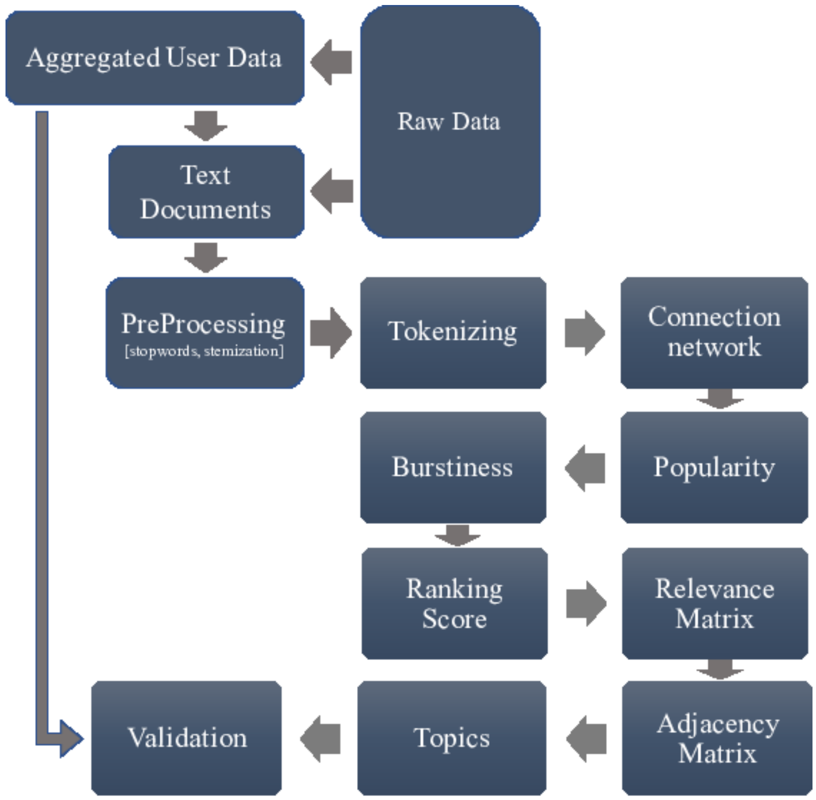


Fig. 5.3: Research workflow for classifying users, processing data, clustering topics and model validation.

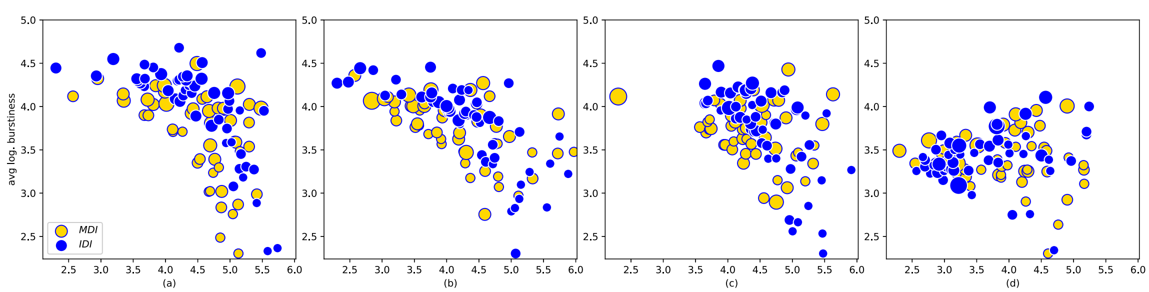
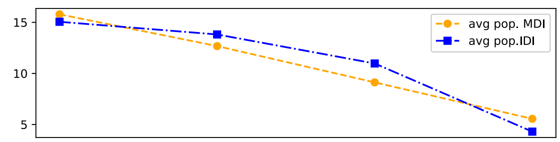


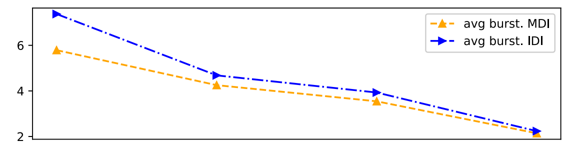
Fig. 5.4: *Burstiness* vs *popularity* for 100 clusters of terms at each temporal granularity (50 cluster of each user class with highest *cluster score*). The circle-size shows the topic consolidation in terms of relevance (see Eq. 5.4) among words in a cluster. Term-clusters among *MDI* are smaller and more consistent in terms of popularity and burstiness while term-clusters resulted from tweets by *IDI* accounts include relatively larger circles with more diverse sizes, which implies more distributed clusters with various strengths and consolidation for temporal granularities of: *a) 1 day:*, *b) 3 days:*, *c) 7 days:* and *d) 21 days:*. The clusters emerged from *IDI* accounts in longer temporal moves towards the *MDI*s as mutual effect among two classes of users emerge.

tory and retrieved stemmed words using *nltk* toolkit[3]. These processes significantly reduced the number of effective terms and computational complexity of graph-based model of terms in memory

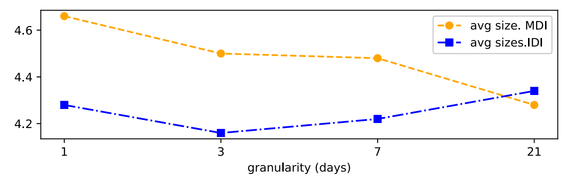
[3] http://www.nltk.org/howto/stem.html



(a) Popularity



(b) Burstiness



(c) Cluster Size

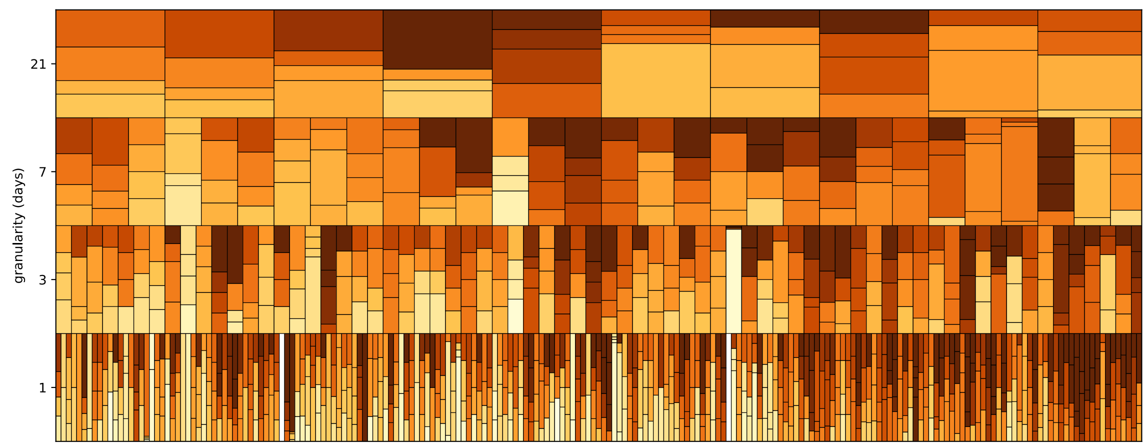
Fig. 5.5: Averaged popularity, burstiness, and size of topics for temporal granularities of 1, 3, 7 and 21 days for the top 100 clusters illustrated in Fig.5.4; (a) *Popularity:* averaged popularity is decreasing with duration of granularity-window time frame while the relative popularity is higher for *IDI* except for longest time frame (b) *Burstiness:* averaged burstiness is also decreasing with duration of granularity-window time frame and is relatively higher for *IDI* (c) *Cluster Size:* averaged cluster size is lower for *IDI* among topics with highest scores except for 21-day time frame.

and time. For instance, each stem word aggregates 4.79 different words and 5000 stemmed terms are equivalent to 23950 terms. In sum, these processes reduced the number of effective statuses (by 15%) as it eliminated those without linguistic content e.g. photos, videos and links. For the graph-based model, I selected 5000 as the maximum dimension associated with tokenizing the documents. The selected words are the words with highest frequencies across all documents. Using *scipy* library on Python 3.6, we create sparse matrix of link-frequency when each link between terms A and B is defined as the number of tweets that include both A and B.

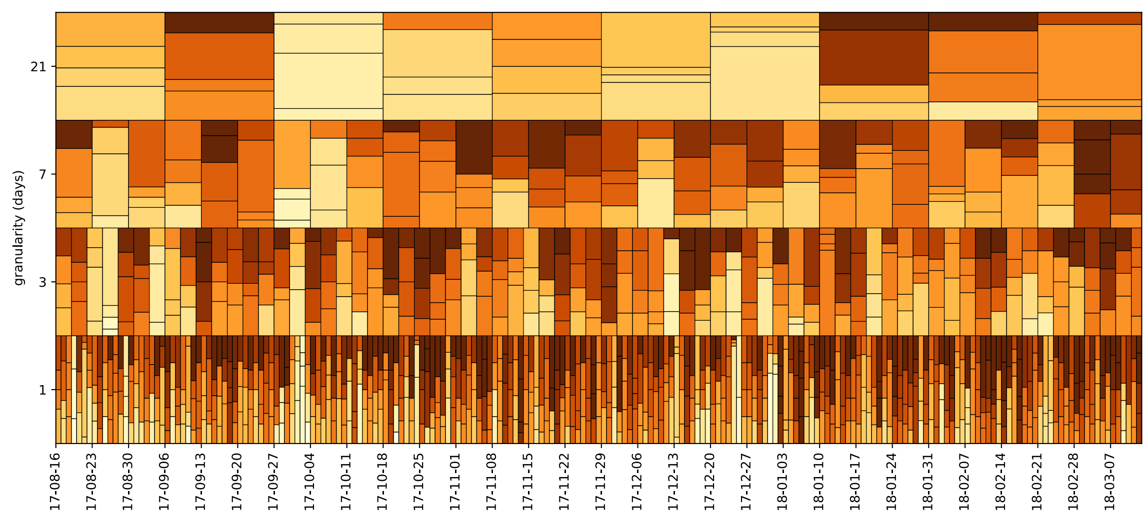
For graphical results and clustering model validation, the three metrics introduced include: cluster *size*, *consolidation* and normalized *score*. In summary, assuming W_k as the set of all stem-words in topic k, $|W_k|$ is cluster size and:

$$consolidation_k = \{l_{ij} : r_{ij} \neq 0 \cap w_i, w_j \in W_k\} \tag{5.9}$$

$$nscore_{[W_k].s} = \frac{\sum_{w_i,w_j \in W_k} r_{ijt}}{|W_k|} \tag{5.10}$$



(a) MDI



(b) IDI

Fig. 5.6: Dominant topic clusters for different temporal granularity: (a) *MDI:* topic clusters among the most frequent terms by twitter accounts with media-driven influence, (b) *IDI:* topic clusters across user network with interactive influence. The clustering analytics is implemented on the connections (i.e. usage links) among top 5000 most frequently used keywords, terms, hashtags and english words. The links with most (99 percentile) *relevance* score are selected in order to achieve sparse matrix for *spectral* algorithm. The darker color shows higher *cluster score*and the relative vertical height indicates higher cluster *consolidation* among the cluster terms.

In sum, Fig 5.3 enumerates the steps for user classification, data processing and clustering topics. In this section, we define three models for visualizing data on clusters, viral topics, and a networked model of similarity among topics:

5.6.1 Popularity vs Burstiness

In Fig. 5.4, the relative *popularity*, *burstiness* for top 50 clusters in terms of their relative *score* for each user-class of *MDI* versus *IDI* are shown. In each figure, x-axis represents popularity and y-axis represents burstiness where both are relative to cluster size and logarithmic and a circle radius represents its cluster *size*. The temporal granularity includes 1, 3, 7 and 21 days, from left figure to right. The y-axis has equal range across all temporal frames for better comparison (2.2 to 5). Although these figures show the logarithmic and relative values, I use relative and non-logarithmic values of popularity, burstiness and scores for discussion.

The two dimensional plot of popularity versus burstiness can distinguish the outliers among other clusters and also distinguish some behavioral differences among *IDI* and *MDI*. In 1-day granularity, the highest scores for these two categories belong to news regarding children losing their healthcare (Oct. 2017) and jail sentence for Larry Nassar from Michigan state university (Jan 2018), respectively. Among 3-day time frames, we notice an outlier with minimum burstiness among *IDI* which relates to an advertisement by Samsung galaxy for discount (Nov). The maximum score for the group also relates to the same topic while the maximum score of *MDI* is regarding the 100 refugees entering to US (Dec). The viral topic developed by the latter group among 7-day time frames relates to Robert Mueller's investigation of Paul Manafort's financial investments (Feb). Nonetheless, the hottest topic among regular users are lightness of Samsung galaxy (March). The outlier with minimum popularity and high burstiness on left of figure relates to an update regarding the stock market's worst correction on Feb 3. Finally, among 21-day frames, we can notice the Dr. Martin Luther King Jr. as the most viral topic among regular users (Jan) and NYC's mayor on terrorist suspect (Oct) among journals. Although the phone advertisement has received highest popularities and scores among topics, it has the lowest burstiness among all topics and time frames. This indicates that commercial product with a viral subject that receive high level of popularity and relatively lower burstiness might've been promoted for months. The detected topics above imply propagation of political news through journals and development of more diverse subjects including social events and tech news by interactive users.

Fig. 5.5 shows averaged values for all three metrics among the top 100 clustered topics in terms of score among *IDI* and *MDI*. The averaged popularity values are higher for *IDI* on two middle time frame granularities: 3 and 7-day frames. Also, the averaged burstiness is consistently higher for the latter group while the averaged cluster size is higher for *MDI* except for 21-day time frames. The smaller size and higher burstiness for most viral topics among others by the former group imply that interactive users change topics more often and focus on more concise and diverse set of topics.

5.6.2 Viral Topics

Figure 5.6 visualizes significant clusters in terms of relative score (i.e. viralness) across time frames. We can already notice the concentration of topics for *IDI* and *MDI* among middle (3 and 7-day) and longer (21-day) time frames in order. In those figures, each rectangle shows a topic, darker color represents higher score and rectangle height represents cluster size. I start with the most noticeable layer of granularity: 21-day time frames. Among the *MDI*, the most viral topics on this time frame are suspected terrorist act in NYC around late October, marriage of prince Harry on mid-November, Michael Wolff's book named Fire and Fury on late December and attorney general being interviewed for Russian investigation on late January. Eight out of 10 most popular topics, one per time frame, are political topics. The exceptions are Samsung advertisement in early December and court hearings by victims of Larry Nassar in early February. Among *IDI* users, the most viral topics in the latter time frame included an application advertisement, namely word correction application, on September, Martin King Jr. on late January. In this respect, four topics out of the top ten hottest topics related to social news where two topics related to technology and another two to lifestyle and one to politics. Similarly, observations on 7 and 3-day time frames confirm that more diverse and less political topics are viral among interactive users.

5.6.3 Network Model of Topics

For a network model of clustered topics, I apply the *similarity* formula introduced in Sec. **??** to calculate similarity-weights for edges between topics as vertices in the network. Although topics are more populated in longer time frames, the total number of clustered topics are higher for shorter windows. Then, the best approach to flexibly visualize the network fo clusters is a circular network with topics belonging to the longer frames being closer to the center. To manage the complexity of weight calculation among topics, I limit the similarity edges to *hierarchal* and *temporal* links. By the former, I calculate similarity between a topic with other topics located on overlapping time frames, i.e. its higher or lower time frames on different granularities. By the latter links, we imply the similarity between a topic and other topics distinguished on the same granularity but different and close time frames. For an instance of hierarchal similarity, a clustered topic on 3-day time frame with center of `2018-02-01` has hierarchal links to topics on 1-day time frames of `01-31`, `02-01` and `02-02`, 7-day time frame of `02-03` and 21-day time frame of `02-10`. For an instance of temporal similarity, the same topic has weights to topics on 3-day frames centered around `01-26`, `01-29`, `02-04` and `02-07` (see Fig. 5.6). While the number of clustered topics for 210 days of tweeting data are 2954 and 2511 in user with *IDI* and *MDI*, by using this method, we reduced the number of weight calculations from (8.7, 6.3) million links to (32756, 24475) links for *IDI* and *MDI*.

Fig. 5.8a and 5.8b show hierarchal networks for *MDI* and *IDI* where each vertex is combination of clustered topics in a frame and its size is proportional to sum of cluster sizes. The left figure shows denser connection among nodes, particularly the connections with nodes on the 21-day granularity. This observation implies that accounts with journalistic behavior focus on fewer topics in comparison to accounts with interactive behavior which are visible across different granularities. This is also consistent with the observations in Sec. 5.6.2. Fig. 5.9 and 5.10 show the temporal

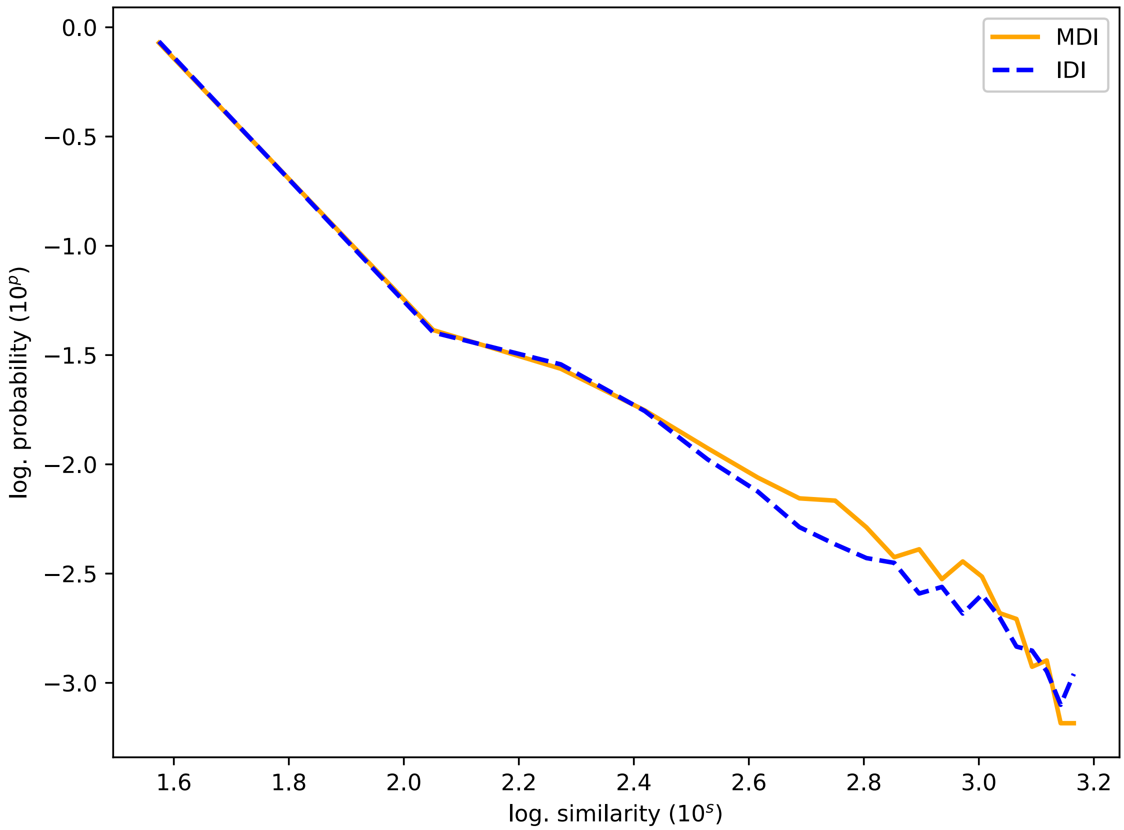
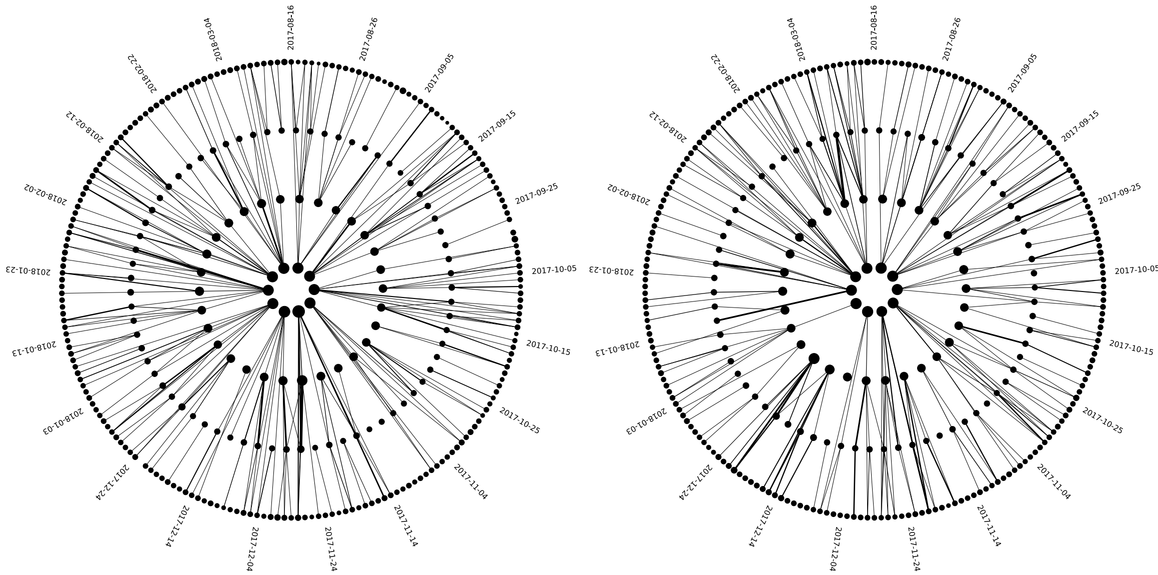


Fig. 5.7: Logarithmic probability distribution corresponding to calculated logarithmic weights of similarity among clustered topics. Assuming scale-free distribution of weights: $P(k) \approx k^{-\gamma}$, we find $\gamma \approx 1.87$.

similarities between consecutive topics where each circle contains all clustered topics at a time frame. The number of visualized topics are limited according to its time frame: (4, 6, 8, 10) for (1, 3, 7, 21)-day frames. At each frame, the topics are sorted based on their relative score as the cluster with highest sore is closest to the inner circle. The network edges show the similarity weights calculated using Eq. 5.8. In these figures, opposed to the hierarchal case, we observe denser connection among clustered topics of *IDI* accounts.



(a) MDI (b) IDI

Fig. 5.8: Hierarchal similarity between aggregated topics on different time frame granularities: (a) *MDI:* topic clusters among the accounts with media-driven influence, (b) *IDI:* topic clusters across user network with interactive influence. Every node shows a time frame where its size the sum of size of all clustered topics the its corresponding time frame.. The inner to outer circles show the clustered topics on 21, 7, 3 and 1-day time frames. Data time increases clockwise from north between `2017-08-16` and `2018-03-13` corresponding to 210 days.

5.7 Discussion and Conclusion

In the networked model of topics in the previous section, a visual connection is established between structural behavior of users and temporal model of viral topics and discourse on Twitter. In this section, I visualize and analyze statistical metrics of topics for the distinguished classes of users during this chapter.

5.7.1 Statistical Analysis

The complete topic networks with weights are still too complicated for the purpose of visual comparison between classified accounts. Fig. 5.7 shows the log-distribution of similarity weights among clustered topics. Accordingly, I limit the visualized weights to $similarity > 100$. Further, I separate the hierarchal weights from the temporal weights. Fig. 5.8 visualizes the first network where each

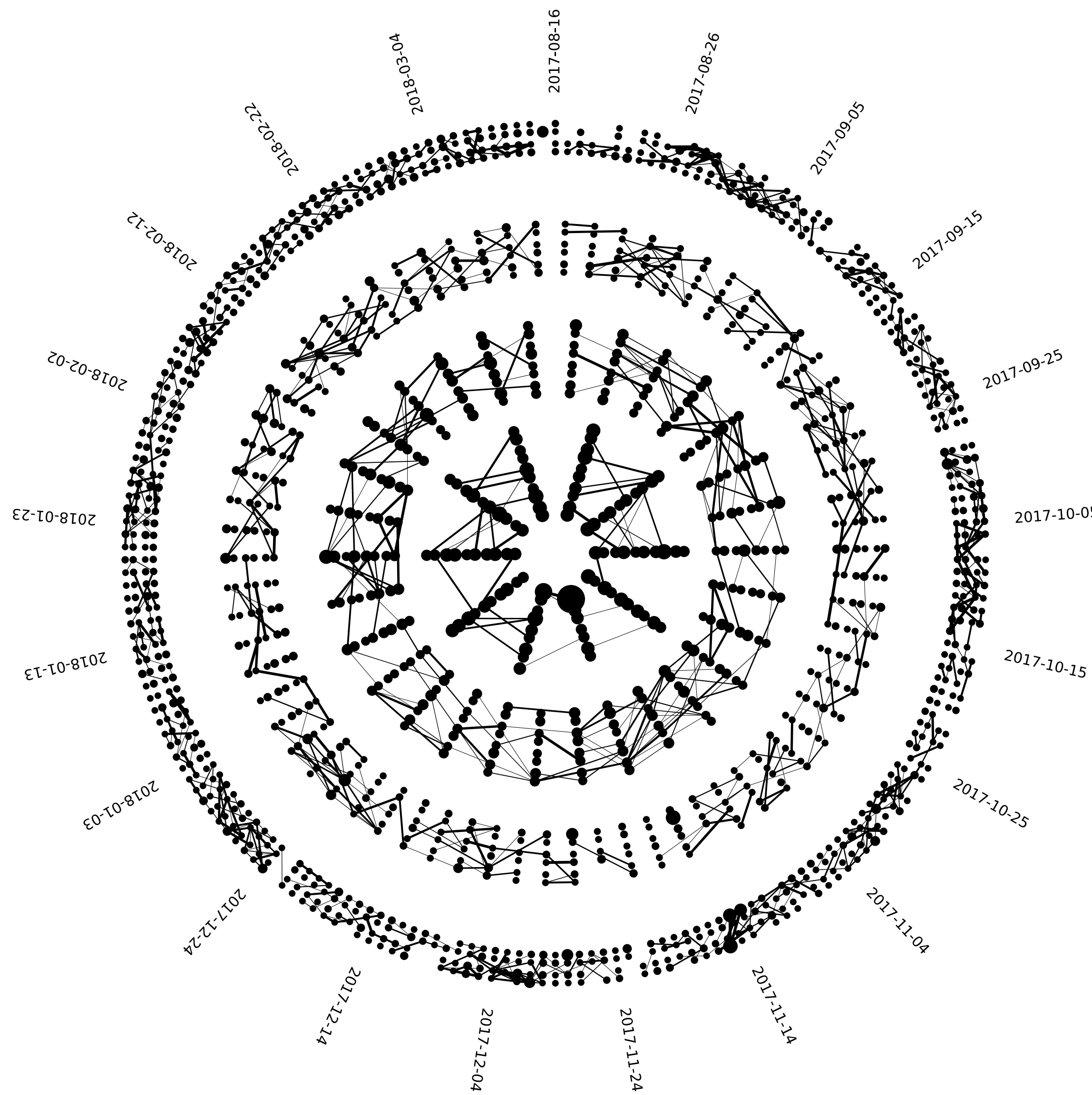


Fig. 5.9: Temporal similarity between clustered topics on consecutive time frames for MDI. Every node shows a topic with its cluster size. The inner to outer circles show the clustered topics on 21, 7, 3 and 1-day time frames. The line width between two topics is proportionate to the calculated similarity between them. Data time increases clockwise from north between `2017-08-16` and `2018-03-13` corresponding to 210 days.

vertex represents the combination of all clustered topics in a time frame. The inner circle contains topics on the time frame with least granularity and the outer circle contains those on daily time frames. Assume that $TF = \{C_i\}$ is a set of all clustered topics in a time frame, the weights of edges in this network are calculated as sum of topic-similarity weights between two hierarchal time

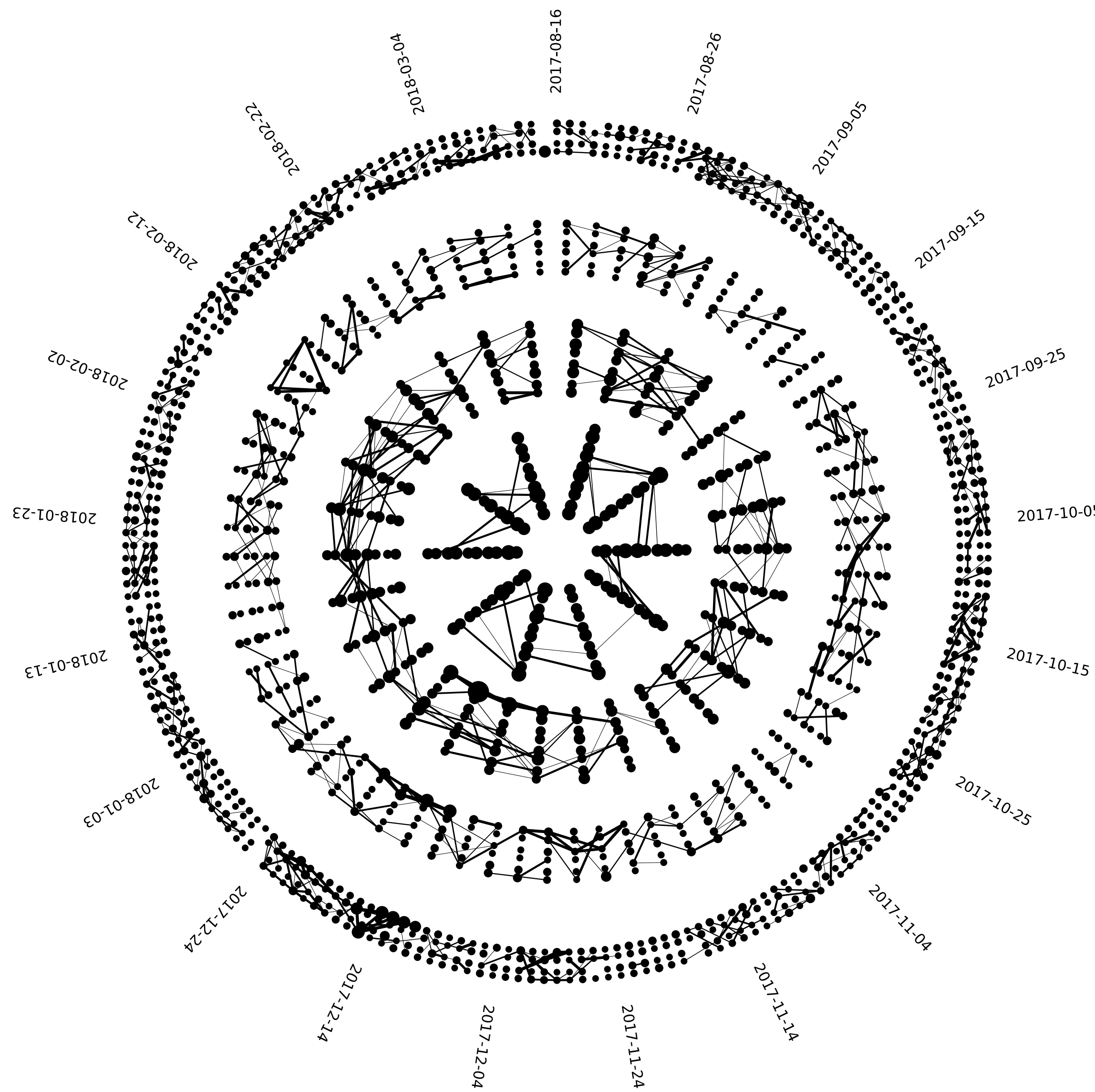


Fig. 5.10: Temporal similarity between clustered topics on consecutive time frames for IDI.

frames:

$$weight(TF_i, TF_j) = \sum_{C_v \in TF_i} \sum_{C_w \in TF_j} similarity(C_v, C_w) \tag{5.11}$$

Fig. 5.11 shows multiple network statistics across different granularities. Accordingly, the number of clustered topics are higher for users with *IDI* behavior. Nonetheless, in Fig.5.11b, the number of temporal links with $similarity > 100$ shows higher weight among topics of *MDI* for lower granu-

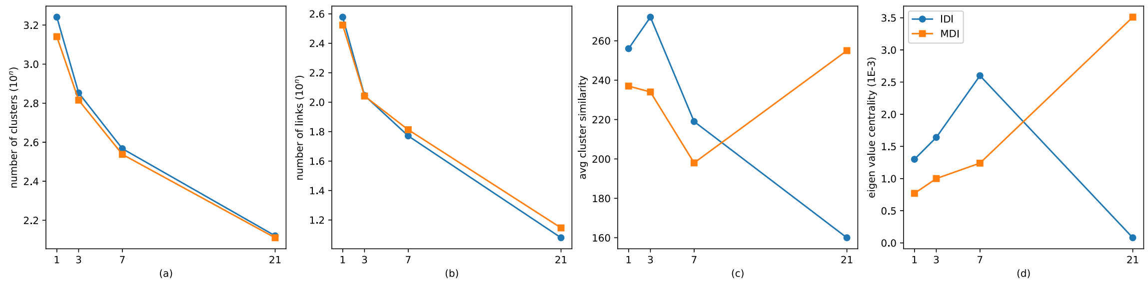


Fig. 5.11: Network statistics for *IDI* and *MDI* and different temporal granularities: **(a) number of topics:** the number of clustered topics in the 210, 70, 30, and 10 possible 1, 3, 7 and 21-day time frames. Averaged number of clusters per time frame are 9.2 and 7.8 for *IDI* and *MDI*. **(b) similarity links:** the number of temporal similarity links between topics with $similarity > 100$. **(c) avg similarity:** the averaged temporal similarity between consecutive topics. **(d) avg centrality:** the averaged centrality of vertices in the networks clustered topics and similarity links.

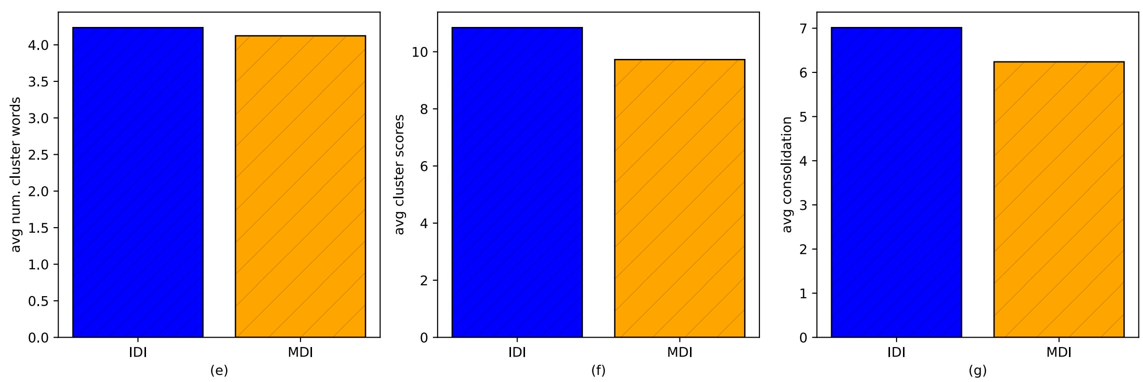
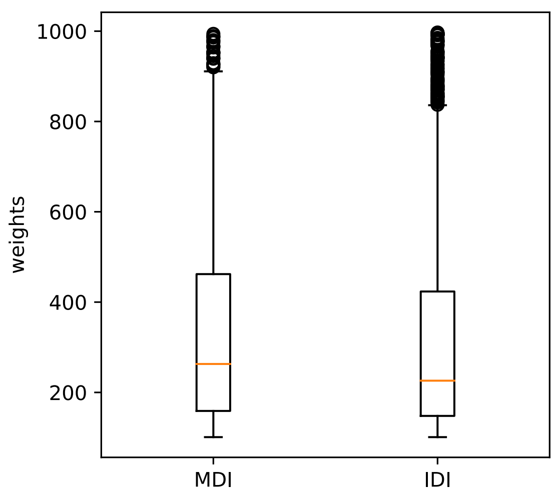


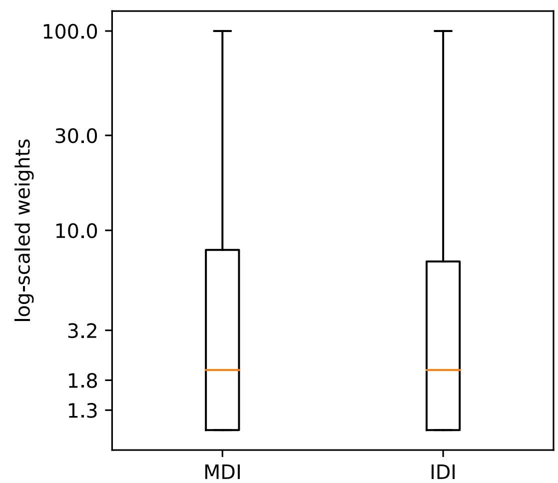
Fig. 5.12: Network statistics for *IDI* and *MDI*: *(e) averaged cluster sizes:* for number of words in clustered topics, *(f) averaged scores:* for clustering score of detected clusters and viral topics, and *(g) averaged consolidations:* for the number and weight of connections in each cluster.

larities. In the latter figure, we expect more links among topics of *IDI* accounts with higher number of clustered topics but this observation suggests that more various topics are discussed among these users. The averaged similarity among the temporal links in Fig.5.11c shows that consecutive topics are more strongly similar among *IDI* accounts. This behavior suddenly changes for the least granularity as the topics are more consistent for 21-day granularities. We observe similar trends regarding the centrality of topics Fig.5.11d.

Fig. 5.12 shows a number of aggregated statistics. The number of *stemwords*, i.e. words, terms, hashtags and compound words, per topic is slightly higher for *IDI*. In Fig.5.12f, averaged cluster scores are also higher for those users. This implies that finding discussion topics are easier among interactive users. The next figure shows the averaged *consolidation* introduced by Eq. 5.9 and indicates the number of links among the terms in a cluster, normalized by its size. According to Fig.5.12h, the centrality of nodes is also higher for *IDI* which is counterintuitive given the higher or comparable number of links among topics discussed by the second group. Finally, the averaged temporal similarity is higher for interactive users while the averaged hierarchal similarity is higher for *MDI* as we observed visually in Fig. 5.8, 5.9 and 5.10.



(a) hierarchical



(b) temporal

Fig. 5.13: Similarity weight distribution for *MDI* and *IDI*: (a) hierarchical similarity-weights for clusters links with $simil \geq 100$ among topics and (b) temporal logarithmic-scaled similarity-weights for cluster links with $simil \leq 100$ among topics.

The weight distribution of similarity metrics has lognormal distribution. The statistical analysis of hierarchical connections among clustered topics shows significantly higher weights among developed topics by *MDI* than those by *IDI* although the frequency of inter-topic connections are higher for the latter topics. To limit the frequency of hierarchical connections and avoid repetitive ones, a threshold for similarity weight equal to 100 is used. In this respect, the frequency of vertical connections with $similarity > 100$ are 2166 versus 2675 while equivalent values for mean log-weights are 275.06 and 254.1 for those groups of users respectively. The results from t-test or show t-statistics of 11.98 with p-value of 0.00054 for rejection of null-hypothesis that differentiates those groups of users.

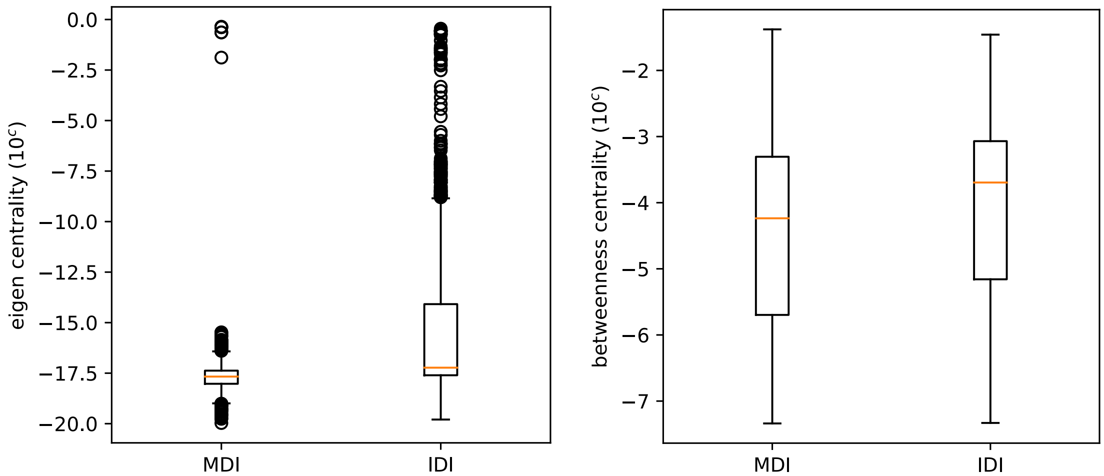
For temporal model of viral topics, t-test results also show that difference in temporal connections among viral topics between the two distinguished groups of users is statistically significant. However, the number of possible temporal connections is lower than hierarchical ones and distribution is closer to a normal distribution. Then the t-test is performed without using similarity weight threshold for selecting connections. The number of temporal connections among viral topics are 4309 versus 5380 and the equivalent values for mean of log-weights are 11.02 versus 9.77 respective to *MDI* and *IDI*. The results from t-test show F-statistics of 12.48 and p-value of 0.00041 for rejection of the the null hypothesis. Fig. 5.13 shows distribution of similarity-weights for hierarchical and temporal connections among *viral* (clustered) topics. The difference between *MDI* and *IDI* is not as much visible as statistically significant.

In addition to above tests, I performed a two-way ANOVA test to distinguish the effect of temporal time-frame on temporal model of topics. The independent variables in this test are labels (*MDI* and *IDI*) and time frames (1, 3, 7, and 21 days). Table 5.2 shows results from this analysis were each of the categorical variables are significantly related to connection weights but their interaction doesn't show similar effect.

Vars	type	sum-sq	df	f-statistic	p-value
frame-sizes	categorical	41.67	3	5.57	0.000809
user-labels	categorical	32.04	1	12.86	0.000337
labels:frames	categorical	4.39	3	0.58	0.623009
Residuals	-	24118	9681	Nan	Nan

Table 5.2: Statistical Analysis of Temporal Connections Among Topics

Another metric that can capture and distinguish a networked structural characteristics of viral topics is centrality. Fig. 5.14 shows two metrics of centrality in a network of viral topics. For eigen-centrality, regarded as a ranking measure, which assigns eigenvalues to nodes assuming that more important nodes are connected to other important nodes. This measures magnifies the effect of



(a) eigen-centrality

(b) betweenness-centrality

Fig. 5.14: Topic centrality for *MDI* and *IDI*: (a) eigen-centrality and (b) betweenness-centrality of viral topics. In both box-plots, only positive centralities are drawn when the number of distinct topics for *MDI* and *IDI* are (1190 vs 1801) and (995 vs 1163) and the number of clustered viral topics are 2496 and 2928 for the two user-groups respectively.

source topics with stronger similarity to other driven topics. The results of t-test shows the difference between the two groups of users is significant with t-statistic of 19.2 and p-value of 0. Betweenness centrality is another metric that captures importance of one topic in terms of being on the shortest paths that connect other topics. This metric might not suit perfectly to a network of viral topics but the results are consistent with eigen-centrality when viral topics achieve higher centrality values among *IDI*. The t-test results for the latter metric also show statistically significant difference among the two user-groups with t-statistic of 5.84 and p-value of $5.7e^{-09}$.

The results from analysis of centrality metrics might look contradictory to results from similarity weights as the *IDI* obtains consistently lower similarity weights but higher centrality measures but this observation is consistent with the hypothesis that more various topics are more discussed among interactive users while fewer topics are generated and circulated by *MDI* users. The comparison of eigen-centrality distributions between MDI and IDI shows that both distributions have heavy tail toward higher values which shows multiple topics with extremely higher eigen-centrality, i.e. influence on other topics.

5.7.2 Questions and Contributions

This chapter contributes: 1) a model of classifying accounts based on their activity on Twitter, 2) a clustering method for analyzing topics based on popularity and burstiness of connections among terms (term-pairs) and 3) a statistical model that finds the connection among the two models for interactive model in future.

For the detailed research questions in Sec. 5.2.1, this chapter introduced an influence-base analysis technique for evolution of content on social networks. For viral topics, a clustering technique was introduced to find viral topics using a network of term-pairs in the published documents based on frequency (popularity) and expected frequency (burstiness) of term-pairs. The results were able to capture real and viral news during the same time frame. The interactions are captured in terms of publishing and distributing documents across Twitter and their effect is categorized into active contribution (tweeting) and reactive contribution (retweeting, replying). A clustering technique was introduced to find viral topics based on a network of term-pairs in the published documents. In the analysis results, framework show that profiles of participants with different interactive behavior show different effect on developed topics and content in terms of time-frame granularity, connectedness, duration, type, and diversity of viral topics. The topics are validated by matching and tracking the archived news during the 210 days of data analysis. I further introduced a network model of topics using cosine similarity formula and defined hierarchal and temporal similarity between topics clustered on deferent granularities of time frame. The results distinguished the two behavioral camps by their effect on development of topics.

The results show that the exogenous behavior of users is more limited in terms of developing diverse topics while the endogenous behavior shows greater potential to develop new topics. In addition, interactive behavior among users is more focused on social, lifestyle and technology with more consistency among topics in terms of their temporal consecutiveness. On the other hand, media-driven topics are more focused on abstract, news-based and political topics on the lowest granularity. Nonetheless, these accounts show higher level of consistency in terms of hierarchal similarity among topics. The results from this research can be applied to design effective learning and broadcasting systems that combine diffusion of messages in social networks and interactive behavior of users. Future research shall develop comparative results of an Agent-based Social Systems (ABSS) simulation model based on the observations in this research to understand suspicious and adversarial behavior of influential accounts.

Chapter 6

Contributions and Discussions

The research questions in this thesis are: `(RQ1)` in Ch. 3 on “How to formulate a pricing and allocative mechanism that incentivizes self-centric components and improve the collective performance of a federated engineering systems?”, `(RQ2)` in Ch. 4 on “How to formulate auction-based algorithms to incentivize inter-federate exchange of resources and drive decentralized components toward better collective metrics such as higher value and lower computational cost?” and `(RQ3)` is the final research question explored in Ch. 5 on “How can exchange mechanisms for human resources and information contribute to better collective metrics in interactive and social networks?”.

In this thesis, I formulated an allocation mechanism to solve the centralized and combinatorial problem of scheduling tasks and routing data in a federated network of satellite systems. An achievement in Ch. 3 was modeling a MILP model based on multiple high level technical assumptions in communication and satellite systems, e.g. limited communication links, multiple federates, distributed resources, and periodic topologies, and low level financial assumptions, e.g. strategic bidding by federates, a bidding language by the auctioneer, utility function for federates with private information, and processing tasks assuming a monetary value and an inter-federate cost. The formulation for a pricing mechanism introduced in the same chapter considered estimation decentralized objective functions by federates by the auctioneer and devised incentive-compatibility in pricing by the auctioneer. In addition, the latter method addressed the issues with a stable operational equilibrium, e.g. Pareto-optimal results and Nash-equilibrium, using a value-maximizing mechanism combined with a federated cost-minimizing approach. The final results using simulation study in five different Pareto-optimal designs showed that the approach enhances values for all participants.

For the next step, in Ch. 4, I generalized the operational model for a general topology of federated task processing elements, i.e. TNE. The new operational model had minimal constraints on network topologies, opportunity cost of using resources by a federate, and number of task-processing elements. Then, the network size was extended while more permutations of a federated topology was considered for simulation study. However, the auction language (bidding) was extended toward two-sided bidding by participants, versus one-sided reverse-bidding in Ch. 3, including resource owners (link bids) and users (path bids). A path bid by a resource user implies an upper-bound for bids by resource owners in processing a task and delivering its data to destinations. Ch. 4 also formulates multiple sealed-bid auctions including first-price reverse-price, non-linear price-maximizing, and binary search for prices. The latter auctions include two auctions with closed-form solution for prices. The online auction with closed form solution (ONA) maximizes prices for each federate, biased toward the federates with lower bids, until final prices satisfy path constrains imposed by path bids. This auction emulates a combinatorial version of VCG scheme in the sense that as long as a solution is valid, the prices for any winner is maximized regardless of her bid. The last algorithm with virtual closed-form pricing, VPA, uses the same logic and maximizes prices on each path biased toward the lowest bids. Nonetheless, averaged prices of resources shared by a federate is proposed as resource prices while the individual prices for each resource is announced as prices to resources. Then, in VPA, prices for resource buyers and sellers are unbalanced while the payment among the federates is balanced, called *virtual pricing* in this thesis.

The final research question addresses an exploratory investigation of interactive models in real-world dynamic social networks. By an user model, a behavioral model of influential users based on the structural activities of users and by a collective model of a network, a model of emergent topics and discourse is intended. Two main classes of influential users are distinguished based on a statistical model of their activity on Twitter. The model is based on semi-supervised learning with initial labeling of two classes of users. Nonetheless, a clustering model of users with multiple output clusters confirmed that the two groups is the most significant distinction observed by the content model. For a model of topics and discourse, I proposed a clustering model of networked words and terms to detect viral topics in multiple granularities in time based on a networked model of terms and words used across millions of documents published by most influential users on Twitter. Multiple collective metrics are devised for statistical analysis of viral topics such as similarity metric, network centrality, and relevance score of topics. The statistical model of users and network model of topics show statistically significant connection between behavioral classes of users and network characteristics of content in a social network.

6.1 Comparative Discussion

Numerical results show the proposed mechanism for pricing resources is effective as it compensates for the adverse effects of strategic bidding on collective value and increases exchanging resources in a federation. Although the pricing mechanism is different from those in federated clouds in terms of operational model and objective function, our auctioneer achieves higher utility and prices for federates and shared resources like a cloud broker (CB) in [173] where an independent federate resembles a public provider (PP) in a multi-cloud system. Results also show heterogeneous prices for resources can increase values for participants in a federation, consistent with higher utility value and profit for bidders and resource providers using multiple auction mechanisms in cloud systems [174, 207]. In this thesis, resource providers both share and use resources and results were simulated from thousands of time steps with adaptive bidders versus single-run assessment and random bidding in the latter works. Finally, the independent and federated cases in the proposed mechanism is similar to the *non-federated* and *non-splitting* approaches investigated by Rebai in [39] when results also match *exact algorithm*'s in terms of value for providers and acceptance ratio of computational loads.

In applying auction-based algorithms, this thesis investigated collective value, dynamic allocation and pricing of resources, and behavioral convergence when results are consistent with and adds to those suggested by mechanisms in literature. The results in [154] give insight to using an incentivizing mechanism for distributed agents to declare their truthful resources where a performance metrics is response time to requests and the mechanism finds optimal payments to agents. In [155], a negotiation mechanism can increase the total utility and the speed of agreement in a market-based cloud system. In the latter work, authors use heuristic and intuitive baselines to compare results, assume known tasks and utility functions, and heuristic behavior by agents. An and Lesser in [156] investigate an NP-complete allocation mechanism for routing in a network and investigate the convergence ratio for a dynamic system with selfish and myopic participants. In a decentralized case, social welfare is improved by up to 90% of a centralized solution where number of participants, concurrency ratio, uncertainty, and number of routing paths have negative effects on decentralized social utility ratio. Han et al. in [208] show that we have a tradeoff between local performance and communication cost in task planning among distributed decision makers (DM) with shared asset, compared to a centralized solution. The results of an auction-based mechanism for resource allocation shows that iterative auction using VCG algorithm for allocating resources among multiple players, with 2D bidding on demands and unit prices, results in fast converging and monotonic increase of social welfare, decreasing bids and increasing demand for resources by selfish players [160]. In these works, the authors either don't consider the effect of topology for networked elements, avoid dynamic resources such as communication links among agents, or assume iterative and non-combinatorial resource allocation.

6.2 Future Research

New forms of networked systems such as clouds, satellite systems, autonomous vehicles, robotic missions, etc. call for more inclusive mechanism design for driving a collection of decentralized systems towards a collective goal. In this thesis, I approached modeling resource allocation in TNE and investigated auction-based algorithms for incentivizing a collective behavior by decentralized entities. Future research shall investigate experimental results from allocative and pricing mechanisms in combinatorial problems such as multi-source and multi-hop routing. The scalability of LP model shall be addressed for bigger networks with thousands of elements such as internet of things (IoT). Future models shall consider more sophisticated set of tasks and resource constraints, such as collaborative tasks requiring a set of computational resources. Future works on federated networks may investigate effects of adversarial behavior by an untruthful and strategic auctioneer, complexity of the operational solution in time for scaling purposes, and efficient algorithms for combinatorial pricing and resource allocation in a network.

For auction-based algorithms, high improvement is achievable through auction language, i.e. information exchange among elements, federates and the auctioneer. In addition, devising a decentralized auctioneer increases the algorithmic issues associated with collective metrics and sharing information while can significantly reduce the computational cost of the auctioneer.

For incentivizing mechanisms for information exchange in interactive social networks, the next steps include developing an Agent-based Social Systems (ABSS) model for simulating effect of micro-level behavior of agents on collective metrics in terms of developed topics. In the proposed model in Ch. 5, similar to temporal model introduced in [209], I use an empirical twitter-follower graph as a basis and consider three dimensions: 1) content (e.g. semantics), 2) social user activity e.g. retweets, replies and tweets, and 3) time. For the first one, a similarity measure among contents with minimum of 0 (for news at each time step), and 1 for retweets is defined. For the second one, an graph model of twitter users is used as a basis for observing the effect of social connections and influence. In this model, each user finds the subject for next activity from users that he follows. In other works, a social structure of user-follower affects the diffusion and propagation of contents. Third, temporal parameters are models using *news* as new pieces of information published by multiple sources in user network at each time step.

Three probabilities capture the relative weights of behavioral activities by a user:

- *Tweet probability* (p_t): a relatively independent activity by a user based on information input from the connected users or the pieces of news from an external platform system.

- *Retweet probability* (p_r): a purely dependent activity in terms of distribution of a content generated by other users. This activity leads to popularity of content among followers of the distributor.

- *Reply probability* (p_p): an engagement activity by a user that also leads to distribution (popularity) of a concept among the followers of distributor and the concept generator.

The above parameters define the characteristics of a user in terms of being *interactive* (IDI) versus *distributor* (MDI) in terms of weights of tweets, replies and retweets in their behavior. Interactive users engage more in distribution of contents and interaction with contents produced by other users (retweet and replies) than publishing their own content independently. Another difference among the three types of activities produced above is an assumed similarity of news pieces to existing ones. By a retweet, I assume the maximum similarity which leads to pure increase of popularity in the network. On the other hand, similarity of a tweet to existing ones is picked form a uniform distribution: $0 \leq s_{ij} \leq 1$. Then, a new tweet helps to change popularity of the concept according to the defined similarity. A reply also increases the popularity of the original tweet except that it doesn't create a new topic. The latter intuition is driven from Twitter where replies are separated from regular tweets and in general don't receive same level of attention and feedback from other users, although they help to popularize another content.

The clustering model introduced in Sec.5.4 will be applied on topics generated from the simulation study for different granularities of time [1]. The future works on interactive mechanisms in social networks shall address incentivizing mechanisms for driving collective behavior of users toward collective metrics. For instance, tutoring mechanisms, blockchain of news sources, and guiding bots can significantly incentivize spreading truthful versus fake content in interactive platforms such as Twitter.

[1] Visualizations, definitions and explanations regarding the simulation study will be covered in the future publications out of this chapter.

Appendix A

Storage Penalty

Assume an element has available memory M, probability p for availability of new task (next time step), and probability q for task delivery (next time step), and storage penalty for $M = 1$ equal to SP_1. To formulate the effect of additional storage ($M = 2$)[1], consider four probable states at next time step:

- **new task (**p**) - new delivery path (**q**)**: storage state doesn't change and ($M = 2$)
- **no task (**$1 - p$**)- no delivery path (**$1 - q$**)**: storage state still doesn't change and ($M = 2$)
- **new task (**p**)- no delivery path (**$1 - q$**)**: storage state changes to ($M = 1$)
- **no task (**$1 - p$**)- new delivery path (**q**)**: storage state changes to ($M = 3$)

For SP_1, I expect:

$$SP_2 = p(1-q)SP_1 + pqSP_2 + (1-p)(1-q)SP_2 + (1-p)qSP_3 \tag{A.1}$$

where the SP_3 is low value and close to zero ($SP_3 \approx 0$) as the opportunity cost of one memory

[1] In orbital federated satellite model, maximum storage unit on each element is equal to the size of two tasks $M = 2$.

unit when three are available $M = 3$ is assumed to be low, then:

$$SP_2 = \frac{p(1-q)}{p+q-2pq} SP_1$$

and assuming steady state model, p and q are close values $p \approx q$ and:

$$SP_2 \approx \frac{SP_1}{2}$$

In the orbital model, more available tasks exist than paths when the model is saturated. I expect $p \approx q$ because if tasks are available but delivery paths don't, this model doesn't process some tasks (without feasible path) and the presumed p decreases.

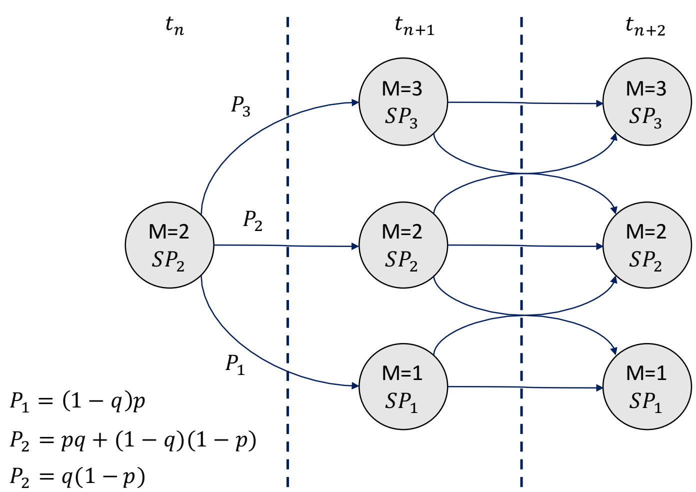


Fig. A.1: ***SP* states**: state transition of storage penalty given initial state of $M = 2$ at current time step t_n, which indicates available storage of 2 in case of delivering existing task on memory (i.e. storage is 1 without releasing memory). The state transition depends on probability of new task availability (p) and the probability of task delivery (q). The corresponding equation is shown in Eq. A.1.

Appendix B

Objective Functions

The operational model uses a temporal network elements in consecutive time steps to process tasks and schedule delivery. In this section, we introduce notations and MILP formulation for the operational models and objective functions.

B.1 Maximize Value

The value-maximizing objective function for a federation is defined as:

$$
\begin{aligned}
J_{value}(t) = &\sum_{T\in\mathbf{T}_a} \sum_{s\in\mathbf{s}_t} \begin{pmatrix} \mathbb{1}(e\in E_d)\times\mathcal{V}(T,t) \\ +\mathbb{1}(e\notin E_d)\times T.penalty \end{pmatrix} x_{resolve}(T,e,s) \\
&- \sum_{l_{ikt}\in L, T\in\mathbf{T}_a} \sum_{s\in\mathbf{s}_t} [T.size\times\epsilon] x_{trans}(T,l_{ikt},s) \\
&- \sum_{T\in\mathbf{T}_a} [T.size\times SP_{T.element}(t)] x_{store}(T,t) \qquad \text{(B.1)}
\end{aligned}
$$

where $SP_{T.element}(t)$ is the storage penalty for element owner of task T at time step t which was defined by Eq.3.4. The intuition is that resolving a task affects the *federation value* through value function of delivering it, or the penalty function of failure to deliver the task. In addition, data transmission through a link affects the federation value with the network communication cost or the opportunity cost of storage penalty ($i = k$) defined by *cost function* ζ in Eq.3.4. The MILP model of an operational run at time t subject to capacity and financial constraints is defined as:

$$\text{find: } x_{process}(T,t), x_{trans}(T,l_{ikt},s),$$
$$x_{resolve}(T,e,s), x_{store}(T,t), x_{read}(T,t)$$
$$T \in \mathbf{T}_a, l_{ikt} \in L, s \in \mathbf{s}_t, t \in \mathbf{t}, e \in E$$

$$\text{maximize: } J_{value}(t) \tag{B.2}$$

subject to:

$$\sum_{T \in \mathbf{T}, s \in \mathbf{s}_t} x_{trans}(T,l,s) \le capacity(l), \forall l \in L \tag{B.3}$$

$$\sum_{T \in \mathbf{T}, t \in \mathbf{t}} T.size \begin{pmatrix} x_{process}(T,t)- \\ \sum_{e \in E} x_{resolve}(T,e,t) \end{pmatrix} \le capacity(e) \tag{B.4}$$

$$\sum_{l \in inlink(e,t)} x_{trans}(T,l,s) - \sum_{l \in outlink(e)} x_{trans}(T,l,s+1)$$
$$- x_{resolve}(T,e,s) = 0, \forall T \in \mathbf{T}, s \in \mathbf{s}_t : e \ne T.element \tag{B.5}$$

$$x_{process}(T,t) + x_{read}(T,t) - \sum_{s \in \mathbf{s}_t} x_{resolve}(T,e,s) - x_{store}(T,t)$$
$$- \sum_{l \in outlink(e,t), s \in \mathbf{s}_t} x_{trans}(T,l,s) = 0, \forall T \in \mathbf{T} : e = T.element \tag{B.6}$$

$$\sum_{s \in \mathbf{s}_t} x_{resolve}(T,e,s) = 1, \text{ if } T.expiration <= t \tag{B.7}$$

where the $inlink$ and $outlink$ are the set of links into and out of an element:

$$inlink(e_k, t) = \{l_{jkt} \in L : e_j \in E\}$$

$$outlink(e_i, t) = \{l_{idt} \in L : e_d \in E\}$$

Constraint B.3 defines the limits on link capacity, Cons. B.4 defines the storage capacity of an element, Cons. B.5 balances the inflow and outflow of data into and out of an element except for the sources, Cons. B.6 is the net flow constraint for source elements, lastly, Cons. B.7 resolves expired task to free up memory of expired data.

B.2 Minimize Cost

$$\text{find: } x_{trans}(T, l_{ikt}, s), x_{resolve}(T, e, s)$$

$$\text{given: } x^{@}_{read}(T, t), x^{@}_{store}(T, t), x^{@}_{process}(T, e, s)$$

$$T \in \mathbf{T}, l_{ikt} \in L, s \in \mathbf{s}_t, t \in \mathbf{t}, e \in E$$

$$\text{minimize: } \sum_{l_{ikt} \in L} \sum_{s \in \mathbf{s}_t} \zeta(\mathcal{F}_t(T), l_{ikt}) x_{trans}(T, l_{ikt}, s) + \sum_{T \in \mathbf{T}_a} \sum_{s \in \mathbf{s}_t} \begin{pmatrix} \mathbb{1}(e \in E_d)\mathcal{V}(T, t) \\ +\mathbb{1}(e \notin E_d) T.penalty \end{pmatrix} x_{resolve}(T, e, s) \tag{B.8}$$

subject to:

$$Constraints\ B.3, B.5, B.6, B.7$$

$$\sum_{s \in \mathbf{s}_t, e \in E} x_{resolve}(T, e, s) = \sum_{s \in \mathbf{s}_t, e \in E} x^{@}_{resolve}(T, e, s) \tag{B.9}$$

where $x^{@}_{store}(T, t)$, $x^{@}_{read}(T, t)$ and $x^{@}_{process}(T, t)$ are the calculated decisions from Eq.3.14. Cons. B.9 ensures that the tasks resolved at time step t using Eq.3.14 would also be resolved in above solution, although, tasks may be delivered to different elements.

Appendix C

Q-Learning

For adaptive bidding model, we apply a generic open-source q-learning module[1]. Nonetheless, we need to resolve three compatibility issues between bidding behavior and the basic q-learning: 1. temporal distance between actions and reward (task pick up and tasks delivery) 2. interdependency between actions and rewards in consecutive times steps, 3. continuous action space in bidding ($c_f \in \mathbb{R}$). The first and second concerns are addressed in updating multiple Q-values given a reward value. Regarding the state-action dimensionality, the larger the action space gets, the smaller will be the probability of visiting the same state again [210]. Then, we define Gaussian distance between states to update the Q-values. Assuming a state action pair as $x_i = (s_i, a_i)$, the learning parameter is:

$$\alpha_{ij} = \frac{\alpha}{K_i} e^{\frac{\Delta s_{ij}^2}{2\sigma_s^2} + \frac{\Delta a_{ij}^2}{2\sigma_a^2}}$$

where α is the learning factor from q-learning (Eq.4.13), $\Delta s_{ij} = |s_j - s_i|$ and $\Delta a_{ij} = |a_j - a_i|$ and K_i is the normalizing factor that ensures the sum of all Q values are updated with α.

[1] https://gist.github.com/kastnerkyle/d127197dcfdd8fb888c2

After receiving each reward $R^{t'}$, all Q-values will be updated according to their α_{ij}:

$$Q(x_{ij}^t) \leftarrow Q(x_{ij}^t) + \alpha_{ij}[\frac{R_{x^t}}{N} + \gamma Q(x_{ij}^{t+1}) - Q(x_{ij}^t)]$$

$$\forall Q(x_{ij}^t) : t > t' - \Delta t \quad \forall j \in S$$

where Δt represents the number of time steps after which the actions are uncoupled from rewards (i.e. actions cannot affect further rewards), N is the total number of actions during Δt.

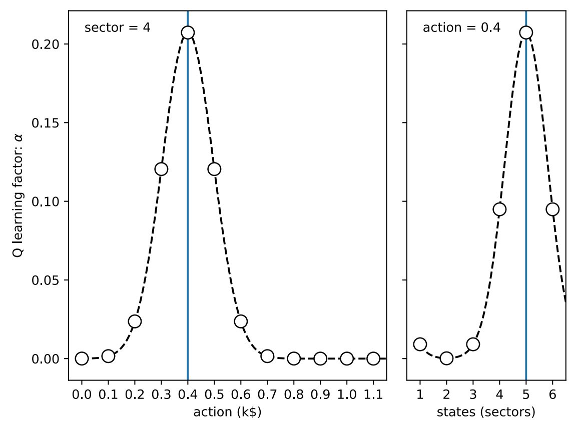


Fig. C.1: Q-value update parameter (α) with two dimensional gaussian update distribution for a sample point of $state(sector) = 4$ and $action(cost) = 0.4$.

Bibliography

[1] Brendan Jennings and Rolf Stadler. Resource management in clouds: Survey and research challenges. *Journal of Network and Systems Management*, 23(3):567–619, 2015.

[2] Chiara Farronato and Andrey Fradkin. The welfare effects of peer entry in the accommodation market: The case of airbnb. Technical report, National Bureau of Economic Research, 2018.

[3] The sharing economy: Understanding the opportunities for growth, JUNE 2017. [Online; posted June-2017].

[4] The 10 billion club: Meet the most valuable startups in the us, Dec 2017. [Online; posted 09-Dec-2017].

[5] Liran Einav, Chiara Farronato, and Jonathan Levin. Peer-to-peer markets. *Annual Review of Economics*, 8:615–635, 2016.

[6] The future of personal satellite, 2016. [Online; posted May-2016].

[7] Here's why you should know what the 'sharing economy' is, JUNE 2018. [Online; posted June-2018].

[8] Arvind Malhotra and Marshall Van Alstyne. The dark side of the sharing economy and how to lighten it. *Communications of the ACM*, 57(11):24–27, 2014.

[9] Joseph E Stiglitz. 8. inequality and economic growth. *The Political Quarterly*, 86(S1):134–155, 2015.

[10] Sarah Underwood. Blockchain beyond bitcoin. *Communications of the ACM*, 59(11):15–17, 2016.

[11] Blockchain market: Forecast (2017 - 2023), AUGUST 2017. [Online; posted Dec-2017].

[12] Blockchain market cap, DECEMBER 2017. [Online; posted 01-Dec-2017].

[13] Mark W Maier. Architecting principles for systems-of-systems. *Systems Engineering: The Journal of the International Council on Systems Engineering*, 1(4):267–284, 1998.

[14] Alessandro Golkar and Ignasi Lluch i Cruz. The federated satellite systems paradigm: Concept and business case evaluation. *Acta Astronautica*, 111:230–248, 2015.

[15] Daniel Selva, Alessandro Golkar, Olga Korobova, Ignasi Lluch i Cruz, Paul Collopy, and Olivier L de Weck. Distributed earth satellite systems: What is needed to move forward? *Journal of Aerospace Information Systems*, 14(8):412–438, 2017.

[16] Kimon P Valavanis and George J Vachtsevanos. Future of unmanned aviation. In *Handbook of unmanned aerial vehicles*, pages 2993–3009. Springer, 2015.

[17] José Luís Azevedo, Bernardo Cunha, and Luís Almeida. Hierarchical distributed architectures for autonomous mobile robots: a case study. In *Emerging Technologies and Factory Automation, 2007. ETFA. IEEE Conference on*, pages 973–980. IEEE, 2007.

[18] Ioan Petri, Tom Beach, Mengsong Zou, Javier Diaz Montes, Omer Rana, and Manish Parashar. Exploring models and mechanisms for exchanging resources in a federated cloud. In *Cloud Engineering (IC2E), 2014 IEEE International Conference on*, pages 215–224. IEEE, 2014.

[19] Yin Chen, Xinjun Mao, Fu Hou, Qiuzen Wang, and Shuo Yang. Combining re-allocating and re-scheduling for dynamic multi-robot task allocation. In *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*, pages 000395–000400. IEEE, 2016.

[20] Ignasi Lluch I Cruz. *A FRAMEWORK FOR ARCHITECTING FEDERATIONS OF SYSTEMS*. PhD dissertation, Skolkovo Institute of Science and Technology, 2017.

[21] Hripsime Matevosyan, Christianna Taylor, and Alessandro Golkar. Evaluating virtual satellite mission opportunities. In *AIAA SPACE 2015 Conference and Exposition*, page 4674, 2015.

[22] Teemu Koponen, Martin Casado, Pankaj Thakkar, Ronghua Zhang, and Daniel J Wendlandt. Packet processing in federated network, February 24 2015. US Patent 8,964,767.

[23] Nan Zheng, Shuangyong Song, and Hongyun Bao. A temporal-topic model for friend recommendations in chinese microblogging systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(9):1245–1253, 2015.

[24] Andreas M Kaplan and Michael Haenlein. The early bird catches the news: Nine things you should know about micro-blogging. *Business horizons*, 54(2):105–113, 2011.

[25] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM, 2007.

[26] Avery E Holton, Mark Coddington, Seth C Lewis, and Homero Gil De Zuniga. Reciprocity and the news: The role of personal and social media reciprocity in news creation and consumption. *International Journal of Communication*, 9:22, 2015.

[27] Jim Edwards. The inventor of the twitter hashtag explains why he didn't patent it, 2013. Available: http://www.businessinsider.com/chris-messina-talks-about-inventing-the-hashtag-on-twitter-2013-11, accessed 2013-11-21.

[28] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 2016.

[29] David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, et al. The science of fake news. *Science*, 359(6380):1094–1096, 2018.

[30] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018.

[31] D Gale Johnson. Resource allocation under share contracts. *Journal of Political economy*, 58(2):111–123, 1950.

[32] Brent Nee Chun and David E Culler. *Market-based proportional resource sharing for clusters*. Computer Science Division, University of California Berkeley, 2000.

[33] Walter Powell. Neither market nor hierarchy. *The sociology of organizations: classic, contemporary, and critical readings*, 315:104–117, 2003.

[34] Juho Hamari, Mimmi Sjöklint, and Antti Ukkonen. The sharing economy: Why people participate in collaborative consumption. *Journal of the Association for Information Science and Technology*, 67(9):2047–2059, 2016.

[35] Panayotis Antoniadis, Serge Fdida, Timur Friedman, and Vishal Misra. Federation of virtualized infrastructures: sharing the value of diversity. In *Proceedings of the 6th International COnference*, page 12. ACM, 2010.

[36] Christina Delimitrou. *Improving resource efficiency in cloud computing*. PhD thesis, Stanfort University, 2015.

[37] Zhen Xiao, Weijia Song, Qi Chen, et al. Dynamic resource allocation using virtual machines for cloud computing environment. *IEEE Trans. Parallel Distrib. Syst.*, 24(6):1107–1117, 2013.

[38] Young Choon Lee and Albert Y Zomaya. Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing*, 60(2):268–280, 2012.

[39] Salma Rebai. *Resource allocation in Cloud federation*. PhD thesis, Institut National des Télécommunications, 2017.

[40] Adam S Hayes. Cryptocurrency value formation: An empirical study leading to a cost of production model for valuing bitcoin. *Telematics and Informatics*, 34(7):1308–1321, 2017.

[41] Kevin Delmolino, Mitchell Arnett, Ahmed Kosba, Andrew Miller, and Elaine Shi. Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab. In *International Conference on Financial Cryptography and Data Security*, pages 79–94. Springer, 2016.

[42] Eleftherios Kokoris Kogias, Philipp Jovanovic, Nicolas Gailly, Ismail Khoffi, Linus Gasser, and Bryan Ford. Enhancing bitcoin security and performance with strong consistency via collective signing. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 279–296. USENIX Association, 2016.

[43] Lin Xiao, Mikael Johansson, and Stephen P Boyd. Simultaneous routing and resource allocation via dual decomposition. *IEEE Transactions on Communications*, 52(7):1136–1144, 2004.

[44] Abbas Ehsanfar and Paul T. Grogan. Mechanism design for sharing resources in federated networks. *Journal of Networks and Systems Management*, forthcoming.

[45] Linquan Zhang, Zongpeng Li, and Chuan Wu. Dynamic resource provisioning in cloud computing: A randomized auction approach. In *INFOCOM, 2014 Proceedings IEEE*, pages 433–441. IEEE, 2014.

[46] Juncheng Jia, Qian Zhang, Qin Zhang, and Mingyan Liu. Revenue generation for truthful spectrum auction in dynamic spectrum access. In *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 3–12. ACM, 2009.

[47] Vipul Jain and Ignacio E Grossmann. Algorithms for hybrid milp/cp models for a class of optimization problems. *INFORMS Journal on computing*, 13(4):258–276, 2001.

[48] Jeremy Geelan et al. Twenty one experts define cloud computing. *Cloud Computing Journal*, 4:1–5, 2009.

[49] Galen Gruman and Eric Knorr. What cloud computing really means. *InfoWorld*, 37:13, 2008.

[50] Luis M Vaquero, Luis Rodero-Merino, Juan Caceres, and Maik Lindner. A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1):50–55, 2008.

[51] Jeroen Famaey, Steven Latré, Tim Wauters, and Filip De Turck. End-to-end resource management for federated delivery of multimedia services. *Journal of Network and Systems Management*, 22(3):396–433, 2014.

[52] Brendan Jennings, Kevin Feeney, and Joel J Fleck. Managing federations and cooperative management. *Journal of Network and Systems Management*, 22(3):297–301, 2014.

[53] Abbas Ehsanfar and Babak Heydari. An incentive-compatible scheme for electricity cooperatives: An axiomatic approach. *IEEE Transactions on Smart Grid*, 2016.

[54] Paul T Grogan, Koki Ho, Alessandro Golkar, and Olivier L de Weck. Multi-actor value modeling for federated systems. *IEEE Systems Journal*, 2016.

[55] Paul T Grogan and Olivier L de Weck. Interactive simulation games to assess federated satellite system concepts. In *Aerospace Conference, 2015 IEEE*, pages 1–13. IEEE, 2015.

[56] Chunhung Richard Lin and Jain-Shing Liu. Qos routing in ad hoc wireless networks. *IEEE Journal on selected areas in communications*, 17(8):1426–1438, 1999.

[57] Onn Shehory and Sarit Kraus. Methods for task allocation via agent coalition formation. *Artificial intelligence*, 101(1-2):165–200, 1998.

[58] Archie C Chapman, Rosa Anna Micillo, Ramachandra Kota, and Nicholas R Jennings. Decentralised dynamic task allocation: a practical game: theoretic approach. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 915–922. International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[59] Guofang Nan, Zhifei Mao, Mei Yu, Minqiang Li, Honggang Wang, and Yan Zhang. Stackelberg game for bandwidth allocation in cloud-based wireless live-streaming social networks. *IEEE Systems Journal*, 8(1):256–267, 2014.

[60] Yannis A Korilis, Aurel A Lazar, and Ariel Orda. Achieving network optima using stackelberg routing strategies. *IEEE/ACM Transactions on Networking (TON)*, 5(1):161–173, 1997.

[61] Nguyen Cong Nguyen, Ping Wang, Dusit Niyato, Yonggang Wen, and Zhu Han. Resource management in cloud networking using economic analysis and pricing models: a survey. *IEEE Communications Surveys & Tutorials*, 2017.

[62] Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Eva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008.

[63] Liping Fu, D Sun, and Laurence R Rilett. Heuristic shortest path algorithms for transportation applications: state of the art. *Computers & Operations Research*, 33(11):3324–3343, 2006.

[64] Zhifang Yang, Haiwang Zhong, Qing Xia, Chongqing Kang, Tianen Chen, and Yan Li. A structural transmission cost allocation scheme based on capacity usage identification. *IEEE Transactions on Power Systems*, 31(4):2876–2884, 2016.

[65] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

[66] Annick Laruelle and Federico Valenciano. Shapley-shubik and banzhaf indices revisited. *Mathematics of operations research*, 26(1):89–104, 2001.

[67] Lena Mashayekhy, Mahyar Movahed Nejad, and Daniel Grosu. Cloud federations in the sky: Formation game and mechanism. *IEEE Transactions on Cloud Computing*, 3(1):14–27, 2015.

[68] Ramasuri Narayanam and Yadati Narahari. A shapley value-based approach to discover influential nodes in social networks. *IEEE Transactions on Automation Science and Engineering*, 8(1):130–147, 2011.

[69] Matthew O Jackson and Alison Watts. The evolution of social and economic networks. *Journal of Economic Theory*, 106(2):265–295, 2002.

[70] Lone Grønbæk Kronbak and Marko Lindroos. Sharing rules and stability in coalition games with externalities. *Marine Resource Economics*, 22(2):137–154, 2007.

[71] Yang Zhang, Chonho Lee, Dusit Niyato, and Ping Wang. Auction approaches for resource allocation in wireless systems: A survey. *IEEE Communications surveys & tutorials*, 15(3):1020–1041, 2013.

[72] Aleksandar Pekeč and Michael H Rothkopf. Combinatorial auction design. *Management Science*, 49(11):1485–1503, 2003.

[73] John Morgan. Efficiency in auctions: theory and practice updated copies of this paper can be found at www. wws. princeton. edu/˜ rjmorgan. *Journal of International Money and Finance*, 20(6):809–838, 2001.

[74] Jun Sun, Eytan Modiano, and Lizhong Zheng. Wireless channel allocation using an auction algorithm. *IEEE Journal on Selected Areas in Communications*, 24(5):1085–1096, 2006.

[75] Hoon Kim, Keunyoung Kim, Youngnam Han, and Sangboh Yun. A proportional fair scheduling for multicarrier transmission systems. In *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, volume 1, pages 409–413. IEEE, 2004.

[76] Olivier Marce, Hoang-Hai Tran, and Bruno Tuffin. Double-sided auctions applied to vertical handover for mobility management in wireless networks. *Journal of network and systems management*, 22(4):658–681, 2014.

[77] Erhan Kutanoglu and S David Wu. On combinatorial auction and lagrangean relaxation for distributed resource scheduling. *IIE transactions*, 31(9):813–826, 1999.

[78] Kevin Leyton-Brown, Mark Pearson, and Yoav Shoham. Towards a universal test suite for combinatorial auction algorithms. In *Proceedings of the 2nd ACM conference on Electronic commerce*, pages 66–76. ACM, 2000.

[79] Chen Xu, Lingyang Song, Zhu Han, Qun Zhao, Xiaoli Wang, Xiang Cheng, and Bingli Jiao. Efficiency resource allocation for device-to-device underlay communication systems: A reverse iterative combinatorial auction based approach. *IEEE Journal on Selected Areas in Communications*, 31(9):348–358, 2013.

[80] Chu-Fen Li. Cloud computing system management under flat rate pricing. *Journal of network and systems management*, 19(3):305–318, 2011.

[81] Marian Mihailescu and Yong Meng Teo. Dynamic resource pricing on federated clouds. In *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, pages 513–517. IEEE Computer Society, 2010.

[82] Chen Xu, Lingyang Song, Zhu Han, Qun Zhao, Xiaoli Wang, and Bingli Jiao. Interference-aware resource allocation for device-to-device communications as an underlay using sequential second price auction. In *Communications (ICC), 2012 IEEE International Conference on*, pages 445–449. IEEE, 2012.

[83] Aurel A Lazar, Nemo Semret, et al. Design, analysis and simulation of the progressive second price auction for network bandwidth sharing. *Columbia University (April 1998)*, 1998.

[84] Hong Zhang, Hongbo Jiang, Bo Li, Fangming Liu, Athanasios V Vasilakos, and Jiangchuan Liu. A framework for truthful online auctions in cloud computing with heterogeneous user demands. *IEEE Transactions on Computers*, 65(3):805–818, 2016.

[85] Udrivolf Pica and Alessandro Golkar. Sealed-bid reverse auction pricing mechanisms for federated satellite systems. *Systems Engineering*, 20(5):432–446, 2017.

[86] Marc Berhault, He Huang, Pinar Keskinocak, Sven Koenig, Wedad Elmaghraby, Paul Griffin, and Anton Kleywegt. Robot exploration with combinatorial auctions. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 2, pages 1957–1962. IEEE, 2003.

[87] Yufeng Wang, Wei Dai, Qun Jin, and Jianhua Ma. Bcinet: A biased contest-based crowdsourcing incentive mechanism through exploiting social networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2018.

[88] Feng Li, Lin Zhang, Yongkui Liu, and Yuanjun Laili. Qos-aware service composition in cloud manufacturing: A gale-shapley algorithm-based approach. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, (99):1–12, 2018.

[89] Sven De Vries and Rakesh V Vohra. Combinatorial auctions: A survey. *INFORMS Journal on computing*, 15(3):284–309, 2003.

[90] Lawrence M Ausubel, Paul Milgrom, et al. The lovely but lonely vickrey auction. *Combinatorial auctions*, 17:22–26, 2006.

[91] Qianyi Huang, Yixin Tao, and Fan Wu. Spring: A strategy-proof and privacy preserving spectrum auction mechanism. In *INFOCOM, 2013 Proceedings IEEE*, pages 827–835. IEEE, 2013.

[92] Miao Pan, Jinyuan Sun, and Yuguang Fang. Purging the back-room dealing: Secure spectrum auction leveraging paillier cryptosystem. *IEEE Journal on Selected Areas in Communications*, 29(4):866–876, 2011.

[93] Robert P Gilles. *The cooperative game theory of networks and hierarchies*, volume 44. Springer Science & Business Media, 2010.

[94] Peter Mell, Tim Grance, et al. The nist definition of cloud computing. 2011.

[95] Candace Jones, William S Hesterly, and Stephen P Borgatti. A general theory of network governance: Exchange conditions and social mechanisms. *Academy of management review*, 22(4):911–945, 1997.

[96] Christopher P Bergh, Max E Metral, David Henry Ritter, Jonathan Ari Sheena, and James J Sullivan. Distributed system for facilitating exchange of user information and opinion using automated collaborative filtering, August 29 2000. US Patent 6,112,186.

[97] Daniel J Brass, Kenneth D Butterfield, and Bruce C Skaggs. Relationships and unethical behavior: A social network perspective. *Academy of management review*, 23(1):14–31, 1998.

[98] Zhu Wang, Daqing Zhang, Xingshe Zhou, Dingqi Yang, Zhiyong Yu, and Zhiwen Yu. Discovering and profiling overlapping communities in location-based social networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(4):499–509, 2014.

[99] Muhammad Ashad Kabir, Jun Han, and Alan Colman. Modeling and coordinating social interactions in pervasive environments. In *Engineering of Complex Computer Systems (ICECCS), 2011 16th IEEE International Conference on*, pages 243–252. IEEE, 2011.

[100] Yohsuke Murase, János Török, Hang-Hyun Jo, Kimmo Kaski, and János Kertész. Multilayer weighted social network model. *Physical Review E*, 90(5):052810, 2014.

[101] Xiaoling Sun, Hongfei Lin, and Kan Xu. A social network model driven by events and interests. *Expert Systems With Applications*, 42(9):4229–4238, 2015.

[102] Riitta Toivonen, Lauri Kovanen, Mikko Kivelä, Jukka-Pekka Onnela, Jari Saramäki, and Kimmo Kaski. A comparative study of social network models: Network evolution models and nodal attribute models. *Social Networks*, 31(4):240–254, 2009.

[103] Yiman Sun. A dynamic model of censorship. 2018.

[104] Lillian Weng, Alessandro Flammini, Alessandro Vespignani, and Fillipo Menczer. Competition among memes in a world with limited attention. *Scientific reports*, 2:335, 2012.

[105] Yaron Singer. How to win friends and influence people, truthfully: influence maximization mechanisms for social networks. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 733–742. ACM, 2012.

[106] Junjie Wu, Haoyan Sun, and Yong Tan. Social media research: A review. *Journal of Systems Science and Systems Engineering*, 22(3):257–282, 2013.

[107] Robert West, Hristo S Paskov, Jure Leskovec, and Christopher Potts. Exploiting social network structure for person-to-person sentiment analysis. *arXiv preprint arXiv:1409.2450*, 2014.

[108] Huiqi Zhang, Ram Dantu, and João W Cangussu. Socioscope: Human relationship and behavior analysis in social networks. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(6):1122–1143, 2011.

[109] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. A bayesian approach toward finding communities and their evolutions in dynamic social networks. In *Proceedings of the 2009 SIAM International Conference on Data Mining*, pages 990–1001. SIAM, 2009.

[110] Santa Agreste, Pasquale De Meo, Emilio Ferrara, Sebastiano Piccolo, and Alessandro Provetti. Analysis of a heterogeneous social network of humans and cultural objects. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(4):559–570, 2015.

[111] Ahmed F AlEroud and George Karabatis. Queryable semantics to detect cyber-attacks: A flow-based detection approach. *IEEE transactions on systems, man, and cybernetics: systems*, 2016.

[112] Sara Keretna, Ahmad Hossny, and Doug Creighton. Recognising user identity in twitter social networks via text mining. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 3079–3082. IEEE, 2013.

[113] John P Dickerson, Vadim Kagan, and VS Subrahmanian. Using sentiment to detect bots on twitter: Are humans more opinionated than bots? In *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*, pages 620–627. IEEE, 2014.

[114] Zafar Gilani, Reza Farahbakhsh, Gareth Tyson, Liang Wang, and Jon Crowcroft. Of bots and humans (on twitter). In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, pages 349–354. ACM, 2017.

[115] Norman Fairclough. Critical discourse analysis as a method in social scientific research. *Methods of critical discourse analysis*, 5:121–138, 2001.

[116] Andrew McCallum, Andres Corrada-Emmanuel, and Xuerui Wang. Topic and role discovery in social networks. 2005.

[117] Chao Gao and Jiming Liu. Network-based modeling for characterizing human collective behaviors during extreme events. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(1):171–183, 2017.

[118] Xiangju Qin, Pádraig Cunningham, and Michael Salter-Townshend. The influence of network structures of wikipedia discussion pages on the efficiency of wikiprojects. *Social Networks*, 43:1–15, 2015.

[119] Manh Cuong Pham and Ralf Klamma. The structure of the computer science knowledge network. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*, pages 17–24. IEEE, 2010.

[120] Andrea Landherr, Bettina Friedl, and Julia Heidemann. A critical review of centrality measures in social networks. *Business & Information Systems Engineering*, 2(6):371–385, 2010.

[121] Alexander G Nikolaev, Raihan Razib, and Ashwin Kucheriya. On efficient use of entropy centrality for social network analysis and community detection. *Social Networks*, 40:154–162, 2015.

[122] Kundan Kandhway and Joy Kuri. Using node centrality and optimal control to maximize information diffusion in social networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(7):1099–1110, 2017.

[123] Sitaram Asur and Bernardo A Huberman. Predicting the future with social media. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 492–499. IEEE Computer Society, 2010.

[124] Andrea Ceron, Luigi Curini, and Stefano M Iacus. Using sentiment analysis to monitor electoral campaigns: Method mattersevidence from the united states and italy. *Social Science Computer Review*, 33(1):3–20, 2015.

[125] Mehdi Alighanbari and Jonathan P How. Decentralized task assignment for unmanned aerial vehicles. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 5668–5673. IEEE, 2005.

[126] Ernesto Nunes and Maria L Gini. Multi-robot auctions for allocation of tasks with temporal constraints. In *AAAI*, pages 2110–2116, 2015.

[127] Paul T Grogan, Koki Ho, Alessandro Golkar, and Olivier L de Weck. Bounding the value of collaboration in federated systems. In *Systems Conference (SysCon), 2016 Annual IEEE*, pages 1–7. IEEE, 2016.

[128] Paul D Collopy and Peter M Hollingsworth. Value-driven design. *Journal of Aircraft*, 48(3):749, 2011.

[129] Sean Rooney, Daniel Bauer, and Rudy Deydier. A federated peer-to-peer network game architecture. *IEEE Communications Magazine*, 42(5):114–122, 2004.

[130] Makhlouf Hadji and Djamal Zeghlache. Mathematical programming approach for revenue maximization in cloud federations. *IEEE Transactions on Cloud Computing*, 5(1):99–111, 2017.

[131] Elizabeth M Royer and Charles E Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 207–218. ACM, 1999.

[132] Sung-Ju Lee, Mario Gerla, and Ching-Chuan Chiang. On-demand multicast routing protocol. In *Wireless Communications and Networking Conference, 1999. WCNC. 1999 IEEE*, volume 3, pages 1298–1302. IEEE, 1999.

[133] Nikolaos A Pantazis, Stefanos A Nikolidakis, and Dimitrios D Vergados. Energy-efficient routing protocols in wireless sensor networks: A survey. *IEEE Communications surveys & tutorials*, 15(2):551–591, 2013.

[134] Luiz Carlos P Albini, Antonio Caruso, Stefano Chessa, and Piero Maestrini. Reliable routing in wireless ad hoc networks: The virtual routing protocol. *Journal of Network and Systems Management*, 14(3):335–358, 2006.

[135] Ismail Chabini. Discrete dynamic shortest path problems in transportation applications: Complexity and algorithms with optimal run time. *Transportation Research Record: Journal of the Transportation Research Board*, (1645):170–175, 1998.

[136] Matthew C Gombolay, Ronald Wilcox, and Julie A Shah. Fast scheduling of multi-robot teams with temporospatial constraints. In *Robotics: Science and Systems*, 2013.

[137] Edward C Rosenthal. Shortest path games. *European Journal of Operational Research*, 224(1):132–140, 2013.

[138] Edward C Rosenthal. A cooperative game approach to cost allocation in a rapid-transit network. *Transportation Research Part B: Methodological*, 97:64–77, 2017.

[139] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[140] Paul T Grogan and Olivier L de Weck. The isos modeling framework for infrastructure systems simulation. *IEEE Systems Journal*, 9(4):1139–1150, 2015.

[141] Ruben E Perez, Peter W Jansen, and Joaquim RRA Martins. pyopt: a python-based object-oriented framework for nonlinear constrained optimization. *Structural and Multidisciplinary Optimization*, 45(1):101–118, 2012.

[142] William Tolone, Gail-Joon Ahn, Tanusree Pai, and Seng-Phil Hong. Access control in collaborative systems. *ACM Computing Surveys (CSUR)*, 37(1):29–41, 2005.

[143] Mohammad Hossein Manshaei, Quanyan Zhu, Tansu Alpcan, Tamer Bacşar, and Jean-Pierre Hubaux. Game theory meets network security and privacy. *ACM Computing Surveys (CSUR)*, 45(3):25, 2013.

[144] Shene Johnson. *The worlds of the federations*. Paramount Pictures Corporation, New York, USA, 1989.

[145] Paul Joseph and Sharon Carton. The law of the federation: Images of law, lawyers, and the legal system in star trek, the next generation. *U. Tol. L. Rev.*, 24:43, 1992.

[146] Geeks' guide to galaxy. The economic lessons of star treks money-free society, May 2016. [Online; posted 25-May-2016].

[147] Rick Webb. The economics of star trek, Nov 2013. [Online; posted 06-Nov-2013].

[148] Aaron Archer, Christos Papadimitriou, Kunal Talwar, and Éva Tardos. An approximate truthful mechanism for combinatorial auctions with single parameter agents. *Internet Mathematics*, 1(2):129–150, 2004.

[149] Sharrukh Zaman and Daniel Grosu. Combinatorial auction-based allocation of virtual machine instances in clouds. *Journal of Parallel and Distributed Computing*, 73(4):495–508, 2013.

[150] Avrim Blum and Jason D Hartline. Near-optimal online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1156–1163. Society for Industrial and Applied Mathematics, 2005.

[151] Kun Zhang, Rui Wang, and Depei Qian. Aim: An auction incentive mechanism in wireless networks with opportunistic routing. In *Computational Science and Engineering (CSE), 2010 IEEE 13th International Conference on*, pages 28–33. IEEE, 2010.

[152] Anirban Mondal, Sanjay Kumar Madria, and Masaru Kitsuregawa. Abide: A bid-based economic incentive model for enticing non-cooperative peers in mobile-p2p networks. In *International Conference on Database Systems for Advanced Applications*, pages 703–714. Springer, 2007.

[153] Luzi Anderegg and Stephan Eidenbenz. Ad hoc-vcg: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 245–259. ACM, 2003.

[154] Daniel Grosu and Anthony T Chronopoulos. Algorithmic mechanism design for load balancing in distributed systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(1):77–84, 2004.

[155] Seokho Son and Kwang Mong Sim. A price-and-time-slot-negotiation mechanism for cloud service reservations. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(3):713–728, 2012.

[156] Bo An and Victor Lesser. Characterizing contract-based multiagent resource allocation in networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(3):575–586, 2010.

[157] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37, 1961.

[158] Theodore Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.

[159] Edward H Clarke. Multipart pricing of public goods. *Public choice*, 11(1):17–33, 1971.

[160] Suli Zou, Zhongjing Ma, and Xiangdong Liu. Auction-based mechanism for dynamic and efficient resource allocation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(1):34–49, 2018.

[161] Daniel Lehmann, Liadan Ita Oćallaghan, and Yoav Shoham. Truth revelation in approximately efficient combinatorial auctions. *Journal of the ACM (JACM)*, 49(5):577–602, 2002.

[162] Shuchi Chawla, Jason D Hartline, David L Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 311–320. ACM, 2010.

[163] Saeed Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *SIAM Journal on Computing*, 43(2):930–972, 2014.

[164] ShiGuang Wang, Ping Xu, XiaoHua Xu, ShaoJie Tang, XiangYang Li, and Xin Liu. Toda: Truthful online double auction for spectrum allocation in wireless networks. In *New Frontiers in Dynamic Spectrum, 2010 IEEE Symposium on*, pages 1–10. IEEE, 2010.

[165] Zongpeng Li, Baochun Li, and Yuefei Zhu. Designing truthful spectrum auctions for multi-hop secondary networks. *IEEE Transactions on Mobile Computing*, 14(2):316–327, 2015.

[166] Shamik Sengupta and Mainak Chatterjee. Designing auction mechanisms for dynamic spectrum access. *Mobile Networks and Applications*, 13(5):498–515, 2008.

[167] Yinglei Teng, Yong Zhang, Fang Niu, Chao Dai, and Mei Song. Reinforcement learning based auction algorithm for dynamic spectrum access in cognitive radio networks. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1–5. IEEE, 2010.

[168] Yedugundla Venkata Kiran and Raffaele Giaffreda. A dynamic pricing method for efficient radio resource management in wireless access networks. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–5. IEEE, 2011.

[169] Sudharman K Jayaweera, Mario Bkassiny, and Keith A Avery. Asymmetric cooperative communications based spectrum leasing via auctions in cognitive radio networks. *IEEE transactions on wireless communications*, 10(8):2716–2724, 2011.

[170] Martin Gaedke and Klaus Turowski. Generic web-based federation of business application systems for e-commerce applications. *EFIS*, 1999:25–42, 1999.

[171] Ligang Cong, Huamin Yang, Yanghui Wang, and Xiaoqiang Di. An auction-gaming based routing model for leo satellite networks. In *International Conference on Machine Learning and Intelligent Communications*, pages 498–508. Springer, 2017.

[172] Misbah Liaqat, Victor Chang, Abdullah Gani, Siti Hafizah Ab Hamid, Muhammad Toseef, Umar Shoaib, and Rana Liaqat Ali. Federated cloud resource management: Review and discussion. *Journal of Network and Computer Applications*, 77:87–105, 2017.

[173] Cuong T Do, Nguyen H Tran, Eui-Nam Huh, Choong Seon Hong, Dusit Niyato, and Zhu Han. Dynamics of service selection and provider pricing game in heterogeneous cloud market. *Journal of Network and Computer Applications*, 69:152–165, 2016.

[174] Adel Nadjaran Toosi, Kurt Vanmechelen, Farzad Khodadadi, and Rajkumar Buyya. An auction mechanism for cloud spot markets. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 11(1):2, 2016.

[175] Ron Lavi, Ahuva Mu'Alem, and Noam Nisan. Towards a characterization of truthful combinatorial auctions. In *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*, pages 574–583. IEEE, 2003.

[176] Mohammad H Hajiesmaili, Lei Deng, Minghua Chen, and Zongpeng Li. Incentivizing device-to-device load balancing for cellular networks: An online auction design. *IEEE Journal on Selected Areas in Communications*, 35(2):265–279, 2017.

[177] Changyan Yi, Jun Cai, and Gong Zhang. Online spectrum auction in cognitive radio networks with uncertain activities of primary users. In *Communications (ICC), 2015 IEEE International Conference on*, pages 7576–7581. IEEE, 2015.

[178] Alex Stamos. An update on information operations on facebook, Sep 2017. [Online; posted 06-Sep-2017].

[179] Larry Kim. Facebook ads, fake news and the shockingly low cost of influencing an election [data], Oct 2017. [Online; posted 11-Oct-2017].

[180] Paul Chadwick. Why fake news on social media travels faster than the truth, Mar 2018. [Online; posted 19-Mar-2018].

[181] Jeffrey Gottfried and Elisa Shearer. News use across social media platforms 2017, 2017. Available: http://www.journalism.org/2017/09/07/news-use-across-social-media-platforms-2017/, accessed 2018-02-14.

[182] Maureen Dowd. To tweet or not to tweet, 2009. Available: http://www.nytimes.com/2009/04/22/opinion/22dowd.html, accessed 2018-02-14.

[183] Yoni Heisler4. Twitter?s 280 character limit increased engagement without increasing the average tweet length, 2018. Available: http://bgr.com/2018/02/08/twitter-character-limit-280-vs-140-user-engagement/, accessed 2018-02-08.

[184] Manya Sleeper, Rebecca Balebako, Sauvik Das, Amber Lynn McConahy, Jason Wiese, and Lorrie Faith Cranor. The post that wasn't: exploring self-censorship on facebook. In *Proceedings of the 2013 conference on Computer supported cooperative work*, pages 793–802. ACM, 2013.

[185] Mark Kaufman. Why twitter is still teeming with bots, Oct 2017. [Online; posted 16-Oct-2017].

[186] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36, 2017.

[187] Jan Kratzer, Christopher Lettl, Nikolaus Franke, and Peter A Gloor. The social network position of lead users. *Journal of Product Innovation Management*, 33(2):201–216, 2016.

[188] Derek Davis, Gerardo Figueroa, and Yi-Shin Chen. Socirank: identifying and ranking prevalent news topics using social media factors. *IEEE transactions on systems, man, and cybernetics: systems*, 47(6):979–994, 2017.

[189] Juan Cigarrán, Ángel Castellanos, and Ana García-Serrano. A step forward for topic detection in twitter: An fca-based approach. *Expert Systems with Applications*, 57:21–36, 2016.

[190] Carlo Lipizzi, Dante Gama Dessavre, Luca Iandoli, and Jose Emmanuel Ramirez Marquez. Towards computational discourse analysis: A methodology for mining twitter backchanneling conversations. *Computers in Human Behavior*, 64:782–792, 2016.

[191] Tomasz Hachaj and Marek R Ogiela. Clustering of trending topics in microblogging posts: A graph-based approach. *Future Generation Computer Systems*, 67:297–304, 2017.

[192] Wei Xie, Feida Zhu, Jing Jiang, Ee-Peng Lim, and Ke Wang. Topicsketch: Real-time bursty topic detection from twitter. *IEEE Transactions on Knowledge and Data Engineering*, 28(8):2216–2229, 2016.

[193] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158. ACM, 2010.

[194] Riley Crane and Didier Sornette. Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of Sciences*, 105(41):15649–15653, 2008.

[195] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models, 2011.

[196] Xin Zhao and Jing Jiang. An empirical comparison of topics in twitter and traditional media. *Singapore Management University School of Information Systems Technical paper series. Retrieved November*, 10:2011, 2011.

[197] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, pages 591–600. AcM, 2010.

[198] Elmar Kiesling, Markus Günther, Christian Stummer, and Lea M Wakolbinger. Agent-based simulation of innovation diffusion: a review. *Central European Journal of Operations Research*, 20(2):183–230, 2012.

[199] Emilio Serrano and Carlos A Iglesias. Validating viral marketing strategies in twitter via agent-based social simulation. *Expert Systems with Applications*, 50:140–150, 2016.

[200] Emilio Serrano, Michael Rovatsos, and Juan A BotíA. Data mining agent conversations: A qualitative approach to multiagent systems analysis. *Information Sciences*, 230:132–146, 2013.

[201] Paul Leger, Carmen Hidalgo-Alcázar, and Manuela López. Using an agent-based model to measure the message repetition effect on twitter. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, pages 1828–1830. ACM, 2018.

[202] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, et al. Prominent features of rumor propagation in online social media. In *International Conference on Data Mining*. IEEE, 2013.

[203] Itai Himelboim, Marc A Smith, Lee Rainie, Ben Shneiderman, and Camila Espina. Classifying twitter topic-networks using social network analysis. *Social Media+ Society*, 3(1):2056305117691545, 2017.

[204] Wei Feng, Chao Zhang, Wei Zhang, Jiawei Han, Jianyong Wang, Charu Aggarwal, and Jianbin Huang. Streamcube: hierarchical spatio-temporal hashtag clustering for event exploration over the twitter stream. In *Data Engineering (ICDE), 2015 IEEE 31st International Conference on*, pages 1561–1572. IEEE, 2015.

[205] Gerard Salton. Automatic text processing: The transformation, analysis, and retrieval of. *Reading: Addison-Wesley*, 1989.

[206] Robertus Nugroho, Jian Yang, Weiliang Zhao, Cecile Paris, and Surya Nepal. What and with whom? identifying topics in twitter through both interactions and text. *IEEE Transactions on Services Computing*, 2017.

[207] Dinesh Kumar, Gaurav Baranwal, Zahid Raza, and Deo Prakash Vidyarthi. A systematic study of double auction mechanisms in cloud computing. *Journal of Systems and Software*, 125:234–255, 2017.

[208] Xu Han, Suvasri Mandal, Krishna R Pattipati, David L Kleinman, and Manisha Mishra. An optimization-based distributed planning algorithm: a blackboard-based collaborative framework. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(6):673–686, 2014.

[209] Adrien Guille and Hakim Hacid. A predictive model for the temporal dynamics of information diffusion in online social networks. In *Proceedings of the 21st international conference on World Wide Web*, pages 1145–1152. ACM, 2012.

[210] Kurt Driessens, Jan Ramon, and Thomas Gärtner. Graph kernels and gaussian processes for relational reinforcement learning. *Machine Learning*, 64(1):91–119, 2006.

Vitae

Abbas Ehsanfar

Email: aehsanfa@stevens.edu

Summary

I finished my Bachelor of Science in Electrical Engineering and Master of Science degree (MBA) in quantitative finance from Sharif University of Technology, Tehran. I competed two Bachelor dissertations in digital control systems and a networked power system of wind energy. During my master degree, I developed quantitative models (trading algorithms) for Forex and commodity markets with Prof. Bahramgiri and I proposed an asymmetric model for a public transportation network with Prof. Isaei at graduate school of management and economics (GSME). In 2013, I started my PhD research in collaboration with Prof. Heydari in Complex Networked Systems (CENS) laboratory and I continued my PhD research with Prof. Grogan in Collective Design (CODEC) laboratory and with Prof. Mansouri in 2017. My research interests included developing mechanisms in networked systems, developing auction-based algorithms for combinatorial resource allocation in federated systems, and developing interactive models to analyze and simulate spreading of adversarial content, e.g. fake news, on social network. I have worked with Prof. Rodic Blaz on developing a Agent-based social systems (ABSS) model on the latter subject.

Skills and Tools

Research: combinatorial algorithms, networked systems, social networks, agent-based simulation models, reinforcement learning

Computing: Linux, Windows, Mac OS X, CrayXE6, CentOS, ASPEN Cluster, Apache Spark, Multi-processing and multi-threading techniques

Languages: Python, R, Java, React, MATLAB, LaTeX

Machine Learning and Optimization: TensorFlow, Keras, scikit learn, gurobi

Quantitative: GRE 170/170 (2012), GMAT 51/52 (2010).

Research Experience

Advanced algorithms for combinatorial auctions in networks: mixed-integer linear program and binary search of price, non-linear search for finding prices by auctioneer, online algorithm with closed form solution for prices, virtual pricing with closed-form solution for prices.

Resource allocation model for routing/scheduling tasks in networks: mixed-integer linear program model, mathematical formulation, technical and financial constraints for non-linear optimization model.

Pricing mechanism for resource allocation in federated networks: minimizing computational complexity associated with simultaneous resource allocation and pricing, developing a linear .

Graphical and computational simulation of networked federated satellite systems: a linear operational model for combinatorial routing in networks, orbital software application for simulation of swarms and constellation of satellites, Gaussian model for reinforcement learning of bidding on resources.

Networked data analytics for StackExchange (Parent entity for Stack Overflow) including text mining, clustering, and regression models on 100GB of questions and responses.

Analytics of discourse on Twitter: querying twitter API to collect data, clustering models of viral content, statistical model of user behavior, and agent-based model of interactive social networks.

Developed classification technique to predict insurance plan of customers given their online activities (Kaggle competition).

A statistical model of demand prediction in biking stations at Washington DC.

Education

Stevens Institute of Technology,	Hoboken, NJ
Systems and Software Division, SSE	
Doctoral Student, Systems Engineering	$2013 - 2015$
Doctoral Candidate, Advisor: Dr. Heydari	$2015 - 2016$
Doctoral Candidate, Advisors: Dr. Grogan, Dr. Mansouri	$2017 - 2018$

Dissertation: Allocative Mechanisms for Resource and Information Exchange in Task Processing Networks of Elements and Interactive Social Networks. Defended: October 2018, Conferral: December 2018.

University of California Los Angeles, Graduate summer school in cyber-physical networks, Summer 2016.

Sharif University of Technology, Tehran, Iran

M.Sc. in Finance (MBA), $2007 - 2010$

BSc in Electrical Engineering, $2003 - 2007$

Industry Experience

Joinery NYC, New York, NY

Full Stack Software Developer May-August 2016

Incubator: EBAY NYC.

Tosan Intelligent Data Miners (TIDM), Tehran

Business Intelligence Project Coordinator/Developer $2011 - 2013$

Publications & Presentations

Ehsanfar, A, and Paul. T. Grogan, *Auction-based Algorithms for Resource Allocation in Federated Networks*, Journal of Networks and Systems Management (2018), Submitted.

Ehsanfar, A, and Mo Mansouri (2018, November), *An Influence-based Clustering Model on Twitter.* INFORMS Annual Meeting 2018. 13^{th} Data Mining and Decision Analytics Workshop, Phoenix, AZ, Presented.

Ehsanfar, A, and Mo Mansouri, *A Networked Model of Influence-based Discourse on Twitter: Exogenous or Endogenous?*, IEEE Transactions on Systems, Man, and Cybernetics (2018), submitted, (Under Revision).

Ehsanfar, A, and Paul T. Grogan, *Mechanism Design for Sharing Resources in Federated Networks*, Journal of Networks and Systems Management (2018), Reviews Received/Resubmitted.

Ehsanfar, A., and Mansouri, M. (2017, June). *Incentivizing the dissemination of truth versus fake news in social networks.* In System of Systems Engineering Conference (SoSE), 2017 12th (pp. 1-6). IEEE.

Rodic, B, Ehsanfar A., and Mansouri M. *A Simulation Model for Dissemination of Fake News in Social Networks*, Working paper.

Ehsanfar, A, and B. Heydari. *An Incentive-Compatible Scheme for Electricity Cooperatives: An Axiomatic Approach.*, IEEE Transactions on Smart Grid(2016).

Ehsanfar, A., Farzinfard, S., and Isaai, M. T. (2008, October). *An asymmetric intelligent model for public transportation networks*. In Intelligent Transportation Systems. ITSC 2008. 11^{th} International IEEE Conference on (pp. 511-516). IEEE.

Ehsanfar, A. and Heydari, B. (2015, April). *Interactive Multi-Consumer Power Cooperatives with Learning and Axiomatic Cost and Risk Disaggregation*. In AAAI Workshop: Computational Sustainability.