SANDIA REPORT

2014-0548
Unlimited Release
Printed January, 2014

Integrated Network Design and Scheduling Problems: Optimization Algorithms and Applications

Sarah G. Nurre, Jeffrey J. Carlson

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

2014-0548
Unlimited Release
Printed January, 2014

Integrated Network Design and Scheduling Problems: Optimization Algorithms and Applications

Sarah G. Nurre

Abstract

We consider the class of integrated network design and scheduling problems. These problems focus on selecting and scheduling operations that will change the characteristics of a network, while being specifically concerned with the performance of the network over time. Motivating applications of INDS problems include infrastructure restoration after extreme events and building humanitarian distribution supply chains. While similar models have been proposed, no one has performed an extensive review of INDS problems from their complexity, network and scheduling characteristics, information, and solution methods.

We examine INDS problems under a parallel identical machine scheduling environment where the performance of the network is evaluated by solving classic network optimization problems. We classify that all considered INDS problems as NP-Hard and propose a novel heuristic dispatching rule algorithm that selects and schedules *sets* of arcs based on their interactions in the network.

We present computational analysis based on realistic data sets representing the infrastructures of coastal New Hanover County, North Carolina, lower Manhattan, New York, and a realistic artificial community CLARC County. These tests demonstrate the importance of a dispatching rule to arrive at near-optimal solutions during real-time decision making activities.

We extend INDS problems to incorporate release dates which represent the earliest an operation can be performed and *flexible* release dates through the introduction of specialized machine(s) that can perform work to move the release date earlier in time. An online optimization setting is explored where the release date of a component is not known.

Acknowledgment

Thank you very much to Sandia National Laboratory for the generous support to complete my dissertation at Rensselaer Polytechnic Institute. Many thanks to my advisor, Dr. Thomas C. Sharkey, for his guidance, continued friendship, and collaboration. I extend sincere gratitude to Jeffrey J. Carlson, my Sandia advisor, for his understanding and guidance throughout the process.

Note, the LDRD program was used to fund my graduate school and dissertation. Due to this fact, much of the report following is similar to my dissertation, entitled *Integrated Network Design and Scheduling Problems: Optimization Algorithms and Applications*, which is a compilation of published papers, papers under review, and papers in preparation for submission.

Contents

1 Introduction **9**

2 Literature Review **13**

3 Problem Statement **19**

4 Complexity Results and Solution Methods **23**

Complexity Results 23

Dispatching Rules 24

Integer Programming Formulations 26

5 Computational Results **29**

Performance analysis through Optimality Gaps and Computational Time . . . 31

Machine Composition and Utilization Analysis 40

6 Conclusions **53**

References **56**

List of Figures

5.1 CLARC County power network 30

5.2 Hurricane Sandy impact 31

5.3 Average optimality gap by damage and instance. 32

5.4 Average computation time by damage and instance. 32

5.5 Average optimality gap for $Pm|MF, r_{ij}|Cumulative$ 33

5.6 Average optimality gap for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ with $m = 1$ 35

5.7 Average optimality gap for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ with $m = 2$ 35

5.8 Average optimality gap for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ with $m = 3$ 36

5.9 Average optimality gap for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ with $m = 4$ 36

5.10 Average optimality gap for $MFM - Pm|MF, flex - r_{ij}|Cumulative$ 40

5.11 Average objective function value for combinations of 2 machines. 42

5.12 Average objective function value for combinations of 3 machines. 43

5.13 Average objective function value for combinations of 4 machines. 43

5.14 Average objective function value for combinations of 5 machines. 44

5.15 Average objective function value for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ by damage and machine profile 45

5.16 Average utilization for $Pm|MF, r_{ij}|Cumulative$ by damage and machine 46

5.17 Average utilization for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ by solution method, damage, and machine profile 47

5.18 Average utilization for $Opt - FF2|MF, flex - r_{ij}|Cumulative$ by solution method, δ, and machine profile 48

5.19 Average utilization for $MFM - Pm|MF, flex - r_{ij}|Cumulative$ by damage and machine 51

List of Tables

4.1 Summary of the complexity results 23

4.2 Customized selection routine 27

4.3 Decision variables used in the INDS IP formulations. 27

5.1 Power flow reduction based on network damage 30

5.2 $Pm|MF, r_{ij}|Cumulative$ computational results 34

5.3 $Opt - FF2|MF, flex - r_{ij}|Cumulative$ computational results for $\delta = 0.1$ . . . 37

5.4 $Opt - FF2|MF, flex - r_{ij}|Cumulative$ computational results for $\delta = 0.2$ . . . 38

5.5 $Opt - FF2|MF, flex - r_{ij}|Cumulative$ computational results for $\delta = 0.3$ . . . 39

5.6 $MFM - Pm|MF, flex - r_{ij}|Cumulative$ computational results 41

5.7 $Pm|MF, r_{ij}|Cumulative$ average utilization 46

5.8 $Opt - FF2|MF, flex - r_{ij}|Cumulative$ average utilization for $\delta = 0.1$ 49

5.9 $Opt - FF2|MF, flex - r_{ij}|Cumulative$ average utilization for $\delta = 0.2$ 49

5.10 $Opt - FF2|MF, flex - r_{ij}|Cumulative$ average utilization for $\delta = 0.3$ 50

5.11 $MFM - Pm|MF, flex - r_{ij}|Cumulative$ average utilization 52

Chapter 1

Introduction

Integrated network design and scheduling (INDS) problems are a new class of problems that model the selection and scheduling of operations that will change the characteristics of a network. These operations could correspond to enhancing existing network components or installing new components. The integration of these two sets of decisions, network design and scheduling, allows our models to capture how the operations improve the performance of the network over time during their implementation.

Traditional network design problems look at selecting the desired characteristics (e.g., which components to install) of a network. A network design is evaluated by looking at the end performance of the network, i.e., how the network operates after the design is completed, without considering the intermediate network performance as the design is being implemented. INDS problems specifically consider the intermediate network performance by incorporating the scheduling decisions required to implement the selected design. These INDS scheduling decisions are related to traditional scheduling problems that seek to allocate resources over time to process a set of tasks.

INDS problems focus on the allocation of machines[1] to process the set of tasks required to achieve a selected network design. The focus of this dissertation will be on INDS problems in a parallel identical machine environment, with some extensions and specific machine characteristics. A solution to these problems involves three main decisions: *selecting* which components to process to change the network, *assigning* the selected components to a machine, and *sequencing* the assigned tasks on each machine. We consider INDS problems with objectives that focus on the cumulative (intermediate) performance of the network over time or that focus on how quickly a desired level of network performance can be achieved.

There are a number of real-life situations that motivate the need for models that integrate network design and scheduling problems. We present applications in restoring a disrupted network, upgrading an existing network, and building a new network.

Our first application arises after an extreme event, such as a hurricane, that has caused large-scale damage to an infrastructure network. Infrastructure managers must make a restoration plan to repair damaged components and re-establish distribution of the services provided by the network. Customers of the infrastructure network evaluate the success of a restoration plan by how quickly their services are restored. Therefore, the repair of damaged components must be scheduled to create an operational network and ensure a quick

[1]Please note that while we use the term 'machine', in many applications, work groups, work crews, or other resources take the role of the machines. However, for consistency with the scheduling literature, we will use the term machine when examining the mathematics of the INDS problem.

restoration of services to many customers. The infrastructure will be providing services as it is implementing its restoration plan and, therefore, infrastructure managers will be concerned with the level of satisfied demand in the network over time. An INDS model can help formulate a restoration plan that allocates infrastructure work crews to process selected repairs in the infrastructure network.

There are many external factors to consider when devising a restoration plan, many of which were experienced after Hurricane Sandy in late October 2012. For example, infrastructures are interdependent in that one infrastructure depends on the services provided by another infrastructure for normal operation and restoration activities. For example, a flooded subway tunnel needs to be pumped free of water which requires power or the use a generator which requires fuel. The transportation network is needed for power crews to move from one restoration task to another. If flooding or debris is present on the roadway, the power managers must consider if a restoration task is accessible and safe to perform. After an extreme event like Hurricane Sandy, different restoration tasks must be completed by different infrastructure systems. For example, debris from a roadway should be removed by license debris contractors. Therefore, if debris is blocking access to a power restoration task, the power company must wait or coordinate with the debris removal crews to gain access to their task. We extend INDS problems to consider many of these external factors by incorporating new network and scheduling characteristics.

Another application area is the implementation of network upgrades in order to improve services provided by the network. One example of this is constructing a new smart grid design for the power infrastructure. Researchers have looked into how to transform the electrical power grid into a smart grid (see Farhangi [25], Momoh [56], Mahmood et al. [51], DeBlasio and Tom [18]), many of which include installations of solar (see Mulder et al. [58]), wind (see Glinkowski et al. [27]), and other environmentally friendly and sustainable technologies (see Clastres [14], Moslehi and Kumar [57], and Liserre et al. [48]). For large projects like these, resources (e.g., budget) often only allow a number of upgrades to take place during a certain time period (e.g., annually). It is, therefore, important to determine which upgrades will be completed in each phase of the project, which can be modeled with an INDS problem. Note that, in this application area, it is likely that all upgrades (tasks) in the problem will be processed (i.e., the end design is fixed); however, INDS problems can help model the order of the tasks in order to optimize the intermediate network performance. Another example of an application of upgrading a network is when a supply chain chooses to enhance existing components in their network, such as improving a machine or expanding inventory capacity at a facility.

Our third application of INDS problems involves creating and building a new network, which is often needed during humanitarian logistics activities. After an extreme event, many people do not have access to essential items such as water, food, and shelter. In remedy of this situation, humanitarian organizations set up locations to distribute these essential items and to act as shelters. These sites should be located in order to provide the greatest benefit to the affected community. There are limited resources available to set up these locations, therefore, we must allocate the resources over time to a selected subset of the potential distribution and shelter sites. It is clear that the partially completed network will need to be operational and goods will be distributed even when all work is not fully completed on the selected logistics network.

The objective of an INDS problem is based on two factors: (i) how the performance of the network is evaluated and (ii) how we evaluate the scheduling decisions. The performance of the network is often dictated by the application of the INDS problem. For example, the type of infrastructure system being restored will dictate the most applicable mathematical model to apply to determine the performance of the network. A power network might care about maximizing flow at each time period of a restoration plan, which is equivalent to maximizing met demand. An emergency network might care about the shortest path between residents and a hospital. We will consider network performance metrics based on four classic network flow problems (see Ahuja et al. [2]). In terms of evaluating the scheduling decisions, we are interested in determining the cumulative performance of the network over time, which means our objective will include the network performance in each period. Alternatively, we are also interested in determining the minimum amount of time required to achieve some set performance value (e.g., all demand is met).

INDS problems are also impacted by the scheduling characteristics associated with the tasks and the machines. During Hurricane Sandy, and after many extreme events, some restoration tasks are not available for immediate processing directly after the event has passed. Often times flooding needs to subside or be pumped out of areas before it is safe for work crews to enter the area surrounding restoration tasks, or debris needs to be removed from roadways to allow access to the restoration tasks. We utilize release dates associated with INDS tasks to represent the earliest point in time a task can begin to be processed. This could, for example, signify the point in time when debris has been removed from a roadway providing access to a set of power restoration tasks. For the application of infrastructure restoration, and many of the other applications that INDS problems can model, the release date time for a particular task is unknown at the start of the decision making process. For example, the power company many not know when a roadway will be clear of debris, allowing access to power restoration tasks, until they receive communication from the county that the debris has been removed. We use an online optimization environment to model this lack of complete information with regards to release dates for INDS problems.

If a vital restoration task for the power company has an unknown release date, or release date far into the future they will want to take actions to move this release date earlier in time. We introduce the novel concept of *flexible* release dates, which allows a set of specialized machines to perform processing (associated with a specific task) to move the release date earlier in time, specifically to the completion time of the specialized machines. For example, the power company could hire licensed work crews to remove the debris blocking the roadway that obstruct the access to a set of power restoration tasks. The new release date for the set of power restoration tasks would then be the completion time of the hired debris removal work crews. We analyze this further by considering the benefits of multi-function machines that perform processing to move the release dates earlier in time (e.g. remove debris) and traditional processing to make components operational in the network.

For our examination of INDS problems, we will first discuss relevant literature in Chapter 2. Work examining both network design and scheduling decisions, separately will be presented in addition to an overview of the limited articles that discuss network design and scheduling decisions jointly. Within these discussions, we outline how this work is different and novel. Furthermore, we present specific literature concerning real life applications that can be modeled using an INDS problem. In Chapter 3 we formally introduce and define

INDS problems in a parallel machine environment. The contributions of this work to the study of INDS problems include: (i) the examination of 12 INDS problems with different network performance metrics and scheduling objectives (which can model many of the previous objectives considered in related work), (ii) introduction of release dates and *flexible* release dates, and (iii) and online optimization framework. In Chapter 4 we classify each of the 12 INDS problems as at least *NP*-Hard. We then propose exact and heuristic solution methods through integer programming formulations and dispatching rule algorithms. We then computationally test the solution methods in Chapter 5 on realistic data sets representing the power, waste water, and emergency supply chain infrastructures of coastal New Hanover County, North Carolina, power and telecommunication networks of lower Manhattan, New York, and the power network of a realistic artificial community CLARC County. These tests demonstrate the importance of a dispatching rule to arrive at near-optimal solutions during real-time decision making activities. We conclude with Chapter 6.

The main contribution of this work is an extensive analysis of the new class of INDS problems from their complexity, network and scheduling characteristics, level of information known, to solution methods.

Chapter 2

Literature Review

In this chapter, we discuss previous work relevant to integrated network design and scheduling problems. First, literature will be cited that looks at the integration of these two sets of decisions. Then, we go more in depth into specifics of scheduling which provide the ground work for ideas associated with the proposed dispatching rules, flexible release date concepts, and extensions. This includes citations about specific solution methods that help frame the proposed dispatching rule. Online optimization is then explored in the context of traditional network design and scheduling problems. Lastly, relevant literature based on the applications of Integrated Network Design and Scheduling (INDS) problems is presented.

Network Design and Scheduling Problems looking at the integration of network design and scheduling decisions have not been widely examined, despite their important applications. Network design problems have been well studied and look to find the best placement of nodes and arcs in a network to balance design costs with the performance of the *end* network. Many network design applications have also been explored. The nodes of a network could model facility locations (see Daskin [17] and Drezner [19]) or the arcs could model the evacuation routes of a city (see Kalafatas and Peeta [43] and Baumann and Skutella [7]). Scheduling problems have also been independently studied (see Pinedo [65]). We will explore literature examining network design and/or scheduling decisions that are relevant to the applications of INDS problems.

Lee et al. [47] use an interdependent layered network to aid in the restoration of many infrastructures after an extreme event. This interdependent layered network models interdependency between infrastructures. The model selects arcs that are a priority to install into the network. A data set representing lower Manhattan after a disruption to the area surrounding the Brooklyn Battery Tunnel is created by Lee et al. [47] and used for testing. While Lee et al. [47] presents a network design restoration plan, this plan only incorporates the selection decisions of arcs to install back into the network. A schedule for how to efficiently assign the arcs to work groups at specific times is not developed.

Lee et al. [47] does not address the scheduling decisions but Gong [28] does develop a model to find the best schedule of arcs for restoration after an extreme event. Logic-based Benders decomposition is used where a constrained programming subproblem feeds into a master problem by adding additional constraints. Gong assumes that all arcs in the problem need to be installed. This is true for the long term restoration efforts but initially short term efforts need to be analyzed. Often a network has redundancy and if two redundant paths are

damaged it is not in our best interest to repair them both before repairing another damaged path that could restore more demand. Further, only looking at the scheduling decisions does not capture how the arcs interact with each other. Therefore, selection decisions are necessary for truly accomplishing the objective of maximizing cumulative satisfied demand. Further, Gong [28] does not consider interdependencies by factoring in how different infrastructure networks interact with one another.

These two models are combined by Cavadaroglu et al. [10] into a mixed integer programming formulation. Both network design with selection and scheduling decisions are addressed. The objective function minimizes the cumulative sum of flow costs and unmet demand costs. When the model is tested on the Manhattan data set developed by Lee et al. [47], the run time is very long and in the study had not been run to optimality. Further pre-processing was done to reduce the amount of arcs that are available for selection. The formulation of Lee et al. [47] was used to determine the 100 best arcs that should be installed. The selection decisions for this model were then limited to these 100 arcs. A heuristic method is also presented, but is specialized for networks with a large connected subsection fully damaged.

The works of Guha et al. [31], Averbakh [5], and Averbakh and Pereira [6] consider INDS problems that are focused on the *recovery times* of the nodes in the network, which is defined as the first time a (demand) node is connected to a 'source' (or supply) node. These works do not consider the capacity constraints within a network, which is important when capturing the network performance in certain applications. Further, INDS problems considered in this dissertation can capture the number of recovered nodes in the network at any point in time, so the objectives can model recovery time-based objectives. Xu et al. [75] look at restoring a power network after an earthquake by scheduling inspection, assessment, and repair operations. This work, again, focuses on the recovery time of each demand node and proposes a genetic algorithm for this problem.

There has further been research that have looked at INDS problems that is interested in the cumulative performance of the network over time. Ang [4] provides an integer programming formulation of a model to restore power, where the focus is minimizing the total cost of power shed over time. Matisziw et al. [53] provides an integer programming formulation of an INDS problem whose network performance is measured as the connectivity between pairs of nodes. Stilp et al. [71] examine debris removal in a transportation infrastructure and how it can reduce long-term effects of an extreme event. Concurrent to this work, Elgindy et al. [22] has considered an INDS problem where we can install one new arc in every time period and is focused on minimizing the total of the length of the shortest path over all time periods. Therefore, this problem is a special case of problems considered in this dissertation. Elgindy et al. [22] provides a *NP*-Hard proof of this problem and approximation algorithms.

Note that other work has examined network design problems that schedule routes and services on the design (see Lederer and Nambimadom [46], Cranic [16], Lai and Lo [45], Agarwal and Ergun [1], and Guihaire and Hao [32]). These problems are fundamentally different than INDS problems since INDS problems look at utilizing machines to schedule the processing of components to make a selected network design operational.

Scheduling and Solution Approaches Pinedo [65] provides a thorough review of scheduling problems. Heuristics, dispatching rules, and algorithms are presented to achieve near optimal solutions for *NP*-Hard scheduling problems, and optimal solutions for polynomial solvable problems. Two common scheduling heuristics that are modified for use in the INDS dispatching rule framework are the WSPT rule and LPT rule.

The Weighted Shortest Processing Time (WSPT) rule was originally introduced by Smith in 1956 [69]. This rule was proven to be optimal for the problem of minimizing total weighted completion time for a single machine, or $1||\sum w_j C_j$, when using the classification notation by Graham et al. [30], where C_j is the completion time of job j and w_j is the weight or priority. According to the rule, the next job scheduled has the greatest ratio of w_j to p_j, where p_j is the processing time of job j.

The Longest Processing Time (LPT) first rule (see Graham [29]) can be applied to $Pm||C_{max}$, minimizing maximum completion time, C_{max}, on m parallel machines. As the name states, the job with the longest processing time is scheduled first on the next available machine. This rule is not exact but provides a solution that is guaranteed to be less than or equal to $\frac{4}{3} - \frac{1}{3m}$ times the optimal solution.

We extend INDS problems to include release dates (see Pinedo [65]) and flexible release dates. The use of another machine to move earlier the release date of a job, as is done for flexible release, has not been widely studied. Dubois et al. [20] discuss flexible release dates but in the context that the release date constraint is not strict but instead a soft constraint. Instead of modeling that a machine cannot start job j until its release date r_j, a 'fuzzy' or soft release date constraint is used that allows an early start time in exchange for a penalty. This constraint allows for a range or threshold of possible start times and factors in a value measuring how satisfied one is for the start time of a specific job. This definition of a flexible release date differs from our flexible release dates. We model the use of an external specialized machine to move forward a release date to a specific point in time and not a soft constraint for the release date. In addition, we allow the possibility for multi-function machines that can process tasks and shift release dates.

Online Optimization We consider INDS problems in an online optimization environment. Online optimization problems have incomplete information at the start, but information is incrementally revealed. Therefore, it is imperative for decision makers to arrive at decisions based on incomplete information and adapt to new information. Online algorithms have been studied by Sleator and Tarjan [68], who also introduced the notion of comparing an online algorithm performance to the offline performance. The offline to online comparison was then formalized as the online competitive ratio by Karlin et al. [44]. For information about the study of online algorithms, competitive ratios, and their many applications please see the following books of Borodin and El-Yaniv [9] and Fiat and Woeginger [26].

An example of an online scheduling problem could involve incomplete information about the set of jobs that need to be processed and how long the processing on machines will take (see Pinedo [65]). Hall et al. [36] examine the single machine scheduling problem seeking to minimize total weighted completion time with precedence constraints in an online setting. Hoogeveen and Vestjens [37] examine single machine online problems where the number of

jobs is not known. Stougie and Vestjens [72] examine randomized algorithms for a variety of online scheduling problems. The online single machine total weighted completion time scheduling problem is considered by Anderson and Potts [3]. Megow and Schulz [54] examine the online average completion time scheduling problem. Chou et al. [13] consider parallel machine scheduling problems with release dates in an online setting. A variety of solution approaches and permutations of single and parallel machine scheduling problems seeking to minimize the total weighted completion time are examined by Correa and Wagner [15].

An introductory tutorial about online optimization problems can be found by Jaillet and Wagner [42]. In the tutorial and more extensively in Jalliet and Wagner [39], they examine the online Traveling Salesman Problem (TSP) and Traveling Repairman Problem (TRP), where cities pop up throughout the horizon of the problem. They further extend this problem by giving release dates to cities, stating that each city must be visited at or after the defined release date. By making assumptions on specifics surrounding the data in the TSP and TRP, they are able to arrive at competitive ratios for proposed online algorithms. Jaillet and Wagner [40] also consider online problems involving resource allocation, and machine scheduling [41]. Jaillet and Stafford [38] examine the online searching problem, where a searcher seeks to find an exit by taking one of m disjoint branches. They examine the value of gathering information prior to the search and its impact on their online algorithms. We perform a similar analysis by quantifying the value of advanced information and its impact on the solution quality.

Applications A main application of INDS problems is infrastructure restoration after an extreme event. When modeling this application, we should take into account the interdependency of infrastructure systems on each other. The analysis of civil infrastructure systems is complex since they are interdependent (see O'Rourke [64]); disruptions in one can spread to others causing cascading failures (see Wallace et al. [74], Mendonca and Wallace [55], and Chang et al. [12]).

Rinaldi et al. [66] note that managers of infrastructure systems have become inclined to consider these interdependencies; however, the managers of a particular infrastructure will have little knowledge of the structure and operations of the other systems. We can expect that the managers of an individual infrastructure will understand the direct connections of it with other infrastructures and, therefore, can weigh the services provided to certain connections more heavily (e.g., a hospital has a higher priority weight than a residential household).

Ellis et al. [23] discuss how an attack on the internet is a vulnerable target, based on accessibility from many countries, and has large scale impact on other critical infrastructures such as emergency services. Little [49] calls for examining the indirect effects which are first-order and then second-order interdependent effects of damage. Rinaldi et al. [66] goes into types of interdependencies including physical, cyber, geographical and logical. These 4 types are used in other literature models.

Geographical proximity is used to model interdependency between the power and water networks by Dueñas-Osorio et al. [21]. A cyber classification is used based on information technology by Haimes and Jiang [35]. O'Rourke [64] explains that interdependency is based on physical proximity and operational interaction which coincides with physical and logical.

Other models are created to further analyze the details of interdependency (see Haimes et al. [34], [33]). These classifications are useful when determining the interdependency of the data set used for realistic testing scenarios.

A resilient network is a network that can return to working order quickly after disruption. Network vulnerability displays how susceptible particular components of the network are to disruption. Through the use of an INDS problem a quick restoration plan can be achieved thereby increasing the resiliency and decreasing the vulnerability.

Many have examined specific ways to measure the resiliency of a graph. A global and local connectivity parameter, vertex degree distribution, and redundancy ratio is a technique used by Dueñas-Osorio et al. [21]. From empirical tests it was concluded that as the percentage of arcs that are damaged in the network increases, there is a very fast decay of satisfied demand in both the power and water infrastructures. Another observation was that a graph is more resilient when arcs are randomly removed than with a targeted disruption representative of a deliberate attack.

O'Rourke [64] emphasizes the idea of resiliency through a people approach. He promotes resiliency through public awareness developed through education, strong leadership and direction, planning, and resource allocation when creating a proper network. Chang [11] gives a concrete definition of a resilient network and the challenges in achieving this type of network. She defines resiliency with 3 criteria: 'lower probability of failure, less-severe negative consequences when failure occurs, and faster recovery from failure.' Through the implementation of INDS problems we can enable the network to recover faster, thereby increasing the resiliency measure of the network. Another definition provided by Tierney and Bruneau [73] uses the R4 framework: robustness, redundancy, resourcefulness, and rapidity as the parameters for measurement. Again, INDS problems specifically address the rapidity and resourcefulness measures of resiliency.

One way of modeling the vulnerability of a network, or a particular component of a network, is by measuring connectivity. Haimes and Jiang [35] present a model based off the Leontief input-output model measuring economic interconnectedness. Simulation techniques are also used to measure connectivity through the use of many experimental scenarios by Matisziw et al. [52]. Upper and lower bounds on flow between origins and destinations are also established through an integer programming formulation by Murray et al. [59]. Both connectivity and bounds on flow allow for knowledge on how the removal of a particular arc will impact the total performance of the network.

Haimes et al. [34] use an inoperability input-output model (IIM) to assess effects between connected and hierarchical systems. Case studies performed by Haimes et al. [33] then use this IIM model to identify the top most vulnerable infrastructure systems. A further component ranking is provided by Nagurney and Qiang [60] that identifies the components that have the greatest impact when damaged in the network. Another assessment of vulnerability provided by Bienstock and Verma [8] answers the question if k components can be damaged or removed without causing failure.

Another main application of INDS problems is humanitarian logistics. We will present case studies using INDS problem to decide where to appropriately locate distribution sites for essential goods to the affected population after an extreme event. Sonmez and Lim [70] examine a similar problem, focusing on facility location and relocation for these distribution sites based on changing demand. Their work differs from INDS problems because set up,

take down, and movement time is not factored into their model. Further, they do not manage the decisions surrounding resources needed to perform these relocation duties.

A good overview of how optimization can help disaster supply chains is provided by Ergun et al. [24]. The problem of debris removal after a disaster is used as an illustrative example addressing a timeline and specific tasks. With the use of INDS problems, we can model different aspects of humanitarian logistical activities. We can determine the creation of a disaster supply chain by deciding on the location and allocation of goods to the affected community from distribution centers after a disaster. Further, we can model debris removal by determining which transportation network components (e.g. roads) are most vital and scheduling resources to remove the debris to make the transportation functional.

After an extreme event, such as Hurricane Sandy in October 2012, it is apparent that the operations of one infrastructure depend on the services provided by another infrastructure. The idea of these operational interdependencies is addressed by Rinaldi et al. [66] , Little [49], and Wallace et al. [74]. These operational interdependencies are extended further to the notion of restoration interdependencies by Sharkey et al. [67]. Restoration interdependencies occur when the restoration task of one infrastructure is impacted by the restoration tasks associated with another infrastructure. For example, a flooded subway tunnel after Hurricane Sandy needs to be pumped free of water which is a restoration task in the subway infrastructure system requiring power. If power is out for the subway tunnel, then this is an example of a restoration interdependency, as the subway restoration task is impact by when the power restoration task is complete.

We use INDS problems with release dates and flexible release dates to capture some level of restoration interdependencies. For the example of the restoration interdependency between the subway and power network, if power was absolutely necessary to pump out the subway tunnel (no generators or other options are available) the release date associated with the pumping restoration task would be the time that the power is restored. Knowing this exact point in time is often desirable, but unrealistic. We use online INDS problems to model this lack of complete information. We can model the situation when the subway has alternative options to the power restoration, such as acquiring generators as an INDS problem with flexible release dates. The release date associated with the restoration task of pumping out the tunnel can be moved earlier in time, specifically to the time when a generator is brought to the tunnel and hooked up.

Chapter 3

Problem Statement

The purpose of this section is to provide a formal problem definition of INDS problems. This includes specifying the network characteristics, network performance metrics, objective functions, and scheduling environment associated with INDS problems.

Let $G_t = (N, A_t, A'_t)$ represent the network at time t, where N represents the set of nodes, A_t represents the set of operational arcs at time t, and A'_t represents the set of non-operational arcs at time t. The performance of the network, denoted by $P(G_t)$, is directly impacted by the set of operational arcs, A_t. The set of arcs $A_t \cup A'_t$ remains constant over time. An arc $(i, j) \in A'_t$ can become operational and transition to $A_{\bar{t}}$, for some $\bar{t} > t$ once it has been processed by a machine. The initial operational network is made up of nodes, N, operational, A_0, and non-operational, A'_0, arcs at time 0.

The set of nodes is not indexed by t because we assume all processing in the network occurs on the arcs. This is without loss of generality since the potential to install a node in the network can be modeled as installing an arc using standard network expansion techniques (e.g., split the node into two nodes and an arc). Within the set of nodes N, we may have (depending on the network performance metric) a subset of supply nodes S and demand nodes D. Each supply node $i \in S$ has an associated supply level, s_i, and each demand node $i \in D$ has an associated demand level d_i. Define a priority, ω_i for each demand node $i \in D$.

Each arc has a set of associated parameters. Arc $(i, j) \in A'_0$ has an associated (constant) processing time $p_{ij} > 0$, representing the length of time it needs to be assigned to a machine to become operational. Every arc $(i, j) \in A_0 \cup A'_0$ has an associated capacity u_{ij} and cost c_{ij}. Define parameter r_{ij} to be the release date for each arc $(i, j) \in A'_0$. Mimicking the release date definition from scheduling literature (see Pinedo [65]), the release date of an arc $(i, j) \in A'_0$ represents the earliest that any machine can begin processing (i, j).

Machines must be allocated to a non-operational arc for it to become operational. We assume m parallel identical machines in a non-preemptive environment, commonly denoted Pm. Parallel identical machines means that all machines can complete arc $(i, j) \in A'_0$ in p_{ij} time periods. The non-preemptive environment signifies that once a machine starts to work on arc $(i, j) \in A'_0$ it must continue to work on arc (i, j) for p_{ij} consecutive time periods to complete arc (i, j). In a traditional scheduling environment, jobs tend to arrive at machines. However, for INDS problems a machine might need to relocate to a specific portion of the network to process its next task. Our models do not specifically consider this routing/relocation time since our motivating applications often have that this time is on a much smaller scale than the processing times associated with the arcs and is, therefore, negligible.

Starting from INDS problems with release dates, we broaden the definition to allow for

flexible release dates. The main idea behind a flexible release date is that some arc with a set release date r_{ij} has the opportunity to be processed on a specialized machine for $\bar{p}_{ij}$ time periods to move the release date earlier in time. Define machines $\ell = 1, \ldots, L$ as the specialized parallel identical machines. Each arc $(i, j) \in A'_0$ now has a specialized processing time $\bar{p}_{ij}$, which represents the amount of time needed for processing on some specialized machine to move up the release date. If an arc (i, j) is processed on a specialized machine, its new release date becomes the completion time of this arc on the specialized machine. If each arc was required to be processed first by a specialized machine and then a traditional machine, we would be in a flexible flow shop environment, $FF2$. Because the processing on the specialized machine is optional, as we can choose to instead wait for the release date, we denote the machine environment as $Opt - F2$, and the incorporation of flexible releases dates by $flex - r_{ij}$.

For INDS problems with flexible release dates, two types of disjoint machines are considered: m machines that process arcs to become operational in the network and L machines that process arcs to move release dates earlier in time. These are disjoint sets because the m machines can *only* process arcs to become operational and cannot process arcs to move release dates earlier. The converse is true for the L specialized machines. We relax this assumption by considering INDS problems with multi-function machines. Denote $\mu = 1, \ldots, Mu$ as parallel identical multi-function machines that can perform two tasks for a non-operational arc within the network: processing to make the arc operational in the network and processing to move the release date of the arc earlier in time.

We consider two objectives: the *Cumulative* and the *Threshold* objectives. The *Cumulative* objective examines a set time horizon T and optimizes the cumulative weighted network performance over time, $\sum_{t=1}^{T} w_t P(G_t)$, where w_t is a weight associated with each time period. The term 'optimize' can mean either minimize or maximize depending on the specific network performance. The *Threshold* objective considers a desired performance value P and minimizes the time needed to reach or exceed this performance value, i.e. $\min \bar{T}$ such that $P(G_{\bar{T}}) \geq P$ for maximization problems ($P(G_{\bar{T}}) \leq P$ for minimization problems).

The performance of the network $P(G_t)$ at each time period is influenced by the set of operational components at time t. Hence, as more components become operational the performance of the network will improve. The manner in which we evaluate the performance will be based on four core network flow problems (see Ahuja et al. [2]). The performance metrics are as follows:

1. Maximum flow (MF): maximize the flow from a supply node to a demand node. This includes, through standard network expansion techniques, problems with multiple supply nodes and/or demand nodes.

2. Prioritized Maximum flow (PMF): maximize the prioritized or weighted flow from a supply node to demand nodes, which can be modeled as a network with a single supply and single demand node through standard network expansion techniques.

3. Minimum cost flow (MCF): minimize the cost of flow that satisfies demand at a set of demand nodes from a set of supply nodes.

4. Shortest path

- Single source, single destination (SP): minimize the shortest path from a single source to a single destination.
- Sum of shortest paths to multiple destinations ($\sum SP$): minimize the sum of shortest paths from a single source to multiple destinations.
- Maximum path of shortest paths to multiple destinations ($\min\max SP$): minimize the maximum length of a shortest path from a source to one of a set of multiple destinations.

5. Minimum spanning tree (MST): minimize the cost of a tree that spans all nodes.

These performance metrics in combination with 2 objective functions and other extensions results in 16 total problems considered. For ease of notation we utilize Graham's triplet notation for scheduling problems (see Graham [30]) and present the problems in $\alpha \mid \beta \mid \gamma$ format where α represents the machine environment, β represents special processing characteristics (including network performance metrics) or constraints, and γ represents the objective function. We consider the following problems:

- $Pm|MF|Cumulative$
- $Pm|PMF|Cumulative$
- $Pm|MF, r_{ij}|Cumulative$
- $Opt-FF2|MF, flex-r_{ij}|Cumulative$
- $MFM{-}Pm|MF, flex{-}r_{ij}|Cumulative$
- $Pm|MF|Threshold$
- $Pm|MCF|Cumulative$
- $Pm|MCF|Threshold$
- $Pm|SP|Cumulative$
- $Pm|SP|Threshold$
- $Pm|\sum SP|Cumulative$
- $Pm|\sum SP|Threshold$
- $Pm|\min\max SP|Cumulative$
- $Pm|\min\max SP|Threshold$
- $Pm|MST|Cumulative$
- $Pm|MST|Threshold$

Online Optimization Environment We focus our attention on the maximum flow performance metric with *Cumulative* objective function for the INDS problems with release dates, flexible release dates, and multi-function machines. We further explore this problem class by examining it under two different information and optimization settings: offline and online. The 'offline' version of INDS problems is when all information, including the time when arcs will be released, is known. For example, let arc $(i, j) \in A'_0$ have a release date of r_{ij}. It is known that arc (i, j) will be released at r_{ij} at time 0 and through the entire time horizon of the problem, allowing decision makers to plan for the release of an arc ahead of time. The 'online' version of INDS problems is when release date information is known only as it occurs. Specifically, let arc $(i, j) \in A'_0$ have release date r_{ij}. The earliest time that it is known that (i, j) is released is at time r_{ij}. This forces decision makers to make decisions without complete information and adapt to the arrival of new information as it is learned.

Chapter 4

Complexity Results and Solution Methods

In this Chapter, we first present a summary of the complexity results of the considered INDS problems. We then discuss exact and heuristic solution methods to solve the problems.

Complexity Results

We now examine the complexity of these INDS problems, discussing that they are all at least *NP*-Hard. Table 4.1 provides an overview of the complexity results, showing whether each problem is ordinarily or strongly *NP*-Hard along with a potential hardness of approximation classification. Note, by showing the problems without release dates and flexible release are *NP*-Hard, we can further conclude that the problems with these extensions are also *NP*-Hard. For the full theoretical analysis, including proofs demonstrating these complexity results, please see Nurre and Sharkey [63].

Table 4.1: Summary of the *NP*-Hard results for the twelve INDS problems considered.

Problem	*NP*-Hard		
	Ordinarily	Strongly	Strongly No Approximation within $\ln\lvert N\rvert$
$Pm\|MF\|Cumulative$		✓	
$Pm\|MF\|Threshold$			✓
$Pm\|MCF\|Cumulative$		✓	
$Pm\|MCF\|Threshold$			✓
$Pm\|SP\|Cumulative$		Elgindy et al. [22]	
$Pm\|SP\|Threshold$	✓		
$Pm\|\sum SP\|Cumulative$		Elgindy et al. [22]	
$Pm\|\sum SP\|Threshold$			✓
$Pm\|\min\max SP\|Cumulative$		Elgindy et al. [22]	
$Pm\|\min\max SP\|Threshold$			✓
$Pm\|MST\|Cumulative$	✓		
$Pm\|MST\|Threshold$	✓		

Dispatching Rules

Due to the hardness of the problems, even for a single machine, we focus on both heuristic and exact solution methods. The first we discuss is a heuristic dispatching rule algorithm framework. We discuss the framework idea, for a full set of details please see Nurre and Sharkey [63].

Dispatching rules are a common heuristic approach for scheduling problems (see Pinedo [65]) that selects the next task to be processed by an available machine by examining the characteristics of the set of unprocessed tasks. These rules can be applied in real-time and often provide near-optimal solutions to scheduling problems. Therefore, the focus of this section is on a framework for creating dispatching rules for INDS problems that can be easily adapted for different performance metrics and problem extensions.

The success of a dispatching rule for a scheduling problem often relies on its ability to approximate the impact of completing a task on the objective of the problem. Dispatching rules that focus on scheduling the next task are often successful for traditional scheduling problems since once a task is processed by a machine, it impacts the objective function and then essentially leaves the scheduling system. INDS problems are unique in the field of scheduling since once an arc is processed and becomes operational, it interacts with other operational arcs in the network to improve its performance. For example, a newly operational arc can interact with other operational arcs in the network to allow additional flow to reach a sink node from a source node. In other words, in the problem $Pm|MF|Cumulative$, the newly operational arc can help form an *augmenting path* with other operational arcs in the network to deliver more flow through the network. These interactions (e.g., opening up new augmenting paths in a maximum flow network) will ultimately improve the performance of the network and, therefore, our dispatching rule framework will focus on understanding these interactions. In particular, our framework will focus on selecting the next *set of arcs* to be processed by the machines based on how they interact with each other and other arcs already in the operational network. For example, the dispatching rules for $Pm|MF|Cumulative$ and $Pm|MF|Threshold$ will select sets of arcs that build new augmenting paths in the network. These selected arcs will then be scheduled on the machines prior to determining the next set of arcs.

The general dispatching rule framework is presented in Algorithm 1 and can be applied to any INDS problem, including ones that are not considered in this chapter (e.g., the network performance is measured through solving a multi-commodity flow problem or a problem has a different machine scheduling environment). The inputs into this framework are: (i) the network performance metric, (ii) the objective function, (iii) the network, and (iv) the scheduling environment. The dispatching rule then alternates between a selection routine that selects the next set of arcs to be scheduled and the scheduling routine that schedules the set of selected arcs, until a stopping criteria is met.

We now discuss the selection and scheduling routines in more detail. The selection routine depends on how we measure the performance of the network and how we measure the impact of selected arcs on the machines. For the INDS problems considered in this chapter, we discuss that these considerations can be captured by answering four questions associ-

[1] for a maximization problem, $P(G_T) \leq P$ for a minimization problem

Algorithm 1 Integrated Network Design and Scheduling Problem Dispatching Rule Framework.

Input: Performance metric and objective function
Input: Network, G_0 with associated sets of operational arcs A_0 and non-operational arcs A'_0
Input: Resources, m parallel identical machines and objective function stopping criteria (for example, time horizon T or desired performance value P)
Set time t to 0
Calculate current performance of network and set to $P(G_t)$
while objective function stopping criteria not met (for example, $t < T$ or $(P(G_t) \geq P)^1$) **do**
    Select set of arcs $\bar{A}_t \subseteq A'_t$ to become operational = selection routine with input: G_t, performance metric
    Updated resources (time t) and network G_t = scheduling routine with input: $\bar{A}_t$
    Calculate current performance of the network $P(G_t)$
end while
Calculate and return objective function value

ated with the performance metric. The scheduling routine is customizable too; we present a scheduling routine that attempts to minimize the 'makespan' of completing the selected arcs. The remainder of this section is then dedicated to discussing how we customize the selection routine to each network performance metric.

Selection From a greedy perspective, a set of arcs that greatly improve the performance of the network while requiring little processing on the set of machines would be ideal to be selected. In other words, we are interested in selecting (for a maximization problem) the set of arcs that optimizes

$$\max_{\bar{A}_t \subseteq A'_t} \frac{\left|P(N_t, A_t \cup \bar{A}_t, A'_t \backslash \bar{A}_t) - P(G_t)\right|}{R(\bar{A}_t)} \tag{4.1}$$

where $R(\bar{A}_t)$ measures the machine (resource) requirements to process arcs in $\bar{A}_t$. For example, $R(\bar{A}_t)$ could measure the makespan of completing this set of arcs on the machines or, for parallel identical machines, could simply measure the total processing time of the arcs. For a set of arcs, $\bar{A}_t$, the ratio from (4.1) essentially provides the network performance improvement per-unit resource consumption. The problem that determines the set of arcs that maximizes (4.1) is a combinatorial optimization problem with a fractional objective function and can, therefore, be quite difficult to solve to optimality. For our proposed selection routines, we will solve an approximation of (4.1) where we estimate the improvement in the network as measured through the optimality conditions of the network performance. The machine (resource) requirements (i.e., the denominator in (4.1)) will be measured as the

total processing time of the arcs in the set, since this provides a good approximation of the makespan requirements for this set.

From a scheduling perspective, selecting the set of arcs that maximizes, or nearly maximizes, (4.1) is a generalization of the weighted shortest processing time (WSPT) rule. The WSPT rule calculates a priority ranking for each task j based on its weight w_j and processing time p_j. The task that is selected for processing next is the one that maximizes its ratio of weight to processing time, $\frac{w_j}{p_j}$. This rule is often applied to a parallel machine environment where we seek to minimize the total weighted completion time. To provide this generalization of the WSPT rule for each performance metric, we will focus on a series of questions: (i) What are the 'tasks'?, (ii) What are the weights of 'tasks'?, (iii) What are the processing times of 'tasks'?, and (iv) How do we find the 'task' that maximizes (minimizes) $\frac{w_j}{p_j}$? The answers to these questions will be based on the network optimality conditions of the performance metric, e.g., the augmenting path optimality conditions will help answer these questions for the maximum flow objective.

Scheduling Once the selection routine selects a set of arcs to be scheduled and become operational, they are fed into the scheduling routine. The scheduling routine determines how the machines in the problem will process this set of selected arcs. This routine does not depend on the specific INDS problem and should, typically, focus on scheduling the selected arcs so they are completed as quickly as possible. For our INDS problems, we will implement the longest processing time (LPT) rule to schedule the selected arcs. The arcs within the selected set are ordered from longest to shortest processing time and added to a queue. When a machine becomes available, the next arc in the queue is assigned and time is allocated for its processing. The LPT rule was chosen because it gives a good approximation when we want to complete the selected arcs as quickly as possible (see Pinedo [65]).

Performance Metric Specific Details As we have described the dispatching rule framework can be easily altered by answering a series of four questions. Table 4.2 presents the answers to these questions. For detailed explanations please see Nurre and Sharkey [63].

Integer Programming Formulations

The integer programming (IP) formulation of any INDS problem includes network constraints, scheduling constraints, and constraints that link these two sets of decisions. Many of the constraints are similar for the formulations of the different performance metrics. Table 4.3 displays the list of all decision variables, their corresponding definitions, and the specific performance metrics and objectives that utilize the decision variables. For the full integer programming formulations of the four performance metrics with the *Cumulative* and *Threshold* objectives please see Nurre and Sharkey [63]. For the integer programming formulation

Table 4.2: Summary of the answers needed to customize the selection routine by performance metric.

Customized Selection Routine for Maximum Flow INDS Problems
(i) Tasks = augmenting paths, P
(ii) Weights = minimum residual capacity of path, $\min_{(i,j)\in P} \upsilon_{ij}$
(iii) Processing time = $\sum_{(i,j)\in P\cap A'_t} p_{ij}$
(iv) Task maximizes ratio $\frac{w_j}{p_j}$ = Iterative algorithm using shortest path problems

Customized Selection Routine for Minimum Cost Flow and Shortest Path INDS Problems
(i) Tasks = Negative cycles, C
(ii) Weights = min residual capacity of cycle times cost, $\min_{(i,j)\in C} \upsilon_{ij} * \sum_{(i,j)\in C} c_{ij}$
(iii) Processing time = $\sum_{(i,j)\in C\cap A'_t} p_{ij}$
(iv) Task minimizes ratio $\frac{w_j}{p_j}$ = Iterative algorithm using minimum cost to time problems

Customized Selection Routine for Minimum Spanning Tree INDS Problems
(i) Tasks = Uninstalled arcs, (i,j)
(ii) Weights = $c_{ij} - \max_{(k,l)\in P_{ij}} c_{kl}$
(iii) Processing time = p_{ij}
(iv) Task minimizes ratio $\frac{w_j}{p_j}$ = linear over all non-operational arcs, $O(\lvert N\rvert)$ searches

with prioritized demand nodes and additional valid inequalities that strengthen the formulation please see Nurre et al. [62]. The inclusion of release dates, flexible release dates, and multi-function machines involves making minor adjustments to these formulations. Note, that the integer programming formulations are exact methods for the offline versions of the problem. There is no known method to find the exact optimal solution to an online optimization problem, as information arrives throughout the horizon of the problem. Therefore, the dispatching rule is the only solution method for online INDS problems. However, we use the offline version of the problem where all information is known to benchmark the solution quality.

Table 4.3: Decision variables used in the INDS IP formulations.

Decision Variable	Definition	Problems Used
x_{ij}, x_{ijt}	Continuous, flow on arc (i,j) (at time t)	MF, MCF, SP
v,v_t	Continuous, maximum flow (at time t)	MF
x^{ℓ}_{ij}, x^{ℓ}_{ijt}	Continuous, flow on arc (i,j) for commodity ℓ (at time t)	MST
ω_{ij},ω_{ijt}	Binary, equals 1 if arc (i,j) is in the MST (at time t)	MST
β_{ijt}	Binary, equals 1 if arc (i,j) is operational at time t	All *Cumulative*
$\alpha_{\mu ijt}$	Binary, equals 1 if arc (i,j) is completed by machine μ at time t	All *Cumulative*
$\bar{T}$	Integer, completion time of last processed task	All *Threshold*
$z_{\mu ij}$	Binary, equals 1 if arc (i,j) is assigned to machine μ	All *Threshold*

Chapter 5

Computational Results

The focus of this Chapter is on computational results testing the performance of the dispatching rules and integer programming formulations on realistic data sets. Many experiments have been conducted. For results focusing on the prioritized weighted maximum flow objective using the infrastructure systems of coastal New Hanover County, North Carolina and lower Manhattan, New York, please see [62]. Results demonstrating the robustness of the dispatching rule for a wide variety of performance metrics and objective functions using the realistic infrastructure systems of lower Manhattan, New York, we refer the reader to [63]. We now present the results of further computational experiments for INDS problems with release dates, flexible release dates, and multi-function machines in an offline and online optimization environment.

We now present the case studies testing the performance of the dispatching rules and empirically making observations based on the results. These case studies focus on the application of infrastructure restoration. We utilize the power network of CustomizabLe ARtificial Community (CLARC) County [50], which was created to mimic a power network in a hurricane prone coastal community with a population of approximately 500,000 (see Figure 5.1). Through the application of INDS problems under the different extensions, we create restoration plans for various degrees of network component damage, random release dates, processing times, work group environments, and levels of release date information.

The power network of CLARC County has $|N| = 838$ nodes and $|A_0 \cup A_0'| = 849$ arcs. Fifteen random damage instances are created that damage $10\%, 20\%$, and 30% of both the transmission network and distribution network for a total of 45 damage instances. Table 5.1 maps this network damage percentage to the impact on the reduction in the maximum flow in the network, showing that even a small percentage of damaged components can result in a very dramatic impact on the services provided by the network (flow). Historical data from NYISO [61] shows that Hurricane Sandy in October 2012 caused a decrease of approximately 70% in the peak power load for Long Island and approximately 30% for Manhattan. See Figure 5.2 for the impact of Sandy on the power load curves. Our current network damage percentages of $10\%, 20\%$, and 30% result in disruption to power flow provided comparable to the impacts seen due to Hurricane Sandy.

For each damage instance, each non-operational arc is assigned a random release date in the range $[0, T]$, where $T = 40$ is the time horizon of the problem corresponding to 10 days, where each time period is 6 hours. The processing time to make an arc operational is

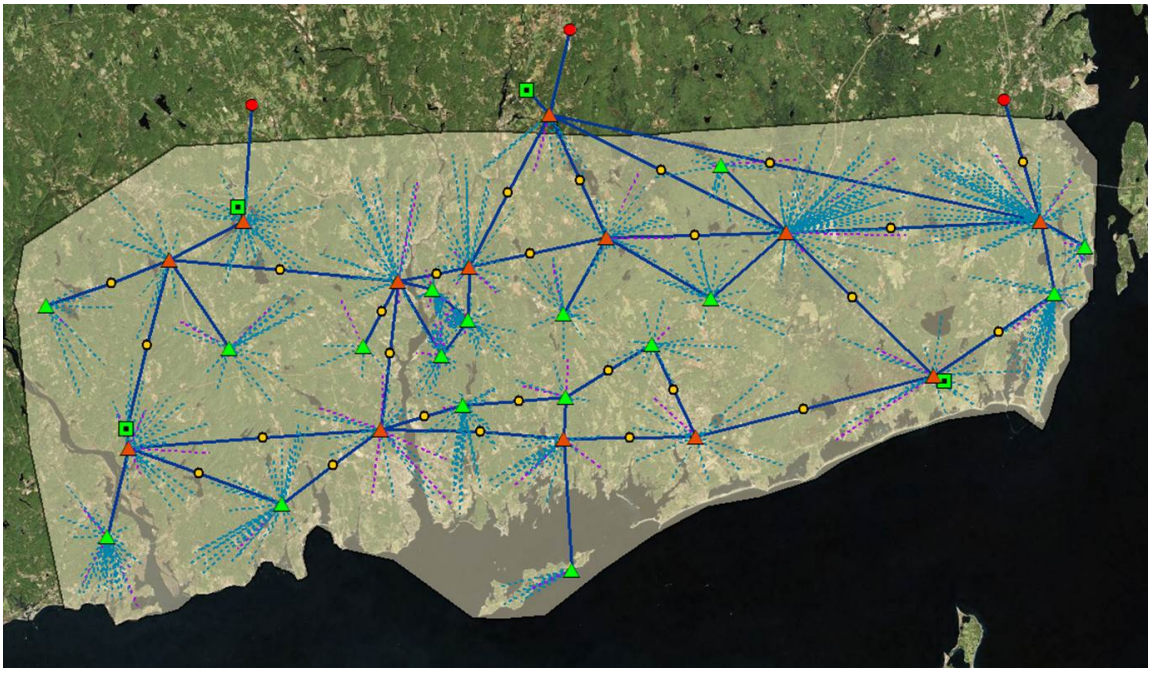


Figure 5.1: Visualization of the CLARC County power network with transmission and distribution components.

Table 5.1: Percentage reduction in the maximum flow at the start of the restoration process based on network component damage percentage in the CLARC County power network.

Damage	Mean	Min	Max
10%	23.97%	11.04%	44.39%
20%	48.11%	23.10%	90.26%
30%	65.60%	51.57%	83.71%

assumed to be fixed and constant over time. The processing time, $\bar{p}_{ij}$, to move the release date earlier in time for a non-operational arc should be based on the effects of the event because many different situations could be contributing to the inability to work on an arc. We set $\bar{p}_{ij}$ to be a function of the release date, specifically $\bar{p}_{ij} = \lceil \delta r_{ij} \rceil$, where $\delta = 0.1, 0.2$, and 0.3, are used to capture the different potential effects.

The offline dispatching rule, online dispatching rule, and integer program solved using CPLEX 12.6 are run for each test instance. The metrics of information captured with each run include: objective function value (upper and lower bounds for CPLEX, if applicable), elapsed computational time, and utilization of each machine (time working/time horizon). A time limit of 900 seconds is set for CPLEX. If an optimal solution is not found in this time limit, a feasible lower bound and the current best upper bound is captured. An optimality gap is calculated for all three solution methods by calculating the percentage difference between the best feasible solution value and the lowest upper bound on the optimal solution

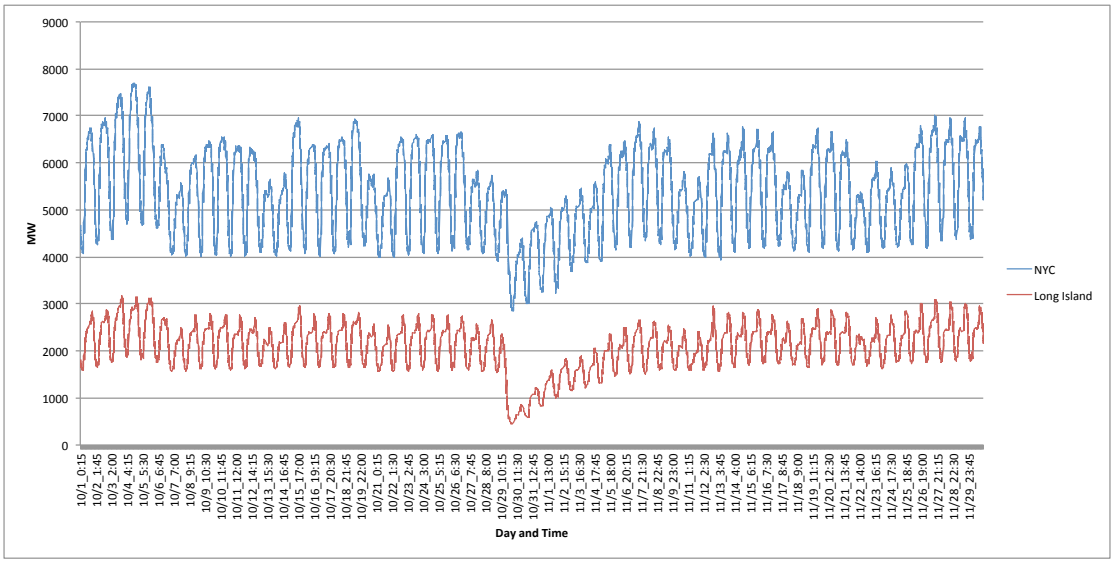


Figure 5.2: The impact of Hurricane Sandy on the power load curves for New York City and Long Island areas.

(which equals the optimal solution, if found). We first examine the performance of the different solution methods measured through optimality gaps and computational time and then analyze machine composition via measures of the objective function value and utilization.

Performance analysis through Optimality Gaps and Computational Time

In this section, we capture the performance of the offline dispatching rules, online dispatching rules, and integer programming formulations solved using CPLEX 12.6 as measured through optimality gaps and elapsed computational time needed to arrive at the solution. An empirical competitive ratio is also captured and is calculated as the maximum optimality gap for the online dispatching rules over the 15 random tests instances. In the following tables and figures, unless otherwise noted we present the average optimality gap and computational time over the 15 random instances. We do note that for each instance the specific components that are damaged and their respective release dates do impact the performance measures. Figure 5.3 shows the average optimality gap for the offline and online dispatching rules by damage percentage and random instance for INDS problems with flexible release dates and multi-function machines. It is apparent that the dispatching rules do not perform as well for high damage percentages, however the average optimality gap is relatively small. Variation is noticeable across the instances, demonstrating the impact of the random damage and release dates. Figure 5.4 presents the average computational time for CPLEX by damage and instance (the two dispatching rules are excluded as they are consistently solved in seconds for each damage and instance scenario). You will notice that some of the instances that led to larger optimality gaps in Figure 5.3 correspond to larger computational

times in Figure 5.4, specifically instances 3 and 10 for 30% damage. This reinforces that the specific components that are damaged and their respective release dates impact the solution and computational time for INDS problems.

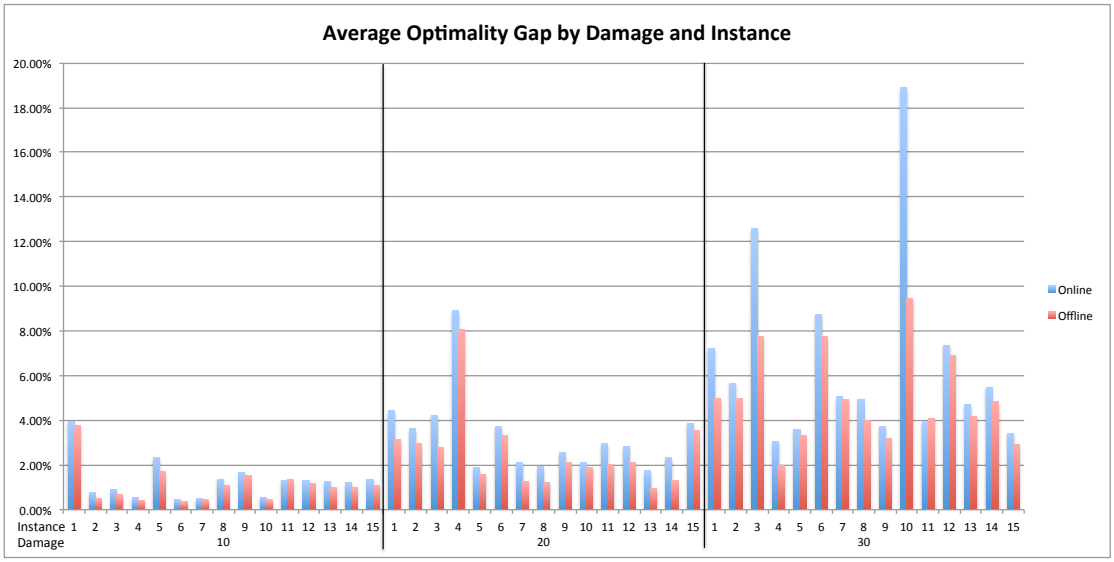


Figure 5.3: Average optimality gap by damage and instance.

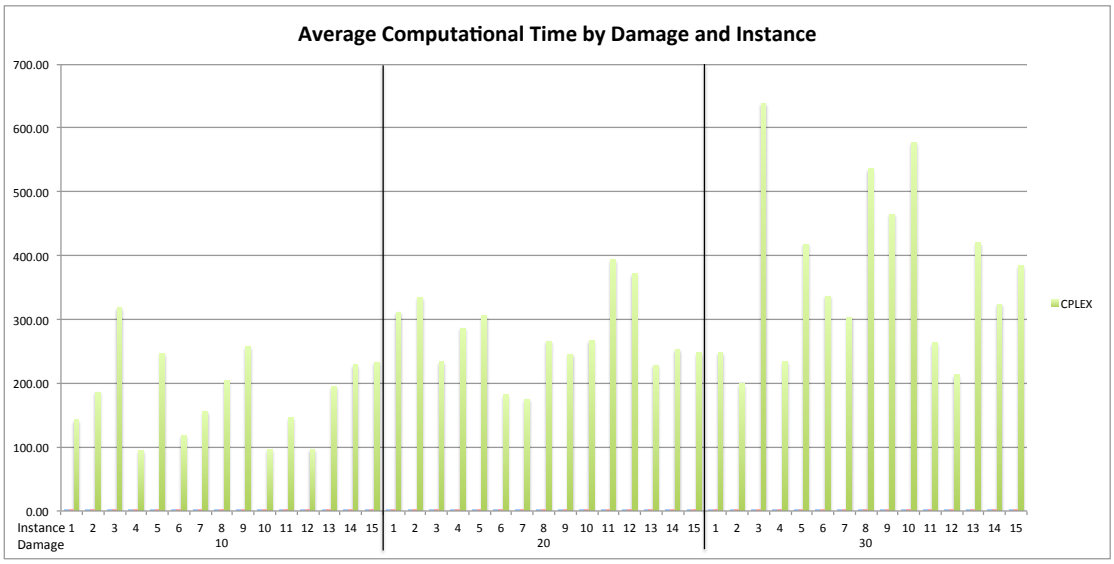


Figure 5.4: Average computation time by damage and instance.

Release Dates Performance We first test INDS problems with release dates for $m = 1, \ldots, 5$ and $10\%, 20\%$, and 30% network damage. Figure 5.5 and Table 5.2 present the results for $Pm|MF, r_{ij}|Cumulative$. For the Gap (%) and Time (s) columns within the table, the values displayed are the average over the 15 random test instances. First, we note that all three solution methods can solve the problem in a matter of seconds. Second, we note that both of the dispatching rules have small optimality gaps, signifying that near optimal restoration plans are found. One anomaly here is that the offline dispatching rule is performing worse than the online dispatching rule, even when the offline method knows the complete set of release dates at the beginning. Upon examination of the solution, the offline solution is selecting highly important components that provide a significant amount of flow at the cost of waiting for their release dates. Our offline dispatching rule currently does not fill in the gaps of any idle time on machines, and should be an avenue for future work. Lastly, we note that even with the unbounded theoretical result for the online competitive ratio, empirically we are seeing acceptable competitive ratios with the maximum begin 16.31% and many others below 10%.

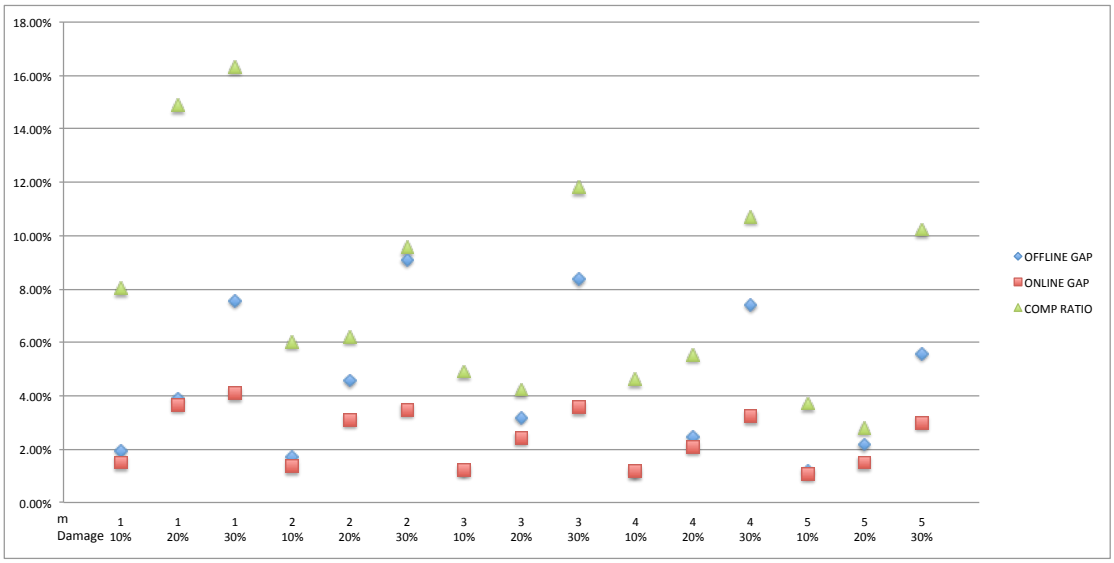


Figure 5.5: Average optimality gap for $\boldsymbol{Pm|MF, r_{ij}|Cumulative}$ by solution method, machine, and damage.

Table 5.2: $\boldsymbol{Pm|MF, r_{ij}|Cumulative}$ computational results comparing the performance of the offline and online dispatching rules and CPLEX 12.6.

					Offline		Online		CPLEX		Empirical
m	L	Mu	Damage	δ	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Comp. Ratio
1	0	0	10%	-	1.91%	0.51	1.48%	0.41	0.00%	1.17	8.02%
			20%	-	3.90%	0.53	3.64%	0.44	0.00%	2.38	14.88%
			30%	-	7.54%	0.57	4.09%	0.51	0.00%	6.05	16.31%
2	0	0	10%	-	1.68%	0.96	1.37%	0.83	0.00%	1.46	6.00%
			20%	-	4.56%	1.13	3.10%	0.82	0.00%	3.45	6.18%
			30%	-	9.06%	1.20	3.47%	1.00	0.00%	9.00	9.58%
3	0	0	10%	-	1.24%	1.42	1.20%	1.23	0.00%	1.56	4.89%
			20%	-	3.16%	1.71	2.42%	1.22	0.00%	4.30	4.23%
			30%	-	8.38%	1.95	3.54%	1.36	0.00%	14.42	11.83%
4	0	0	10%	-	1.19%	1.84	1.16%	1.62	0.00%	2.05	4.60%
			20%	-	2.46%	2.20	2.09%	1.75	0.00%	6.39	5.54%
			30%	-	7.40%	2.57	3.25%	1.77	0.00%	14.27	10.72%
5	0	0	10%	-	1.16%	2.27	1.07%	2.05	0.00%	2.41	3.72%
			20%	-	2.17%	2.83	1.48%	2.47	0.00%	6.32	2.80%
			30%	-	5.56%	3.21	2.99%	2.28	0.00%	10.53	10.21%

Flexible Release Dates Performance We now test INDS problems with flexible release dates, specifically $Opt - FF2|MF, flex - r_{ij}|Cumulative$ with different machine compositions, damage values, and δ values. Each combination of $m = 1, \ldots, 4$ traditional machines with $L = 1, \ldots, 4$ specialized machines is tested for each damage and δ value. The optimality gap, elapsed computational time, and empirical competitive ratio are captured by solution method. Figures 5.6, 5.7, 5.8, and 5.9 show the test results for $m = 1, 2, 3$, and 4, respectively and Tables 5.3, 5.4, and 5.5 show the test results for $\delta = 0.1, 0.2$, and 0.3, respectively. On average, both dispatching rules perform well with optimality gaps predominantly below 10%. A trend is apparent, with the dispatching rules performing better for smaller damage percentages. The unbounded nature of the competitive ratio is empirically demonstrated in Figure 5.6 with one competitive ratio value very high at 133.25%, but others very acceptable in the 10% area or less. Note that the average online gap is often very reasonable indicating that the online dispatching rule performs well even when benchmarked against the integer programming formulation with complete information. Some of the higher optimality gaps could also be attributed to the inability for CPLEX to find the optimal solution within the 900 second time limit. If an optimal solution is not found then the calculated optimality gap is an *overestimate* on the actual optimality gap. Further, we note that CPLEX struggled with the flexible release date instances, in particular when considering larger number of work groups. Both of the dispatching rules scale well when considering larger number of work groups, by consistently finding near optimal solutions in less than 2 seconds.

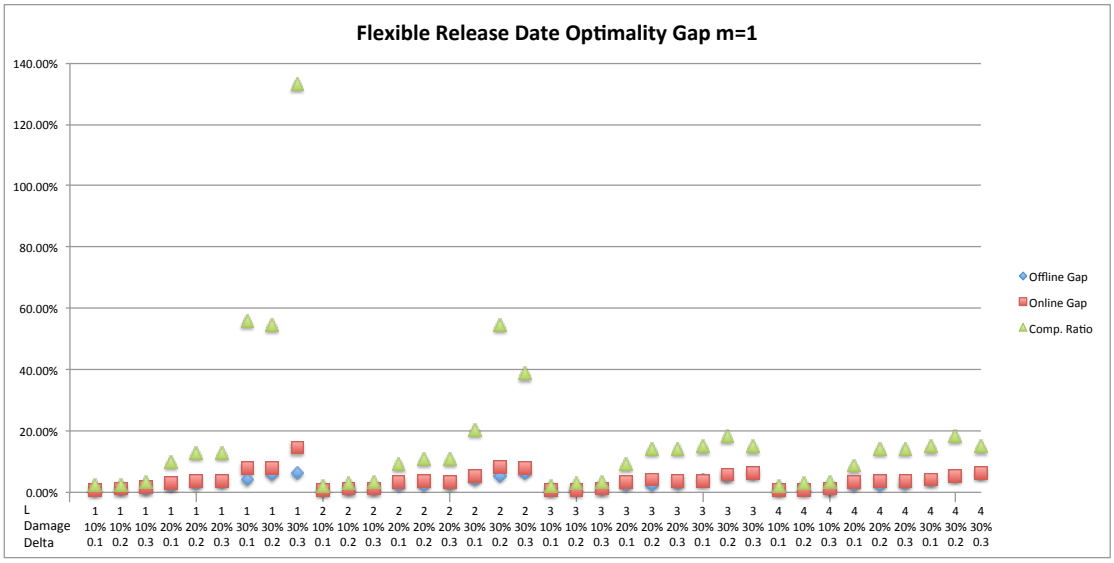


Figure 5.6: Average optimality gap for $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ with $\boldsymbol{m = 1}$ by solution method, machine, damage, and $\boldsymbol{\delta}$.

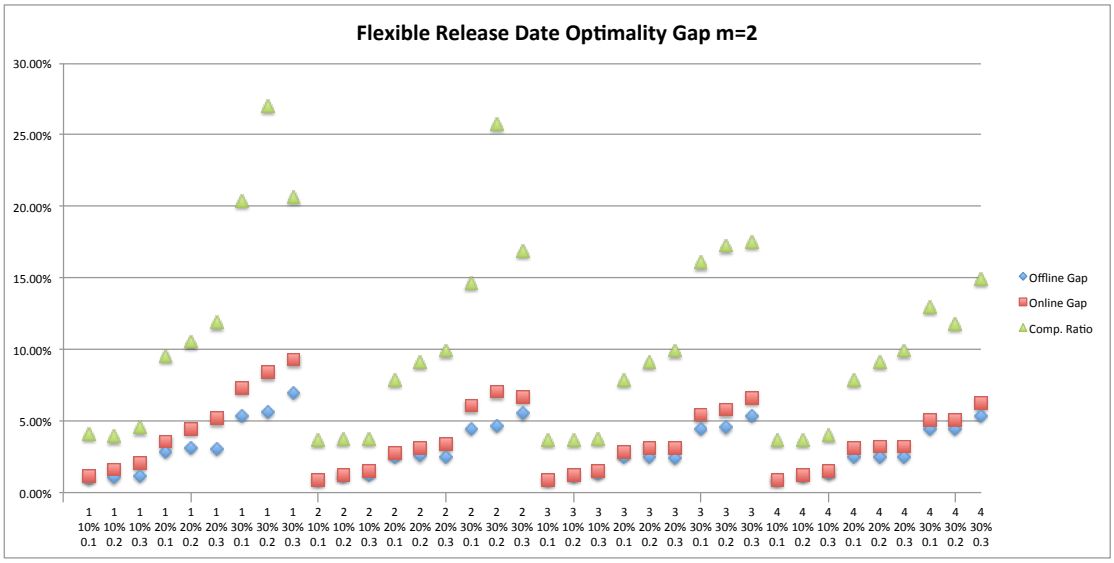


Figure 5.7: Average optimality gap for $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ with $\boldsymbol{m = 2}$ by solution method, machine, damage, and $\boldsymbol{\delta}$.

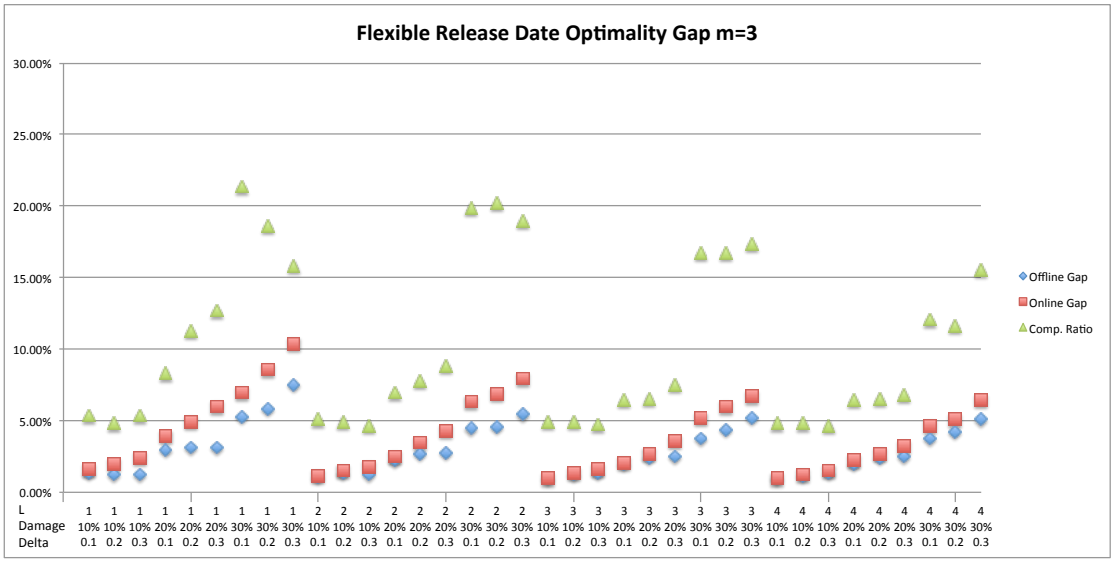


Figure 5.8: Average optimality gap for $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ with $\boldsymbol{m = 3}$ by solution method, machine, damage, and $\boldsymbol{\delta}$.

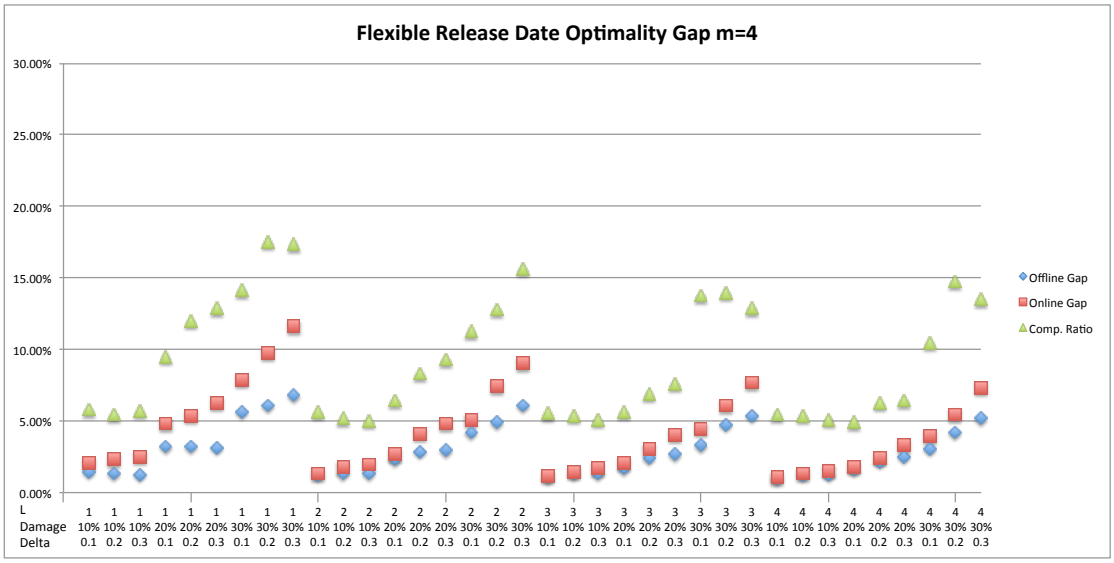


Figure 5.9: Average optimality gap for $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ with $\boldsymbol{m = 4}$ by solution method, machine, damage, and $\boldsymbol{\delta}$.

Table 5.3: $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ computational results comparing the performance of the offline and online dispatching rules and CPLEX 12.6 when $\boldsymbol{\delta = 0.1}$.

					Offline		Online		CPLEX		Empirical
m	L	Mu	Damage	δ	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Comp. Ratio
1	1	0	10%	0.1	0.86%	0.41	0.86%	0.39	0.00%	2.39	2.37%
			20%	0.1	2.59%	0.43	2.78%	0.41	0.00%	11.81	10.13%
			30%	0.1	4.34%	0.37	8.09%	0.36	0.00%	41.44	55.88%
	2	0	10%	0.1	0.87%	0.41	0.84%	0.40	0.00%	2.70	2.00%
			20%	0.1	2.60%	0.43	3.52%	0.40	0.00%	13.28	9.30%
			30%	0.1	4.35%	0.37	5.32%	0.39	0.00%	35.95	20.48%
	3	0	10%	0.1	0.87%	0.41	0.84%	0.40	0.00%	2.80	2.00%
			20%	0.1	2.60%	0.43	3.39%	0.41	0.00%	14.03	9.10%
			30%	0.1	4.35%	0.37	3.94%	0.41	0.00%	32.33	14.98%
	4	0	10%	0.1	0.87%	0.41	0.78%	0.40	0.00%	3.25	2.00%
			20%	0.1	2.60%	0.43	3.24%	0.41	0.00%	16.42	8.55%
			30%	0.1	4.35%	0.37	4.27%	0.40	0.00%	38.83	14.98%
2	1	0	10%	0.1	1.00%	0.82	1.16%	0.75	0.00%	35.66	4.07%
			20%	0.1	2.89%	0.89	3.60%	0.81	0.00%	66.67	9.53%
			30%	0.1	5.33%	0.90	7.36%	0.79	0.16%	261.83	20.35%
	2	0	10%	0.1	0.86%	0.85	0.94%	0.77	0.00%	4.58	3.64%
			20%	0.1	2.53%	0.89	2.74%	0.81	0.00%	34.03	7.84%
			30%	0.1	4.49%	0.93	6.09%	0.82	0.00%	125.54	14.66%
	3	0	10%	0.1	0.85%	0.85	0.93%	0.77	0.00%	5.00	3.64%
			20%	0.1	2.53%	0.89	2.88%	0.78	0.00%	35.04	7.84%
			30%	0.1	4.49%	0.93	5.48%	0.82	0.00%	124.77	16.12%
	4	0	10%	0.1	0.85%	0.85	0.92%	0.77	0.00%	4.98	3.64%
			20%	0.1	2.53%	0.89	3.08%	0.77	0.00%	35.30	7.83%
			30%	0.1	4.49%	0.93	5.13%	0.82	0.00%	150.28	12.93%
3	1	0	10%	0.1	1.32%	1.11	1.64%	1.00	0.06%	431.22	5.38%
			20%	0.1	2.93%	1.29	3.93%	1.08	0.16%	555.37	8.28%
			30%	0.1	5.24%	1.46	6.94%	1.28	0.39%	707.71	21.35%
	2	0	10%	0.1	1.05%	1.29	1.15%	1.12	0.00%	26.97	5.09%
			20%	0.1	2.20%	1.34	2.51%	1.16	0.02%	176.70	7.01%
			30%	0.1	4.50%	1.50	6.31%	1.25	0.02%	303.18	19.87%
	3	0	10%	0.1	0.95%	1.30	1.01%	1.14	0.00%	8.37	4.90%
			20%	0.1	1.93%	1.35	2.09%	1.16	0.00%	60.90	6.48%
			30%	0.1	3.76%	1.52	5.15%	1.27	0.00%	186.88	16.75%
	4	0	10%	0.1	0.95%	1.30	0.98%	1.15	0.00%	7.98	4.84%
			20%	0.1	1.93%	1.36	2.23%	1.17	0.00%	68.10	6.48%
			30%	0.1	3.76%	1.52	4.69%	1.29	0.00%	184.08	12.09%
4	1	0	10%	0.1	1.47%	1.39	2.03%	1.15	0.10%	738.85	5.78%
			20%	0.1	3.22%	1.68	4.81%	1.34	0.42%	886.35	9.50%
			30%	0.1	5.61%	1.88	7.84%	1.61	1.04%	900.00	14.15%
	2	0	10%	0.1	1.25%	1.71	1.37%	1.48	0.06%	420.77	5.66%
			20%	0.1	2.32%	1.91	2.72%	1.62	0.20%	521.83	6.46%
			30%	0.1	4.18%	2.00	5.08%	1.75	0.38%	698.74	11.30%
	3	0	10%	0.1	1.06%	1.77	1.12%	1.56	0.00%	15.86	5.54%
			20%	0.1	1.74%	1.96	2.06%	1.62	0.03%	170.35	5.65%
			30%	0.1	3.30%	2.07	4.51%	1.70	0.05%	427.15	13.81%
	4	0	10%	0.1	0.99%	1.79	1.04%	1.58	0.00%	15.17	5.48%
			20%	0.1	1.61%	1.98	1.83%	1.65	0.00%	95.69	4.96%
			30%	0.1	3.03%	2.08	3.92%	1.63	0.09%	375.26	10.43%

Table 5.4: $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ computational results comparing the performance of the offline and online dispatching rules and CPLEX 12.6 when $\boldsymbol{\delta = 0.2}$.

					Offline		Online		CPLEX		Empirical
m	L	Mu	Damage	δ	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Comp. Ratio
1	1	0	10%	0.2	1.05%	0.40	1.10%	0.38	0.00%	2.92	2.49%
			20%	0.2	2.95%	0.41	3.66%	0.38	0.00%	10.47	12.84%
			30%	0.2	5.65%	0.36	7.94%	0.36	0.00%	63.61	54.72%
	2	0	10%	0.2	1.07%	0.40	1.08%	0.38	0.00%	2.40	2.80%
			20%	0.2	2.67%	0.42	3.76%	0.37	0.00%	11.38	10.95%
			30%	0.2	5.49%	0.36	8.43%	0.35	0.00%	40.69	54.72%
	3	0	10%	0.2	1.07%	0.41	1.02%	0.39	0.00%	2.73	2.80%
			20%	0.2	2.67%	0.42	4.07%	0.37	0.00%	10.46	14.10%
			30%	0.2	5.50%	0.36	5.81%	0.37	0.00%	41.02	18.38%
	4	0	10%	0.2	1.07%	0.40	1.02%	0.39	0.00%	3.06	2.80%
			20%	0.2	2.67%	0.42	3.75%	0.39	0.00%	12.59	14.10%
			30%	0.2	5.50%	0.36	5.50%	0.40	0.00%	35.22	18.38%
2	1	0	10%	0.2	1.08%	0.73	1.62%	0.64	0.00%	77.86	3.96%
			20%	0.2	3.09%	0.85	4.45%	0.73	0.00%	156.22	10.52%
			30%	0.2	5.66%	0.88	8.38%	0.77	0.32%	308.10	27.05%
	2	0	10%	0.2	1.13%	0.82	1.26%	0.74	0.00%	37.42	3.73%
			20%	0.2	2.55%	0.87	3.09%	0.78	0.00%	122.53	9.08%
			30%	0.2	4.61%	0.90	7.04%	0.79	0.16%	288.86	25.75%
	3	0	10%	0.2	1.16%	0.83	1.27%	0.74	0.00%	4.95	3.66%
			20%	0.2	2.54%	0.88	3.16%	0.76	0.00%	34.87	9.08%
			30%	0.2	4.54%	0.91	5.83%	0.80	0.09%	222.09	17.26%
	4	0	10%	0.2	1.15%	0.85	1.25%	0.74	0.00%	4.85	3.66%
			20%	0.2	2.54%	0.88	3.21%	0.76	0.00%	32.30	9.08%
			30%	0.2	4.47%	0.91	5.10%	0.80	0.00%	161.28	11.83%
3	1	0	10%	0.2	1.28%	1.00	2.00%	0.85	0.01%	240.01	4.82%
			20%	0.2	3.12%	1.20	4.88%	1.01	0.05%	472.07	11.27%
			30%	0.2	5.81%	1.34	8.60%	1.20	0.58%	635.82	18.62%
	2	0	10%	0.2	1.33%	1.11	1.55%	0.96	0.09%	547.14	4.92%
			20%	0.2	2.65%	1.28	3.49%	1.10	0.17%	705.39	7.78%
			30%	0.2	4.58%	1.45	6.87%	1.24	0.57%	793.86	20.19%
	3	0	10%	0.2	1.25%	1.25	1.31%	1.08	0.04%	292.99	4.95%
			20%	0.2	2.42%	1.34	2.71%	1.15	0.11%	336.89	6.54%
			30%	0.2	4.41%	1.48	5.99%	1.21	0.20%	466.65	16.72%
	4	0	10%	0.2	1.18%	1.27	1.23%	1.11	0.00%	14.19	4.85%
			20%	0.2	2.43%	1.34	2.70%	1.13	0.01%	137.31	6.54%
			30%	0.2	4.23%	1.47	5.14%	1.24	0.05%	333.50	11.64%
4	1	0	10%	0.2	1.30%	1.26	2.34%	0.98	0.02%	305.74	5.48%
			20%	0.2	3.19%	1.53	5.35%	1.24	0.15%	714.10	12.01%
			30%	0.2	6.07%	1.74	9.77%	1.48	0.88%	855.33	17.50%
	2	0	10%	0.2	1.35%	1.44	1.74%	1.17	0.12%	766.95	5.22%
			20%	0.2	2.87%	1.66	4.10%	1.38	0.46%	887.75	8.36%
			30%	0.2	4.88%	1.87	7.38%	1.62	1.40%	900.00	12.80%
	3	0	10%	0.2	1.33%	1.57	1.47%	1.37	0.15%	734.92	5.37%
			20%	0.2	2.43%	1.82	2.99%	1.50	0.42%	829.68	6.88%
			30%	0.2	4.74%	1.95	6.11%	1.67	1.38%	858.38	13.93%
	4	0	10%	0.2	1.25%	1.70	1.33%	1.46	0.05%	446.97	5.41%
			20%	0.2	2.13%	1.88	2.43%	1.64	0.19%	456.99	6.22%
			30%	0.2	4.24%	2.03	5.45%	1.68	0.76%	715.73	14.72%

Table 5.5: $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ computational results comparing the performance of the offline and online dispatching rules and CPLEX 12.6 when $\boldsymbol{\delta = 0.3}$.

					Offline		Online		CPLEX		Empirical
m	L	Mu	Damage	δ	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Comp. Ratio
1	1	0	10%	0.3	1.18%	0.38	1.46%	0.34	0.00%	4.10	3.33%
			20%	0.3	3.37%	0.40	3.82%	0.37	0.00%	15.71	12.87%
			30%	0.3	6.18%	0.34	14.46%	0.35	0.00%	69.61	133.25%
	2	0	10%	0.3	1.25%	0.39	1.36%	0.37	0.00%	2.68	3.42%
			20%	0.3	2.70%	0.41	3.15%	0.38	0.00%	14.39	10.95%
			30%	0.3	6.17%	0.34	7.93%	0.34	0.00%	51.22	38.92%
	3	0	10%	0.3	1.25%	0.39	1.34%	0.37	0.00%	2.52	3.42%
			20%	0.3	2.73%	0.41	3.75%	0.35	0.00%	12.78	14.10%
			30%	0.3	6.21%	0.34	6.21%	0.36	0.00%	44.80	15.09%
	4	0	10%	0.3	1.25%	0.40	1.32%	0.38	0.00%	2.82	3.42%
			20%	0.3	2.73%	0.41	3.68%	0.36	0.00%	15.53	14.10%
			30%	0.3	6.21%	0.35	6.23%	0.36	0.00%	39.57	15.15%
2	1	0	10%	0.3	1.20%	0.67	2.01%	0.56	0.00%	34.46	4.57%
			20%	0.3	3.08%	0.79	5.16%	0.64	0.00%	100.19	11.92%
			30%	0.3	7.01%	0.79	9.33%	0.71	0.25%	244.35	20.70%
	2	0	10%	0.3	1.27%	0.76	1.54%	0.66	0.00%	100.44	3.76%
			20%	0.3	2.53%	0.86	3.41%	0.74	0.01%	179.34	9.93%
			30%	0.3	5.51%	0.86	6.75%	0.79	0.14%	361.18	16.91%
	3	0	10%	0.3	1.34%	0.80	1.50%	0.70	0.00%	22.43	3.76%
			20%	0.3	2.43%	0.86	3.15%	0.76	0.00%	167.22	9.97%
			30%	0.3	5.41%	0.88	6.62%	0.77	0.20%	280.08	17.55%
	4	0	10%	0.3	1.36%	0.81	1.51%	0.73	0.00%	8.15	3.99%
			20%	0.3	2.49%	0.87	3.24%	0.75	0.00%	56.25	9.97%
			30%	0.3	5.34%	0.87	6.23%	0.76	0.06%	248.55	14.95%
3	1	0	10%	0.3	1.26%	0.94	2.38%	0.76	0.00%	50.19	5.40%
			20%	0.3	3.13%	1.12	5.97%	0.90	0.00%	266.67	12.70%
			30%	0.3	7.48%	1.29	10.37%	1.06	0.42%	480.76	15.87%
	2	0	10%	0.3	1.27%	1.05	1.79%	0.84	0.01%	266.05	4.67%
			20%	0.3	2.74%	1.22	4.26%	0.99	0.01%	570.91	8.83%
			30%	0.3	5.42%	1.40	7.98%	1.17	0.45%	633.00	18.94%
	3	0	10%	0.3	1.32%	1.09	1.60%	0.92	0.03%	494.64	4.72%
			20%	0.3	2.52%	1.24	3.53%	1.04	0.21%	630.82	7.48%
			30%	0.3	5.17%	1.42	6.66%	1.20	0.60%	772.63	17.32%
	4	0	10%	0.3	1.37%	1.17	1.50%	1.02	0.07%	409.94	4.68%
			20%	0.3	2.47%	1.29	3.19%	1.11	0.16%	538.60	6.79%
			30%	0.3	5.10%	1.43	6.47%	1.21	0.43%	669.69	15.57%
4	1	0	10%	0.3	1.26%	1.18	2.49%	0.99	0.00%	148.18	5.69%
			20%	0.3	3.08%	1.45	6.25%	1.20	0.01%	377.18	12.85%
			30%	0.3	6.78%	1.70	11.63%	1.36	0.66%	796.03	17.33%
	2	0	10%	0.3	1.35%	1.29	1.96%	0.98	0.00%	319.46	5.02%
			20%	0.3	2.91%	1.53	4.80%	1.21	0.17%	714.94	9.26%
			30%	0.3	6.05%	1.73	9.07%	1.41	1.15%	862.49	15.69%
	3	0	10%	0.3	1.34%	1.39	1.67%	1.12	0.08%	743.43	5.11%
			20%	0.3	2.66%	1.64	3.99%	1.26	0.37%	893.53	7.58%
			30%	0.3	5.38%	1.83	7.70%	1.54	1.41%	893.00	12.87%
	4	0	10%	0.3	1.29%	1.50	1.49%	1.24	0.07%	716.23	5.14%
			20%	0.3	2.49%	1.74	3.28%	1.42	0.52%	850.18	6.43%
			30%	0.3	5.18%	1.94	7.31%	1.56	1.40%	852.96	13.49%

Multi-Function Machines Performance We now test INDS problems with multi-function machines, specifically $MFM - Pm|MF, flex - r_{ij}|Cumulative$ for $Mu = 1, \ldots, 5$ and each damage and δ value. The results of these tests are presented in Figure 5.10 and Table 5.6. The level of damage and δ do not appear to have a significant impact on the performance of the dispatching rules, as we see only very slight upward trends as the damage and δ values increase. The dispatching rule framework again demonstrates its robustness by consistently arriving at high quality solutions in a matter of seconds. CPLEX continues to struggle to find the optimal solution within the time limit, specifically when considering a larger number of multi-function work groups. We note that the empirical competitive ratio is again very high when considering 1 traditional work group and 30% damage. The competitive ratio for the corresponding tests for flexible release dates, also had very high empirical competitive ratios. However, on average the online dispatching rule is performing only marginally worse than the offline dispatching rule, which is promising, given the motivating application and the reality of the lack of complete information. We do notice that the empirical competitive ratio decreases, on average, as more multi-function machines are considered. We believe that as more machines are considered the sequencing decisions become easier, as 5 selected network components can be processed 'first' with 5 multi-function machines. Essentially, with more machine you have a greater chance of getting things 'right', particularly in the online environment.

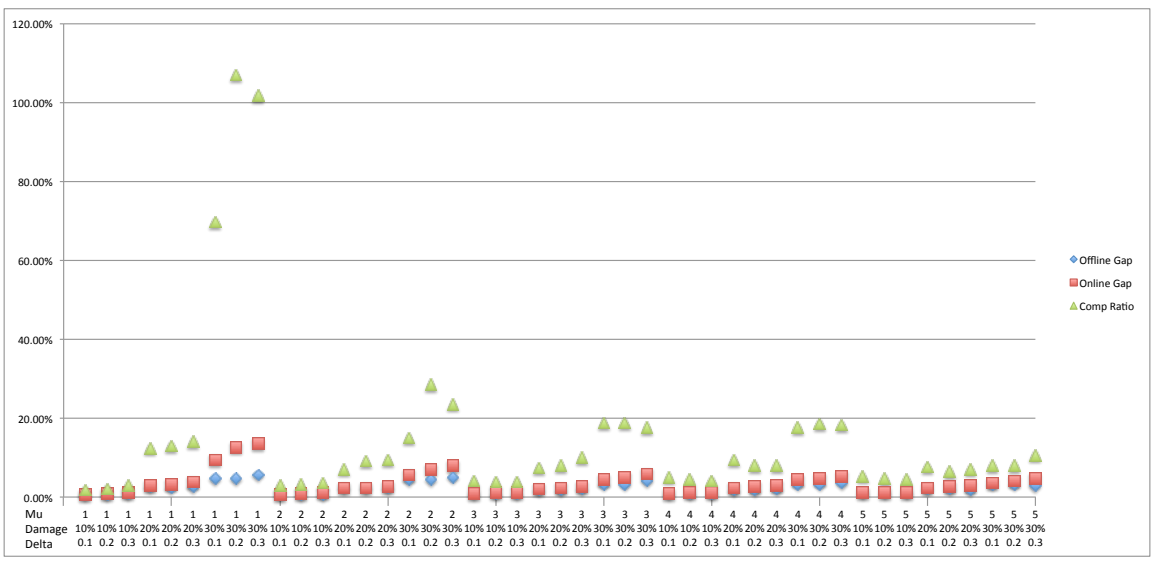


Figure 5.10: Average optimality gap for $\boldsymbol{MFM - Pm|MF, flex - r_{ij}|Cumulative}$ by solution method, machine, damage, and $\boldsymbol{\delta}$.

Machine Composition and Utilization Analysis

In this section, we analyze how the composition of machines impacts the amount of met demand (objective function value) and the utilization of the machines. The results demonstrate the benefit of specialized work groups moving release dates earlier in time, the

Table 5.6: $\boldsymbol{MFM-Pm|MF, flex-r_{ij}|Cumulative}$ computational results comparing the performance of the offline and online dispatching rules and CPLEX 12.6.

					Offline		Online		CPLEX		Empirical
m	L	Mu	Damage	δ	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Comp. Ratio
0	0	1	10%	0.1	0.58%	0.38	0.61%	0.36	0.00%	3.89	1.77%
				0.2	0.67%	0.37	0.92%	0.34	0.00%	3.41	2.17%
				0.3	0.93%	0.36	1.03%	0.32	0.00%	2.53	2.94%
			20%	0.1	2.53%	0.40	2.98%	0.36	0.00%	15.00	12.20%
				0.2	2.45%	0.38	3.27%	0.32	0.00%	13.56	13.03%
				0.3	2.65%	0.37	3.67%	0.27	0.00%	9.74	14.25%
			30%	0.1	4.67%	0.36	9.42%	0.34	0.00%	26.25	69.56%
				0.2	4.72%	0.35	12.67%	0.34	0.00%	40.48	107.18%
				0.3	5.52%	0.32	13.57%	0.32	0.00%	45.45	101.76%
0	0	2	10%	0.1	0.61%	0.71	0.71%	0.64	0.00%	36.81	3.04%
				0.2	0.69%	0.65	0.90%	0.56	0.00%	15.14	3.34%
				0.3	0.71%	0.62	1.17%	0.49	0.00%	10.38	3.54%
			20%	0.1	2.42%	0.80	2.28%	0.77	0.00%	72.79	7.04%
				0.2	2.26%	0.78	2.46%	0.71	0.00%	45.61	9.08%
				0.3	2.34%	0.75	2.75%	0.66	0.00%	33.59	9.41%
			30%	0.1	4.27%	0.85	5.45%	0.84	0.20%	226.50	14.98%
				0.2	4.44%	0.82	7.18%	0.74	0.22%	172.57	28.49%
				0.3	4.98%	0.79	8.02%	0.67	0.26%	132.85	23.48%
0	0	3	10%	0.1	0.79%	1.03	0.87%	0.98	0.00%	248.76	4.20%
				0.2	0.83%	0.95	1.03%	0.75	0.00%	132.79	3.83%
				0.3	0.83%	0.89	1.19%	0.61	0.00%	58.22	3.85%
			20%	0.1	1.83%	1.16	2.12%	1.03	0.03%	469.04	7.24%
				0.2	1.79%	1.10	2.25%	0.93	0.01%	286.41	8.01%
				0.3	2.04%	1.09	2.66%	0.83	0.00%	138.48	9.85%
			30%	0.1	3.16%	1.38	4.43%	1.25	0.45%	546.59	18.77%
				0.2	3.36%	1.31	4.88%	1.16	0.43%	402.85	18.88%
				0.3	4.19%	1.28	5.92%	1.02	0.45%	306.98	17.54%
0	0	4	10%	0.1	0.91%	1.40	0.97%	1.22	0.03%	576.93	4.95%
				0.2	0.87%	1.25	1.05%	0.93	0.02%	413.28	4.35%
				0.3	0.92%	1.21	1.27%	0.77	0.01%	264.83	4.09%
			20%	0.1	1.90%	1.56	2.46%	1.32	0.36%	863.93	9.49%
				0.2	1.82%	1.45	2.59%	1.09	0.11%	618.97	8.03%
				0.3	1.91%	1.40	2.97%	0.88	0.00%	377.91	7.87%
			30%	0.1	3.36%	1.78	4.29%	1.53	1.16%	850.93	17.67%
				0.2	3.21%	1.71	4.55%	1.43	0.77%	658.00	18.48%
				0.3	3.63%	1.68	5.23%	1.35	0.54%	584.63	18.28%
0	0	5	10%	0.1	1.03%	1.83	1.08%	1.56	0.08%	707.46	5.38%
				0.2	1.03%	1.61	1.12%	1.16	0.10%	701.03	4.80%
				0.3	1.07%	1.44	1.30%	0.83	0.06%	600.93	4.54%
			20%	0.1	1.95%	2.07	2.38%	1.69	0.64%	900.00	7.74%
				0.2	1.95%	1.82	2.62%	1.25	0.50%	877.11	6.50%
				0.3	1.77%	1.72	3.07%	1.05	0.20%	748.57	7.00%
			30%	0.1	3.30%	2.19	3.58%	1.77	1.67%	900.00	7.93%
				0.2	3.16%	2.05	4.12%	1.52	1.17%	889.87	7.90%
				0.3	2.94%	2.09	4.56%	1.37	0.54%	756.60	10.59%

greater benefit of multi-function machines, and a phenomenon of diminishing return when considering more specialized work groups.

Machine Composition We first examine the improvement in the average objective function value for different compositions of machines. Figure 5.11 displays every combination of 2 machines: (i) $m = 2, L = 0, Mu = 0$, (ii) $m = 1, L = 1, Mu = 0$, and (iii) $m = 0, L = 0, Mu = 2$ for each of the damage values. The improvement is then calculated as the percentage improvement against $m = 2, L = 0, Mu = 0$, where no specialized or multi-function machines are considered. Figures 5.12, 5.13, and 5.14 display the same for compositions of 3, 4, and 5 machines, respectively where the improvement is calculated against the case with only release dates i.e., $L = 0$ and $Mu = 0$. It is clear that the multi-function machines perform best in all cases, which indicates the advantage of skilled work groups that can perform many tasks. This is expected as the solution to each case with release dates and flexible release dates is feasible for the multi-function machine setting. Negative improvement values do appear often times when the number of specialized machines outnumber the traditional machines. This decrease is more apparent with higher damage percentages and a smaller number of total machines. When the number of traditional machines equals or exceeds the number of specialized machines, we do see improvement due to flexible release dates. This type of analysis touches upon the need for skilled labor, but also coordination of efforts after an extreme event. We assume for flexible release dates and multi-function machines problems that a central decision maker can plan and allocate their resources. This is not always the case, as many tasks have jurisdiction dedicated to specific entities and would prevent this coordination. We demonstrate that when efforts are coordinated and the specialized work groups and the traditional work groups work in unison (in either the flexible release dates or multi-function machine settings), that the services proved improve when compared to no coordination ($L = 0$ and $Mu = 0$).

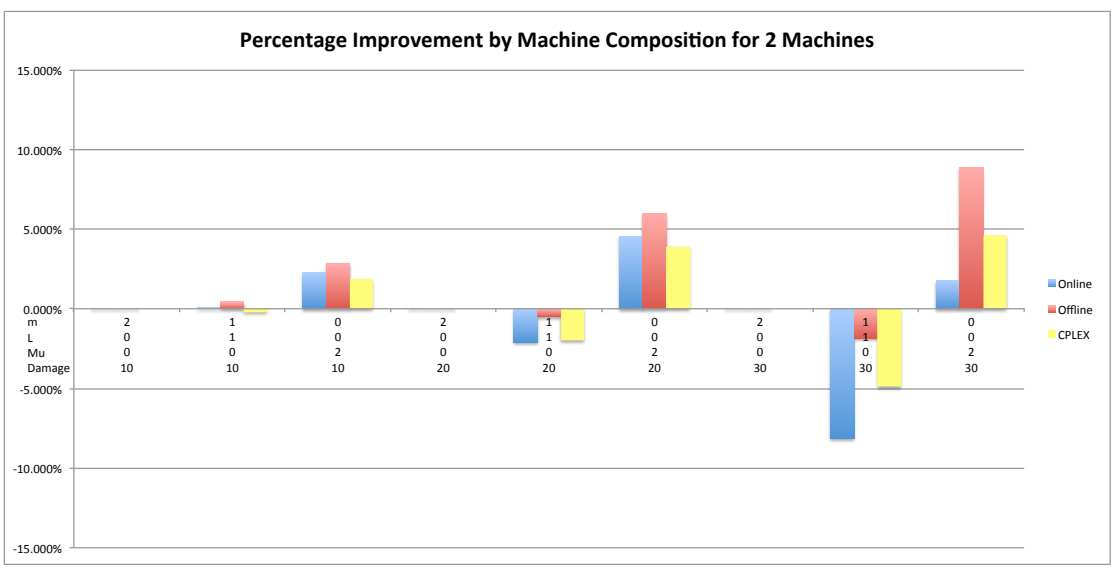


Figure 5.11: Average objective function value for combinations of 2 machines.

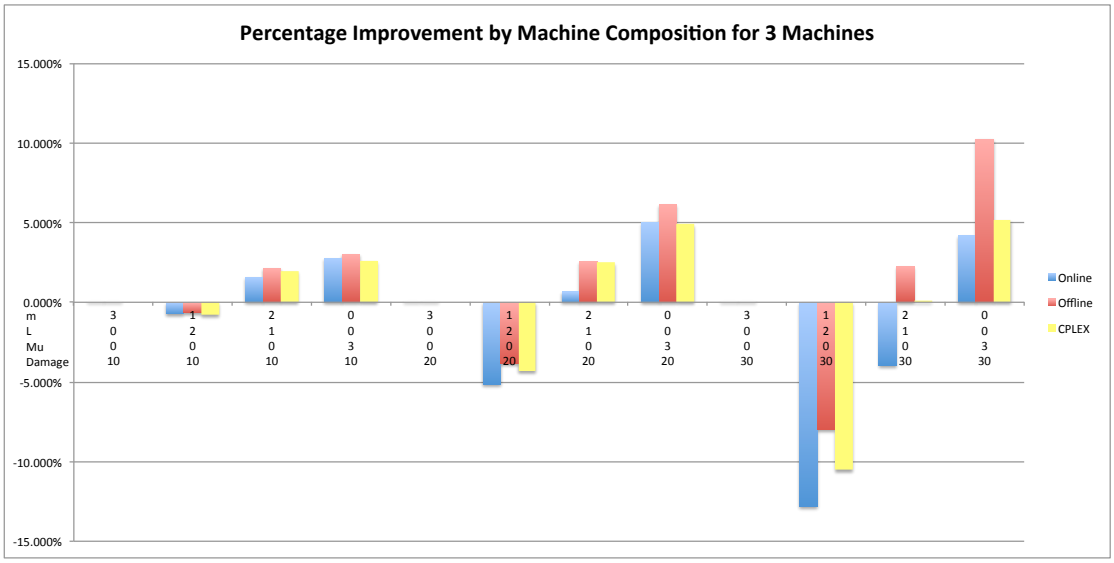


Figure 5.12: Average objective function value for combinations of 3 machines.

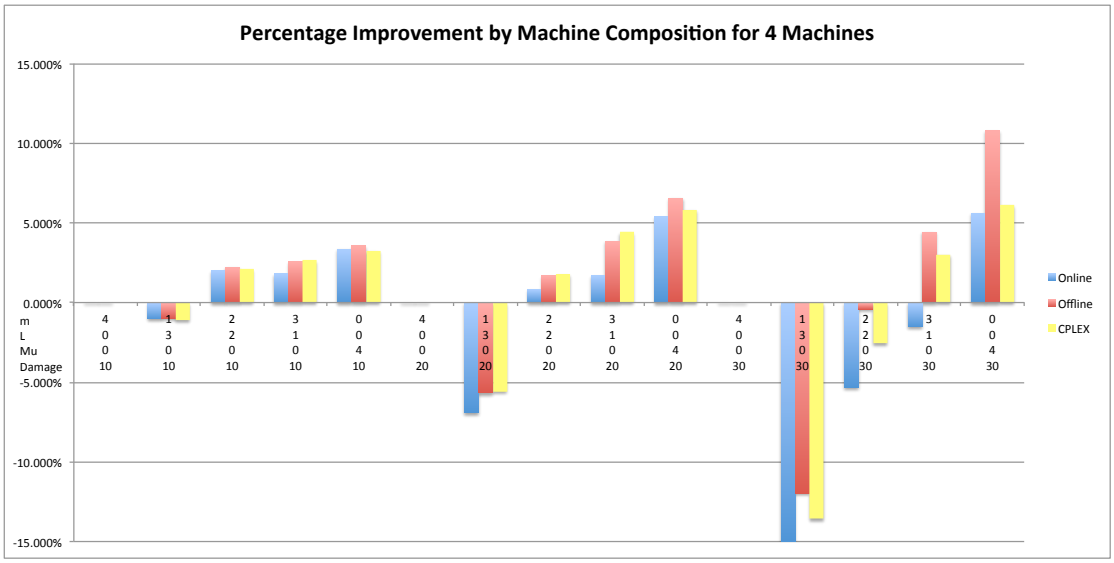


Figure 5.13: Average objective function value for combinations of 4 machines.

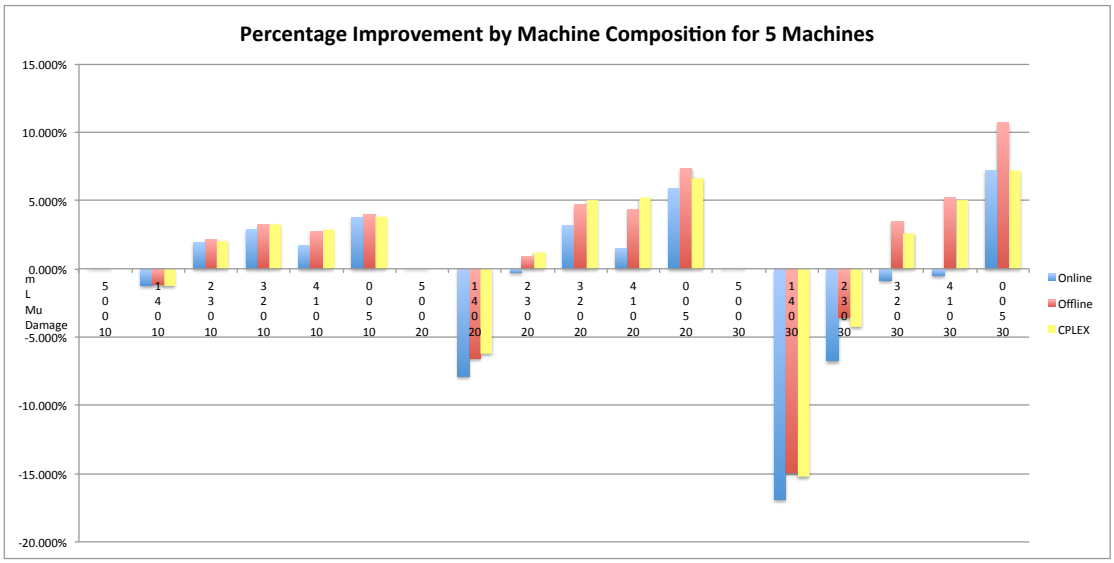
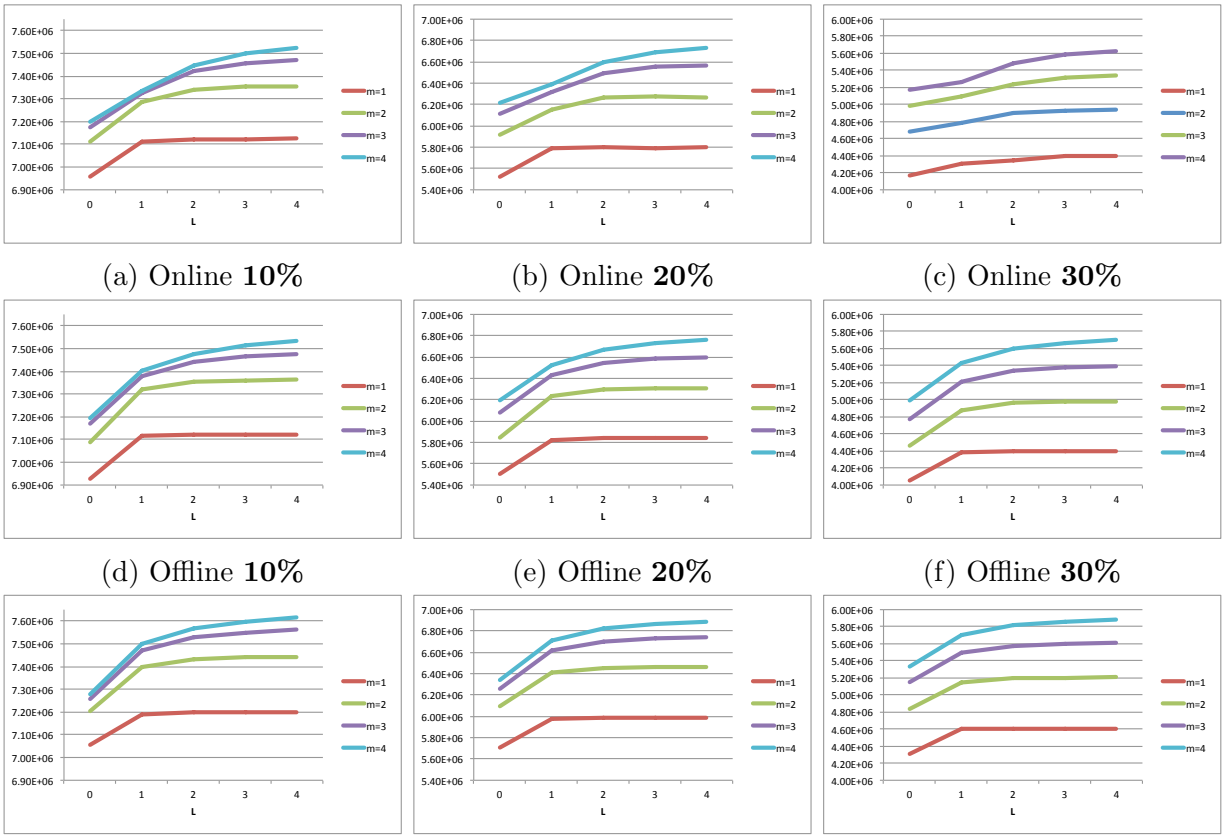


Figure 5.14: Average objective function value for combinations of 5 machines.

We have demonstrated that specialized machines in the flexible release date setting improve the objective function value. A decision maker now must make decisions on how many specialized machines to hire or coordinate with. Figures 5.15 look at the benefit (change in the objective function value) of adding extra specialized work groups for INDS problems with flexible release dates. In these graphs we can see that there is a phenomenon of diminishing returns as more specialized work groups are added. An initial jump occurs with the addition of one specialized machine and then less benefit is realized with each additional specialized machine. It appears, that the benefit starts to plateau when the number of traditional machines equals the number of specialized machines. This makes sense because the traditional machines need to 'keep up' with the specialized machines. Take the example with 4 specialized machines and 1 traditional machine. Provided the processing times are around the same magnitude, the specialized machines are able to move the release dates of 4 arcs earlier at one time. However, there is only 1 traditional machine able to process these arcs once their release dates are moved earlier in time, creating a bottleneck in the operation. Lastly, we would expect that the benefit of extra specialized work groups would be more dramatic for the online problem, since release dates are unknown and these machines would be expected to be utilized more frequently. This figure does not indicate that this hypothesis is as strong as one would expect.



(a) Online **10%** (b) Online **20%** (c) Online **30%**

(d) Offline **10%** (e) Offline **20%** (f) Offline **30%**

(g) CPLEX **10%** (h) CPLEX **20%** (i) CPLEX **30%**

Figure 5.15: Average objective function value for $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ by damage and machine profile.

Release Dates Utilization We now examine the utilization of the traditional machines for $Pm|MF, r_{ij}|Cumulative$, where the utilization is calculated as $\frac{\text{time working}}{\text{total time}}$. Figure 5.16 shows the average utilization across all machines by damage and machine number. Specifically, if $m = 2$ the utilization for the first traditional machine and second traditional machine are averaged and displayed. Table 5.7 shows the average utilization for each individual machine by number of machines and damage. Both the figure and table average their respective utilization values over the 15 random test instances. We notice that as the damage percentage increases so does the utilization. Further, it is clear that the offline dispatching rule is not fully utilizing the machine resources, which attributes to a point previously addressed about occupying machines while they are waiting for the release date of an important arc. Also, we notice that even with 5 work groups we still see very high utilization numbers for both the online dispatching rule and IP. This type of analysis would be useful for managers when determining the proper number of work groups as they are scenario planning for extreme events.

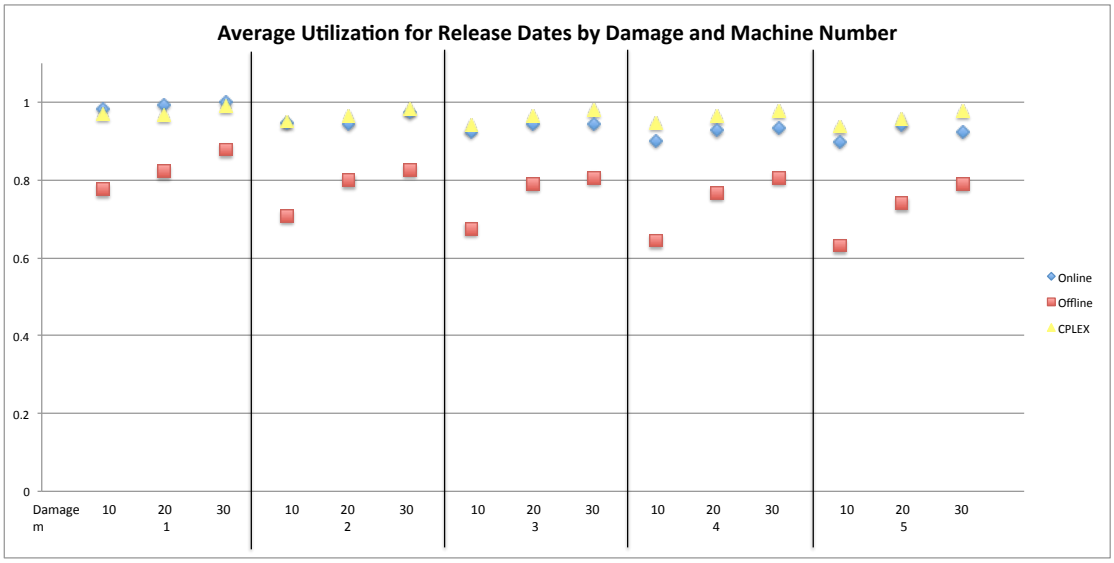
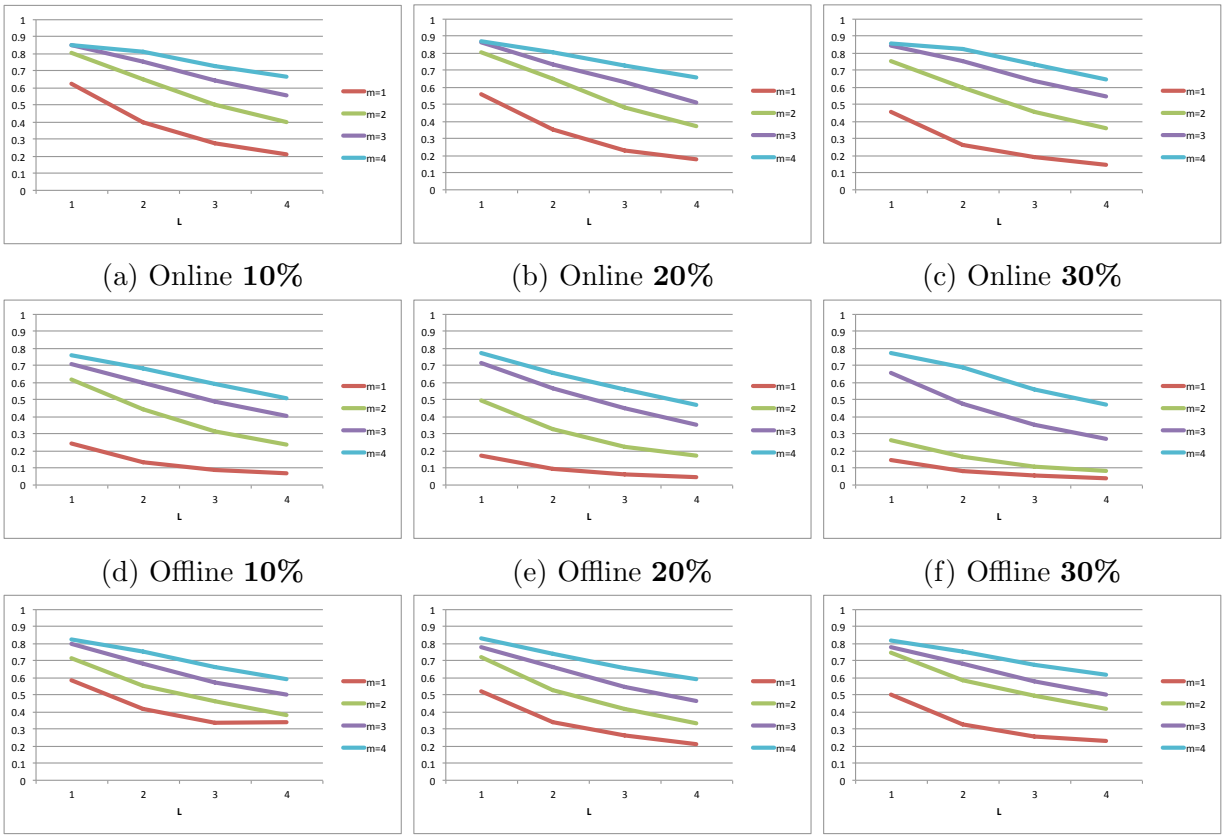


Figure 5.16: Average utilization for $\boldsymbol{Pm|MF, r_{ij}|Cumulative}$ by damage and machine.

Table 5.7: $\boldsymbol{Pm|MF, r_{ij}|Cumulative}$ average utilization by machine.

					Offline					Online					IP				
m	L	Mu	Damage	δ	$m1$	$m2$	$m3$	$m4$	$m5$	$m1$	$m2$	$m3$	$m4$	$m5$	$m1$	$m2$	$m3$	$m4$	$m5$
1	0	0	10%	-	77.67%	-	-	-	-	98.50%	-	-	-	-	97.17%	-	-	-	-
			20%	-	82.17%	-	-	-	-	99.50%	-	-	-	-	96.67%	-	-	-	-
			30%	-	87.83%	-	-	-	-	100.00%	-	-	-	-	99.00%	-	-	-	-
2	0	0	10%	-	69.17%	72.33%	-	-	-	95.33%	94.33%	-	-	-	95.33%	94.83%	-	-	-
			20%	-	80.50%	79.33%	-	-	-	94.50%	94.50%	-	-	-	97.17%	95.67%	-	-	-
			30%	-	81.00%	84.17%	-	-	-	97.50%	97.17%	-	-	-	97.50%	99.00%	-	-	-
3	0	0	10%	-	69.00%	64.17%	68.83%	-	-	94.67%	91.00%	91.33%	-	-	94.50%	94.00%	94.00%	-	-
			20%	-	77.50%	77.83%	81.83%	-	-	95.33%	93.83%	94.00%	-	-	96.67%	95.33%	96.83%	-	-
			30%	-	79.83%	81.50%	80.67%	-	-	95.33%	93.50%	94.17%	-	-	98.33%	98.17%	97.83%	-	-
4	0	0	10%	-	67.83%	60.33%	62.50%	67.33%	-	94.00%	90.33%	89.50%	87.17%	-	93.67%	95.17%	94.67%	95.67%	-
			20%	-	74.17%	73.17%	78.17%	80.83%	-	93.67%	93.83%	92.83%	91.33%	-	97.00%	96.00%	97.00%	95.67%	-
			30%	-	80.17%	79.67%	78.00%	84.50%	-	93.83%	94.83%	92.67%	92.33%	-	98.67%	97.50%	97.17%	97.67%	-
5	0	0	10%	-	65.00%	55.50%	69.00%	65.17%	60.67%	93.33%	90.33%	89.33%	87.83%	88.33%	93.50%	94.83%	93.00%	94.00%	94.00%
			20%	-	75.17%	68.00%	76.67%	74.00%	76.17%	94.83%	95.00%	94.00%	93.50%	92.83%	95.17%	97.83%	95.50%	94.83%	96.00%
			30%	-	77.33%	78.67%	77.33%	80.83%	80.33%	94.83%	92.33%	91.00%	91.67%	91.83%	97.83%	97.67%	97.83%	98.00%	97.33%

Flexible Release Dates Utilization We now examine the utilization of the traditional and specialized machines for $Opt - FF2|MF, flex - r_{ij}|Cumulative$. Figures 5.17 and 5.18 show the average utilization by solution method and different combinations of traditional and specialized machines for damage and δ, respectively. Tables 5.8, 5.9, and 5.10 display the average utilization for $\delta = 0.1, 0.2$, and 0.3, respectively. In Figures 5.17, we see that as the number of specialized machines increases, the average utilization decreases for all solution methods and damage percentages. The decreasing trend is more dramatic (negative slope) for smaller numbers of traditional machines. We do not notice this downward trend as dramatically in Figures 5.18, in particular for higher δ values. Higher δ values directly increase the processing times, $\bar{p}_{ij}$, for specialized machines, which for the same solution would cause higher utilization values. However, there should be a break-even point where if the processing times for the specialized machine become too large, it will be beneficial to wait instead of process. This demonstrates the optional nature of INDS problems with flexible release dates. This option to wait is only available for the offline setting as in the online setting the release dates are unknown and we determine that it is too risky to wait around an unknown amount of time for the release date of an arc. This is reflected in the higher utilization values for the online dispatching rule shown in Figure 5.18 that remain high even for the largest δ value.

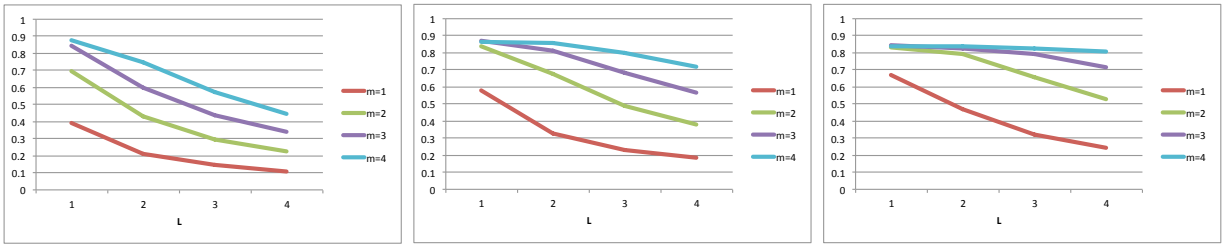


(a) Online **10%** (b) Online **20%** (c) Online **30%**

(d) Offline **10%** (e) Offline **20%** (f) Offline **30%**

(g) IP **10%** (h) IP **20%** (i) IP **30%**

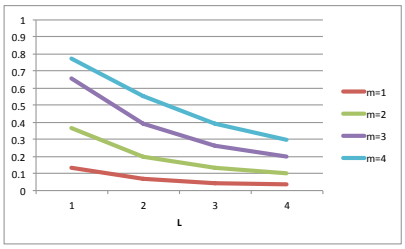
Figure 5.17: Average utilization for $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ by solution method, damage, and machine profile.



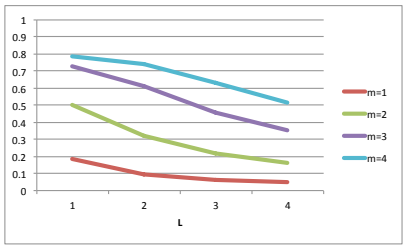
(a) Online **0.1**

(b) Online **0.2**

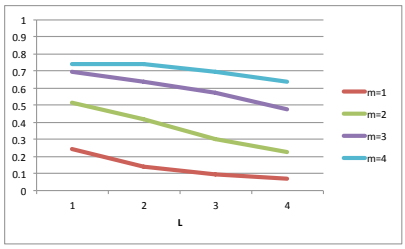
(c) Online **0.3**



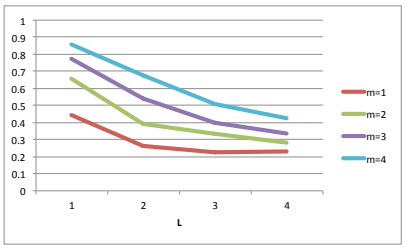
(d) Offline **0.1**



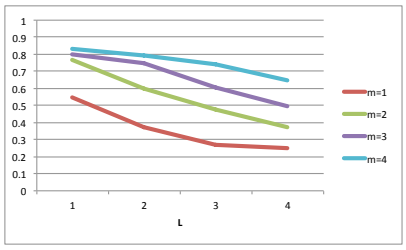
(e) Offline **0.2**



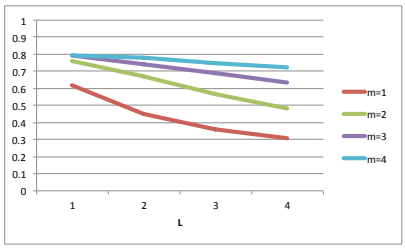
(f) Offline **0.3**



(g) IP **0.1**



(h) IP **0.2**



(i) IP **0.3**

Figure 5.18: Average utilization for $\boldsymbol{Opt-FF2|MF, flex-r_{ij}|Cumulative}$ by solution method, $\boldsymbol{\delta}$, and machine profile.

Table 5.8: $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ average utilization by machine for $\boldsymbol{\delta = 0.1}$.

					Offline								Online								IP							
m	L	Mu	Damage	δ	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$
1	1	0	10%	0.1	96.33%	-	-	-	18.83%	-	-	-	96.33%	-	-	-	44.00%	-	-	-	97.00%	-	-	-	51.33%	-	-	-
			20%	0.1	96.83%	-	-	-	10.83%	-	-	-	96.67%	-	-	-	43.00%	-	-	-	97.50%	-	-	-	39.50%	-	-	-
			30%	0.1	97.33%	-	-	-	10.50%	-	-	-	96.17%	-	-	-	31.33%	-	-	-	98.17%	-	-	-	42.83%	-	-	-
	2	0	10%	0.1	96.33%	-	-	-	10.00%	8.83%	-	-	96.33%	-	-	-	24.00%	23.17%	-	-	97.17%	-	-	-	31.67%	31.17%	-	-
			20%	0.1	96.83%	-	-	-	5.00%	5.83%	-	-	96.83%	-	-	-	21.50%	21.33%	-	-	97.67%	-	-	-	24.83%	22.00%	-	-
			30%	0.1	97.33%	-	-	-	5.83%	4.67%	-	-	97.33%	-	-	-	18.50%	17.67%	-	-	98.17%	-	-	-	25.33%	24.00%	-	-
	3	0	10%	0.1	96.33%	-	-	-	6.67%	7.00%	5.17%	-	96.33%	-	-	-	18.17%	15.00%	14.00%	-	97.17%	-	-	-	32.33%	36.33%	34.67%	-
			20%	0.1	96.83%	-	-	-	5.00%	4.67%	1.17%	-	96.83%	-	-	-	15.50%	14.67%	14.33%	-	97.67%	-	-	-	18.00%	12.50%	17.67%	-
			30%	0.1	97.33%	-	-	-	4.17%	4.00%	2.33%	-	97.33%	-	-	-	12.67%	12.67%	12.67%	-	98.17%	-	-	-	22.67%	15.17%	13.83%	-
	4	0	10%	0.1	96.33%	-	-	-	5.50%	5.83%	4.67%	2.83%	96.33%	-	-	-	14.00%	12.33%	10.67%	11.17%	97.17%	-	-	-	33.17%	32.33%	32.00%	38.00
			20%	0.1	96.83%	-	-	-	3.83%	5.00%	1.17%	0.83%	96.83%	-	-	-	12.17%	12.17%	10.50%	9.17%	97.50%	-	-	-	20.17%	15.00%	15.17%	15.00
			30%	0.1	97.33%	-	-	-	3.50%	4.00%	2.33%	0.67%	97.33%	-	-	-	10.67%	8.83%	9.17%	7.50%	98.17%	-	-	-	24.17%	14.17%	17.83%	16.33
2	1	0	10%	0.1	95.50%	91.50%	-	-	51.67%	-	-	-	91.33%	89.00%	-	-	71.67%	-	-	-	96.50%	96.17%	-	-	65.83%	-	-	-
			20%	0.1	96.67%	94.67%	-	-	39.33%	-	-	-	95.67%	92.33%	-	-	72.00%	-	-	-	97.83%	97.83%	-	-	62.67%	-	-	-
			30%	0.1	96.83%	93.00%	-	-	18.17%	-	-	-	92.17%	91.33%	-	-	65.50%	-	-	-	98.17%	98.00%	-	-	68.83%	-	-	-
	2	0	10%	0.1	96.00%	94.83%	-	-	29.17%	25.17%	-	-	94.83%	93.83%	-	-	46.67%	45.33%	-	-	96.50%	96.33%	-	-	41.33%	33.33%	-	-
			20%	0.1	96.67%	96.00%	-	-	20.33%	19.67%	-	-	93.50%	93.50%	-	-	44.83%	41.83%	-	-	97.67%	97.50%	-	-	38.33%	37.00%	-	-
			30%	0.1	97.33%	96.50%	-	-	14.33%	10.17%	-	-	95.50%	93.83%	-	-	41.33%	40.33%	-	-	98.00%	98.33%	-	-	41.83%	43.33%	-	-
	3	0	10%	0.1	96.00%	95.33%	-	-	19.17%	17.83%	17.33%	-	94.50%	94.17%	-	-	32.83%	30.83%	31.33%	-	96.50%	97.00%	-	-	34.67%	31.00%	36.67%	-
			20%	0.1	96.67%	96.00%	-	-	14.00%	12.50%	13.50%	-	91.50%	91.83%	-	-	30.17%	30.00%	28.00%	-	97.50%	97.50%	-	-	28.17%	24.00%	28.67%	-
			30%	0.1	97.33%	96.50%	-	-	9.50%	8.33%	6.67%	-	96.67%	95.17%	-	-	29.33%	27.50%	28.33%	-	98.33%	98.00%	-	-	43.00%	36.67%	36.00%	-
	4	0	10%	0.1	96.00%	95.33%	-	-	16.17%	12.00%	12.17%	14.00%	94.00%	94.17%	-	-	25.67%	23.33%	23.00%	23.33%	96.33%	97.00%	-	-	32.67%	29.00%	23.50%	34.67
			20%	0.1	96.67%	96.00%	-	-	12.00%	10.17%	8.67%	9.17%	91.83%	91.50%	-	-	22.83%	23.67%	21.17%	20.67%	97.17%	97.67%	-	-	27.83%	19.00%	19.50%	24.17
			30%	0.1	97.33%	96.50%	-	-	8.17%	5.67%	4.67%	6.00%	96.33%	95.17%	-	-	21.83%	21.67%	22.67%	21.00%	98.17%	98.17%	-	-	35.67%	28.00%	29.83%	33.50
3	1	0	10%	0.1	90.67%	83.17%	77.67%	-	69.50%	-	-	-	82.50%	76.83%	73.50%	-	83.17%	-	-	-	95.50%	96.33%	95.83%	-	77.33%	-	-	-
			20%	0.1	93.83%	90.50%	87.50%	-	69.00%	-	-	-	87.33%	84.83%	82.50%	-	83.50%	-	-	-	96.33%	97.17%	97.17%	-	77.50%	-	-	-
			30%	0.1	96.67%	92.67%	89.33%	-	58.17%	-	-	-	92.17%	90.83%	86.33%	-	85.83%	-	-	-	97.33%	96.83%	98.33%	-	76.33%	-	-	-
	2	0	10%	0.1	95.50%	95.00%	93.17%	-	46.33%	44.17%	-	-	92.33%	89.67%	87.33%	-	63.50%	61.83%	-	-	96.33%	97.17%	97.17%	-	58.67%	53.33%	-	-
			20%	0.1	96.50%	96.00%	95.00%	-	40.50%	38.50%	-	-	92.83%	92.00%	90.50%	-	58.67%	58.33%	-	-	97.00%	96.50%	98.17%	-	53.50%	44.83%	-	-
			30%	0.1	97.17%	96.50%	93.00%	-	34.17%	32.33%	-	-	94.17%	92.83%	91.00%	-	59.17%	59.00%	-	-	98.67%	97.67%	98.00%	-	58.33%	56.67%	-	-
	3	0	10%	0.1	96.17%	95.50%	95.17%	-	31.67%	31.83%	29.00%	-	90.00%	91.83%	93.33%	-	47.17%	45.67%	44.83%	-	95.83%	97.00%	96.83%	-	44.50%	39.67%	39.50%	-
			20%	0.1	96.83%	96.00%	96.67%	-	27.17%	27.83%	26.00%	-	91.83%	92.00%	92.50%	-	42.50%	42.00%	40.83%	-	97.00%	96.83%	97.00%	-	39.50%	35.17%	33.83%	-
			30%	0.1	97.33%	96.50%	96.00%	-	22.00%	23.33%	20.33%	-	94.33%	92.50%	92.17%	-	44.00%	42.83%	42.00%	-	97.83%	98.17%	98.17%	-	44.67%	41.33%	42.17%	-
	4	0	10%	0.1	96.17%	95.50%	95.17%	-	24.50%	22.83%	22.17%	23.00%	92.00%	92.17%	92.83%	-	37.33%	34.67%	36.50%	33.83%	96.83%	96.67%	96.33%	-	37.00%	29.50%	36.33%	32.50
			20%	0.1	96.83%	96.00%	96.67%	-	21.83%	20.50%	19.33%	19.33%	93.33%	91.67%	93.83%	-	32.83%	32.67%	31.17%	30.83%	97.00%	96.33%	97.17%	-	31.50%	23.50%	30.83%	33.17
			30%	0.1	97.33%	96.50%	96.00%	-	16.33%	15.67%	16.33%	17.33%	93.33%	93.00%	92.67%	-	33.00%	35.00%	33.50%	33.67%	98.33%	98.17%	97.67%	-	38.17%	33.33%	36.00%	41.17
4	1	0	10%	0.1	85.33%	76.17%	72.00%	71.17%	78.00%	-	-	-	68.50%	68.50%	67.83%	63.50%	85.33%	-	-	-	94.67%	96.67%	95.00%	95.00%	84.83%	-	-	-
			20%	0.1	87.00%	85.00%	82.67%	84.83%	78.00%	-	-	-	80.50%	73.50%	72.83%	71.00%	87.33%	-	-	-	95.50%	96.50%	96.50%	96.00%	85.83%	-	-	-
			30%	0.1	91.50%	89.17%	84.00%	84.67%	76.17%	-	-	-	87.00%	85.83%	80.00%	75.00%	90.33%	-	-	-	96.67%	96.17%	95.67%	97.83%	86.17%	-	-	-
	2	0	10%	0.1	92.67%	90.83%	89.50%	88.67%	60.33%	56.67%	-	-	87.33%	88.50%	84.33%	82.00%	76.50%	73.67%	-	-	95.17%	95.67%	96.17%	95.33%	69.00%	66.67%	-	-
			20%	0.1	95.83%	95.33%	94.00%	91.67%	53.67%	51.83%	-	-	91.17%	91.33%	89.83%	87.50%	73.83%	72.17%	-	-	96.83%	97.00%	96.50%	97.50%	66.17%	65.50%	-	-
			30%	0.1	96.83%	96.50%	93.17%	92.33%	55.33%	54.00%	-	-	94.00%	92.17%	89.83%	89.00%	76.33%	76.33%	-	-	97.50%	98.00%	97.50%	97.83%	69.17%	69.00%	-	-
	3	0	10%	0.1	95.67%	94.83%	95.33%	93.50%	42.50%	41.83%	42.17%	-	92.17%	92.17%	91.00%	89.33%	60.67%	59.00%	56.83%	-	95.83%	96.17%	96.50%	95.50%	48.50%	50.67%	52.33%	-
			20%	0.1	96.33%	95.83%	96.67%	94.50%	37.17%	36.83%	38.00%	-	91.17%	90.33%	93.83%	88.83%	57.17%	56.50%	56.33%	-	96.67%	97.17%	97.17%	97.83%	50.50%	47.00%	51.67%	-
			30%	0.1	97.33%	96.50%	96.17%	93.83%	38.50%	39.33%	38.00%	-	91.83%	92.17%	92.00%	89.00%	55.67%	55.17%	55.83%	-	97.67%	97.83%	98.00%	98.17%	54.83%	50.50%	50.83%	-
	4	0	10%	0.1	95.67%	95.50%	95.50%	95.67%	33.67%	31.67%	32.00%	31.67%	92.83%	91.67%	91.67%	92.33%	48.33%	47.00%	46.50%	46.00%	95.83%	95.33%	96.33%	95.33%	40.17%	43.33%	40.00%	38.33
			20%	0.1	96.83%	96.00%	96.50%	96.00%	29.17%	29.67%	27.33%	26.50%	92.00%	92.17%	92.33%	93.00%	44.83%	46.33%	44.67%	44.50%	97.50%	96.50%	96.67%	96.67%	41.33%	41.00%	37.33%	45.33
			30%	0.1	97.33%	96.50%	96.17%	96.17%	29.67%	29.00%	28.33%	28.33%	91.50%	89.67%	87.83%	89.33%	42.00%	42.50%	40.50%	41.50%	98.33%	97.67%	98.00%	98.00%	49.83%	44.67%	43.17%	46.00

Table 5.9: $\boldsymbol{Opt - FF2|MF, flex - r_{ij}|Cumulative}$ average utilization by machine for $\boldsymbol{\delta = 0.2}$.

					Offline								Online								IP							
m	L	Mu	Damage	δ	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$
1	1	0	10%	0.2	93.17%	-	-	-	25.67%	-	-	-	93.33%	-	-	-	67.17%	-	-	-	97.83%	-	-	-	61.33%	-	-	-
			20%	0.2	94.17%	-	-	-	16.17%	-	-	-	94.17%	-	-	-	57.67%	-	-	-	97.33%	-	-	-	54.17%	-	-	-
			30%	0.2	94.83%	-	-	-	14.00%	-	-	-	94.67%	-	-	-	48.67%	-	-	-	98.33%	-	-	-	49.00%	-	-	-
	2	0	10%	0.2	94.33%	-	-	-	15.67%	11.00%	-	-	94.00%	-	-	-	39.67%	38.33%	-	-	97.67%	-	-	-	45.17%	39.50%	-	-
			20%	0.2	95.33%	-	-	-	7.83%	8.33%	-	-	95.17%	-	-	-	31.67%	35.67%	-	-	97.83%	-	-	-	36.33%	36.33%	-	-
			30%	0.2	95.50%	-	-	-	8.83%	6.50%	-	-	95.33%	-	-	-	26.33%	26.33%	-	-	98.33%	-	-	-	34.33%	31.00%	-	-
	3	0	10%	0.2	94.33%	-	-	-	9.33%	9.33%	8.00%	-	94.17%	-	-	-	29.50%	28.00%	24.33%	-	97.67%	-	-	-	31.33%	27.17%	28.00%	-
			20%	0.2	95.33%	-	-	-	7.33%	7.17%	1.67%	-	95.17%	-	-	-	25.17%	21.83%	22.67%	-	97.83%	-	-	-	31.50%	24.17%	27.67%	-
			30%	0.2	95.50%	-	-	-	7.17%	5.17%	3.00%	-	95.33%	-	-	-	18.17%	20.17%	18.00%	-	98.33%	-	-	-	28.17%	20.83%	25.67%	-
	4	0	10%	0.2	94.33%	-	-	-	7.33%	8.67%	7.83%	2.83%	94.17%	-	-	-	19.67%	19.83%	21.50%	20.67%	97.67%	-	-	-	32.50%	25.50%	34.17%	29.67%
			20%	0.2	95.33%	-	-	-	6.17%	7.67%	1.67%	0.67%	95.17%	-	-	-	19.33%	19.83%	18.83%	18.67%	97.83%	-	-	-	28.17%	17.00%	13.17%	22.00%
			30%	0.2	95.50%	-	-	-	5.83%	5.17%	3.00%	1.33%	95.33%	-	-	-	17.17%	14.67%	13.33%	16.17%	98.33%	-	-	-	30.83%	20.83%	18.67%	25.00%
2	1	0	10%	0.2	83.67%	83.17%	-	-	66.50%	-	-	-	79.50%	72.83%	-	-	84.17%	-	-	-	95.50%	96.33%	-	-	74.17%	-	-	-
			20%	0.2	93.33%	90.83%	-	-	53.50%	-	-	-	88.50%	83.00%	-	-	84.33%	-	-	-	96.83%	97.50%	-	-	77.33%	-	-	-
			30%	0.2	92.83%	89.00%	-	-	30.33%	-	-	-	88.50%	86.33%	-	-	82.83%	-	-	-	98.00%	98.00%	-	-	78.00%	-	-	-
	2	0	10%	0.2	93.00%	92.00%	-	-	49.17%	44.67%	-	-	91.00%	89.83%	-	-	73.00%	67.67%	-	-	96.67%	97.33%	-	-	61.50%	59.67%	-	-
			20%	0.2	95.33%	94.67%	-	-	35.17%	31.83%	-	-	93.50%	92.00%	-	-	69.50%	68.00%	-	-	97.67%	98.00%	-	-	56.00%	56.17%	-	-
			30%	0.2	95.17%	95.17%	-	-	16.17%	16.67%	-	-	93.00%	92.33%	-	-	63.50%	62.83%	-	-	98.33%	97.33%	-	-	64.00%	62.83%	-	-
	3	0	10%	0.2	94.17%	93.50%	-	-	30.17%	31.83%	33.00%	-	92.17%	90.67%	-	-	52.17%	50.83%	50.83%	-	96.67%	96.83%	-	-	50.33%	48.00%	46.67%	-
			20%	0.2	95.50%	94.67%	-	-	22.67%	22.50%	21.50%	-	93.17%	90.50%	-	-	51.17%	46.00%	45.33%	-	98.00%	97.50%	-	-	49.33%	39.83%	42.50%	-
			30%	0.2	95.50%	95.50%	-	-	13.50%	11.50%	8.00%	-	94.17%	93.67%	-	-	49.33%	46.67%	47.83%	-	97.67%	98.17%	-	-	56.67%	49.50%	45.50%	-
	4	0	10%	0.2	94.33%	94.00%	-	-	26.33%	24.50%	22.17%	22.00%	92.33%	91.17%	-	-	40.67%	41.50%	37.83%	37.83%	96.83%	96.33%	-	-	47.33%	34.67%	33.83%	35.83%
			20%	0.2	95.50%	94.67%	-	-	18.67%	16.83%	15.17%	16.00%	92.50%	90.33%	-	-	37.50%	39.00%	36.33%	35.83%	98.00%	97.67%	-	-	38.00%	29.83%	31.00%	33.83%
			30%	0.2	95.50%	95.50%	-	-	9.33%	9.00%	6.83%	7.83%	94.00%	93.50%	-	-	39.33%	35.83%	38.17%	36.00%	98.00%	97.83%	-	-	44.33%	41.83%	39.17%	36.67%
3	1	0	10%	0.2	78.00%	70.67%	68.33%	-	73.67%	-	-	-	66.67%	66.00%	59.67%	-	85.33%	-	-	-	94.83%	95.67%	95.67%	-	80.00%	-	-	-
			20%	0.2	85.50%	84.00%	82.17%	-	76.00%	-	-	-	74.83%	82.67%	74.67%	-	89.17%	-	-	-	96.50%	96.67%	97.83%	-	81.17%	-	-	-
			30%	0.2	90.50%	87.50%	86.33%	-	69.17%	-	-	-	81.83%	85.17%	81.83%	-	87.67%	-	-	-	98.17%	98.50%	97.83%	-	78.00%	-	-	-
	2	0	10%	0.2	86.67%	82.33%	82.67%	-	67.67%	65.00%	-	-	80.33%	78.00%	70.83%	-	82.50%	80.50%	-	-	95.50%	95.67%	94.83%	-	76.50%	74.33%	-	-
			20%	0.2	92.33%	91.17%	89.00%	-	67.17%	61.67%	-	-	85.50%	86.67%	82.50%	-	82.83%	79.83%	-	-	96.67%	96.50%	97.33%	-	75.00%	73.67%	-	-
			30%	0.2	94.50%	95.00%	90.33%	-	52.83%	51.67%	-	-	90.83%	92.00%	90.00%	-	81.00%	79.50%	-	-	97.17%	97.83%	97.83%	-	74.00%	74.50%	-	-
	3	0	10%	0.2	92.67%	89.50%	90.50%	-	54.67%	52.50%	52.83%	-	89.33%	86.33%	83.17%	-	72.00%	70.50%	68.00%	-	96.83%	96.67%	96.17%	-	65.00%	58.67%	59.83%	-
			20%	0.2	95.50%	94.67%	93.00%	-	48.17%	48.17%	45.67%	-	92.00%	90.67%	89.00%	-	68.00%	67.00%	68.17%	-	97.17%	97.00%	97.33%	-	56.33%	57.50%	62.00%	-
			30%	0.2	95.00%	95.33%	92.00%	-	39.17%	37.67%	33.00%	-	92.00%	92.33%	91.50%	-	67.50%	67.17%	67.00%	-	97.83%	98.33%	97.50%	-	64.67%	62.83%	57.50%	-
	4	0	10%	0.2	94.17%	92.50%	91.67%	-	42.17%	43.00%	41.83%	39.17%	90.50%	90.17%	86.17%	-	62.17%	59.33%	58.33%	57.33%	96.17%	96.50%	96.50%	-	59.50%	46.33%	48.00%	47.00%
			20%	0.2	95.50%	94.83%	94.17%	-	36.83%	36.67%	36.83%	32.83%	90.67%	90.00%	89.50%	-	54.83%	55.67%	51.83%	51.33%	96.83%	97.00%	97.67%	-	52.83%	46.83%	48.50%	47.17%
			30%	0.2	95.50%	95.50%	92.67%	-	31.50%	26.67%	26.00%	28.00%	91.33%	92.17%	90.50%	-	58.50%	56.67%	56.00%	54.67%	97.83%	98.67%	97.33%	-	55.67%	48.00%	48.83%	46.33%
4	1	0	10%	0.2	71.00%	70.00%	61.83%	66.33%	75.83%	-	-	-	59.00%	60.00%	53.17%	49.00%	85.33%	-	-	-	94.33%	95.33%	96.00%	93.33%	83.17%	-	-	-
			20%	0.2	81.00%	78.83%	74.50%	76.50%	79.83%	-	-	-	74.00%	74.50%	62.00%	66.50%	87.33%	-	-	-	96.33%	96.50%	96.17%	96.33%	85.17%	-	-	-
			30%	0.2	86.00%	84.00%	82.67%	81.83%	81.17%	-	-	-	79.00%	80.33%	74.50%	70.17%	86.17%	-	-	-	96.83%	97.00%	96.17%	97.67%	81.33%	-	-	-
	2	0	10%	0.2	79.17%	80.50%	78.33%	71.17%	76.50%	73.00%	-	-	72.33%	67.17%	67.50%	62.17%	83.67%	85.17%	-	-	94.17%	95.00%	94.17%	95.17%	82.17%	79.83%	-	-
			20%	0.2	86.83%	85.17%	86.50%	81.50%	73.50%	74.67%	-	-	79.67%	76.83%	72.67%	69.50%	86.33%	85.50%	-	-	95.17%	95.17%	96.33%	95.50%	80.17%	75.83%	-	-
			30%	0.2	90.67%	92.67%	86.17%	84.33%	73.00%	74.50%	-	-	86.33%	86.00%	81.00%	79.33%	86.33%	88.00%	-	-	96.33%	95.83%	97.00%	96.00%	83.00%	76.17%	-	-
	3	0	10%	0.2	85.33%	82.33%	85.17%	84.50%	67.67%	64.67%	64.17%	-	79.67%	80.83%	76.50%	77.67%	79.67%	79.33%	77.67%	-	93.50%	94.67%	95.33%	96.00%	73.17%	73.17%	75.00%	-
			20%	0.2	91.00%	92.50%	91.33%	90.00%	61.67%	62.00%	60.33%	-	83.67%	86.83%	83.00%	83.67%	81.00%	80.17%	77.83%	-	97.00%	96.00%	96.00%	96.83%	78.33%	75.33%	70.17%	-
			30%	0.2	94.50%	94.33%	90.50%	90.00%	63.83%	62.67%	60.17%	-	86.67%	89.67%	87.00%	84.33%	80.83%	82.17%	79.50%	-	97.67%	97.00%	96.33%	97.00%	78.00%	74.00%	70.33%	-
	4	0	10%	0.2	90.50%	89.33%	90.67%	89.00%	56.83%	56.17%	52.50%	54.17%	86.17%	85.83%	85.83%	83.83%	72.00%	72.17%	75.67%	72.67%	94.83%	94.83%	95.67%	95.33%	68.50%	64.83%	61.50%	58.83%
			20%	0.2	94.83%	94.00%	92.83%	93.33%	51.00%	50.50%	50.50%	47.00%	91.67%	90.33%	90.67%	90.33%	73.33%	72.50%	72.50%	70.50%	95.83%	96.33%	97.00%	96.67%	68.83%	64.00%	65.00%	59.67%
			30%	0.2	95.50%	95.50%	93.17%	93.83%	52.83%	49.33%	49.33%	48.50%	91.50%	90.17%	88.83%	88.00%	71.50%	70.50%	67.83%	71.00%	97.00%	98.00%	98.17%	97.33%	67.67%	62.00%	69.17%	66.83%

Table 5.10: $\boldsymbol{Opt-FF2|MF, flex-r_{ij}|Cumulative}$ average utilization by machine for $\boldsymbol{\delta = 0.3}$.

					Offline								Online								IP							
m	L	Mu	Damage	δ	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$	$m1$	$m2$	$m3$	$m4$	$L1$	$L2$	$L3$	$L4$
1	1	0	10%	0.3	90.00%	-	-	-	27.67%	-	-	-	86.50%	-	-	-	76.67%	-	-	-	96.83%	-	-	-	63.00%	-	-	-
			20%	0.3	90.67%	-	-	-	24.83%	-	-	-	89.50%	-	-	-	67.67%	-	-	-	97.50%	-	-	-	63.67%	-	-	-
			30%	0.3	93.83%	-	-	-	19.67%	-	-	-	93.50%	-	-	-	57.17%	-	-	-	97.50%	-	-	-	58.17%	-	-	-
	2	0	10%	0.3	92.33%	-	-	-	19.33%	14.00%	-	-	92.17%	-	-	-	56.17%	56.83%	-	-	97.33%	-	-	-	52.67%	50.50%	-	-
			20%	0.3	93.83%	-	-	-	15.00%	13.50%	-	-	93.83%	-	-	-	50.17%	52.33%	-	-	97.67%	-	-	-	45.67%	38.00%	-	-
			30%	0.3	95.33%	-	-	-	10.83%	10.83%	-	-	95.33%	-	-	-	33.83%	33.17%	-	-	97.83%	-	-	-	39.50%	42.67%	-	-
	3	0	10%	0.3	92.83%	-	-	-	13.67%	11.83%	7.83%	-	92.67%	-	-	-	39.83%	36.50%	40.67%	-	97.17%	-	-	-	44.83%	31.33%	37.33%	-
			20%	0.3	93.83%	-	-	-	10.50%	10.83%	7.17%	-	93.83%	-	-	-	34.00%	31.17%	26.33%	-	97.83%	-	-	-	36.33%	34.17%	34.67%	-
			30%	0.3	95.33%	-	-	-	8.17%	7.83%	5.67%	-	95.33%	-	-	-	22.67%	27.50%	28.50%	-	97.83%	-	-	-	32.83%	37.67%	36.83%	-
	4	0	10%	0.3	92.83%	-	-	-	10.50%	10.83%	8.00%	4.00%	93.00%	-	-	-	31.17%	32.33%	29.83%	27.00%	97.33%	-	-	-	37.67%	40.50%	40.00%	35.00%
			20%	0.3	93.83%	-	-	-	7.83%	11.50%	7.17%	2.00%	93.67%	-	-	-	25.67%	24.00%	22.50%	21.50%	97.83%	-	-	-	35.17%	21.50%	20.83%	30.83%
			30%	0.3	95.33%	-	-	-	7.17%	7.83%	5.67%	1.00%	95.33%	-	-	-	21.00%	17.33%	18.67%	19.17%	97.83%	-	-	-	30.33%	22.83%	28.00%	27.00%
2	1	0	10%	0.3	77.50%	73.00%	-	-	67.17%	-	-	-	70.83%	62.67%	-	-	85.83%	-	-	-	96.33%	95.00%	-	-	74.17%	-	-	-
			20%	0.3	87.00%	84.83%	-	-	56.67%	-	-	-	76.67%	76.17%	-	-	85.50%	-	-	-	96.83%	97.50%	-	-	77.17%	-	-	-
			30%	0.3	92.17%	82.83%	-	-	30.00%	-	-	-	85.83%	78.67%	-	-	78.00%	-	-	-	98.00%	98.00%	-	-	77.50%	-	-	-
	2	0	10%	0.3	87.83%	82.67%	-	-	61.00%	56.17%	-	-	78.50%	78.67%	-	-	80.50%	79.17%	-	-	96.83%	95.33%	-	-	67.33%	68.00%	-	-
			20%	0.3	92.50%	92.67%	-	-	45.33%	45.50%	-	-	86.33%	86.83%	-	-	80.33%	85.17%	-	-	97.17%	97.50%	-	-	61.00%	69.17%	-	-
			30%	0.3	94.00%	91.00%	-	-	24.00%	18.67%	-	-	93.50%	89.67%	-	-	78.33%	72.33%	-	-	97.83%	99.00%	-	-	66.17%	71.67%	-	-
	3	0	10%	0.3	90.17%	90.33%	-	-	43.00%	43.17%	46.33%	-	86.00%	82.00%	-	-	69.67%	64.83%	69.83%	-	97.00%	96.83%	-	-	60.00%	57.67%	52.33%	-
			20%	0.3	94.00%	94.17%	-	-	36.33%	31.00%	29.50%	-	92.67%	90.83%	-	-	70.83%	66.50%	66.00%	-	97.00%	97.50%	-	-	52.50%	55.17%	54.33%	-
			30%	0.3	95.17%	92.00%	-	-	16.17%	15.00%	10.67%	-	93.83%	91.00%	-	-	60.50%	63.83%	58.17%	-	98.17%	98.33%	-	-	63.33%	58.50%	58.50%	-
	4	0	10%	0.3	92.33%	90.17%	-	-	39.17%	33.33%	33.00%	26.83%	88.33%	86.17%	-	-	57.00%	57.17%	58.50%	54.33%	96.17%	97.33%	-	-	53.50%	46.17%	42.83%	44.33%
			20%	0.3	94.00%	94.17%	-	-	26.17%	22.67%	25.83%	22.17%	92.00%	89.17%	-	-	55.00%	54.00%	51.67%	50.00%	96.67%	97.67%	-	-	44.83%	49.17%	43.00%	38.67%
			30%	0.3	95.17%	92.17%	-	-	10.50%	9.67%	12.00%	9.67%	93.00%	90.50%	-	-	49.00%	49.33%	48.50%	49.67%	98.17%	97.83%	-	-	53.50%	55.50%	57.00%	47.33%
3	1	0	10%	0.3	72.17%	67.83%	60.50%	-	70.50%	-	-	-	57.00%	59.00%	56.17%	-	86.83%	-	-	-	95.67%	95.50%	95.00%	-	82.17%	-	-	-
			20%	0.3	79.17%	85.00%	78.33%	-	68.83%	-	-	-	77.00%	69.00%	66.83%	-	85.67%	-	-	-	97.83%	96.67%	96.50%	-	76.33%	-	-	-
			30%	0.3	89.00%	79.67%	83.00%	-	69.17%	-	-	-	76.00%	78.67%	73.33%	-	80.83%	-	-	-	97.83%	98.67%	97.00%	-	79.50%	-	-	-
	2	0	10%	0.3	79.67%	75.17%	74.67%	-	65.50%	72.50%	-	-	65.67%	65.33%	62.50%	-	83.33%	79.33%	-	-	94.67%	94.67%	95.33%	-	75.50%	72.17%	-	-
			20%	0.3	86.83%	89.17%	82.67%	-	64.17%	66.67%	-	-	77.83%	75.50%	71.50%	-	80.00%	80.17%	-	-	97.00%	96.00%	96.67%	-	77.17%	74.83%	-	-
			30%	0.3	92.17%	89.83%	89.00%	-	56.33%	58.33%	-	-	79.50%	85.33%	83.33%	-	88.00%	84.00%	-	-	98.00%	97.17%	98.17%	-	75.83%	69.00%	-	-
	3	0	10%	0.3	83.33%	79.00%	80.83%	-	68.83%	61.50%	60.00%	-	75.17%	72.67%	73.00%	-	81.83%	78.33%	74.33%	-	96.00%	96.17%	95.17%	-	64.00%	75.00%	68.33%	-
			20%	0.3	89.67%	91.33%	88.17%	-	59.67%	60.17%	63.50%	-	84.83%	81.67%	81.50%	-	78.50%	76.17%	82.33%	-	96.50%	97.17%	95.33%	-	72.17%	65.83%	69.00%	-
			30%	0.3	94.50%	91.50%	90.83%	-	48.83%	49.50%	46.17%	-	89.17%	87.50%	88.17%	-	80.17%	76.83%	84.67%	-	97.50%	97.83%	97.83%	-	68.50%	72.83%	67.83%	-
	4	0	10%	0.3	89.17%	83.17%	85.83%	-	59.50%	56.83%	55.17%	54.00%	82.67%	78.83%	81.67%	-	74.83%	73.00%	71.33%	69.33%	96.33%	96.33%	95.50%	-	68.33%	68.33%	68.33%	58.00%
			20%	0.3	93.17%	93.67%	90.83%	-	49.67%	50.17%	50.00%	48.33%	88.50%	89.17%	86.33%	-	68.83%	72.00%	64.00%	67.83%	97.17%	96.17%	96.83%	-	70.50%	53.50%	58.00%	61.33%
			30%	0.3	95.17%	92.17%	91.17%	-	35.33%	38.33%	36.83%	36.50%	90.83%	90.33%	89.00%	-	74.83%	73.33%	75.17%	72.17%	97.67%	97.50%	97.83%	-	62.33%	62.33%	64.00%	66.33%
4	1	0	10%	0.3	66.50%	63.17%	59.17%	58.83%	73.50%	-	-	-	60.17%	61.00%	43.50%	52.50%	84.00%	-	-	-	93.67%	96.00%	94.00%	94.67%	80.00%	-	-	-
			20%	0.3	75.33%	73.33%	76.67%	75.17%	74.50%	-	-	-	70.33%	68.83%	62.50%	59.83%	87.00%	-	-	-	95.83%	97.17%	96.17%	97.00%	79.50%	-	-	-
			30%	0.3	87.83%	75.50%	81.00%	81.33%	75.17%	-	-	-	78.83%	75.67%	70.50%	64.33%	80.50%	-	-	-	98.00%	96.83%	96.67%	97.67%	77.83%	-	-	-
	2	0	10%	0.3	74.83%	67.17%	67.00%	65.00%	70.50%	73.67%	-	-	64.17%	56.33%	55.17%	49.00%	84.00%	83.67%	-	-	93.83%	92.50%	93.17%	96.17%	78.17%	76.17%	-	-
			20%	0.3	85.17%	80.17%	74.67%	77.67%	70.67%	71.00%	-	-	70.50%	67.50%	66.17%	62.00%	81.33%	83.83%	-	-	97.00%	96.33%	96.17%	95.83%	81.67%	76.67%	-	-
			30%	0.3	86.17%	86.50%	82.17%	83.50%	79.67%	77.50%	-	-	77.33%	76.67%	74.50%	69.17%	85.67%	83.17%	-	-	96.67%	98.50%	96.83%	96.83%	79.00%	77.33%	-	-
	3	0	10%	0.3	77.00%	72.00%	76.17%	73.83%	69.00%	72.50%	71.17%	-	64.33%	64.83%	63.67%	60.50%	79.83%	83.67%	80.83%	-	94.17%	95.00%	95.17%	94.83%	77.33%	68.00%	77.17%	-
			20%	0.3	88.00%	83.00%	79.50%	85.00%	70.17%	71.67%	67.50%	-	73.17%	74.17%	74.50%	67.00%	82.50%	83.83%	81.83%	-	96.33%	95.00%	94.33%	95.50%	76.00%	70.33%	74.50%	-
			30%	0.3	91.00%	87.50%	89.00%	85.33%	71.67%	65.50%	67.00%	-	81.17%	82.17%	80.67%	77.33%	84.83%	84.50%	81.00%	-	95.17%	96.33%	97.17%	96.67%	77.67%	78.17%	74.83%	-
	4	0	10%	0.3	80.50%	78.00%	83.33%	79.67%	70.67%	66.17%	61.83%	62.67%	68.00%	73.17%	73.17%	69.50%	82.33%	82.50%	77.50%	76.67%	96.00%	94.83%	93.50%	93.33%	74.50%	72.67%	74.83%	69.83%
			20%	0.3	88.83%	89.50%	87.17%	86.17%	67.83%	62.17%	58.33%	63.67%	82.00%	81.33%	83.83%	76.50%	78.67%	82.83%	79.83%	79.83%	95.33%	95.83%	96.17%	95.17%	71.50%	71.33%	74.83%	68.67%
			30%	0.3	93.17%	89.83%	89.33%	89.83%	60.50%	64.17%	62.33%	63.33%	90.17%	83.67%	84.67%	81.50%	82.50%	81.17%	83.00%	82.33%	96.67%	96.17%	97.50%	96.67%	73.83%	75.00%	73.00%	68.17%

Multi-Function Machines Utilization We now examine the utilization for $MFM - Pm|MF, flex - r_{ij}|Cumulative$, INDS problems with multi-function machines. Figure 5.19 and Table 5.11 display the average utilization numbers across the 15 random test instances. Very high utilization numbers are apparent for each machine, damage, and δ combination. This again reinforces the desire for skilled labor, as work groups can perform two sets of tasks instead of one, leading to a higher utilization.

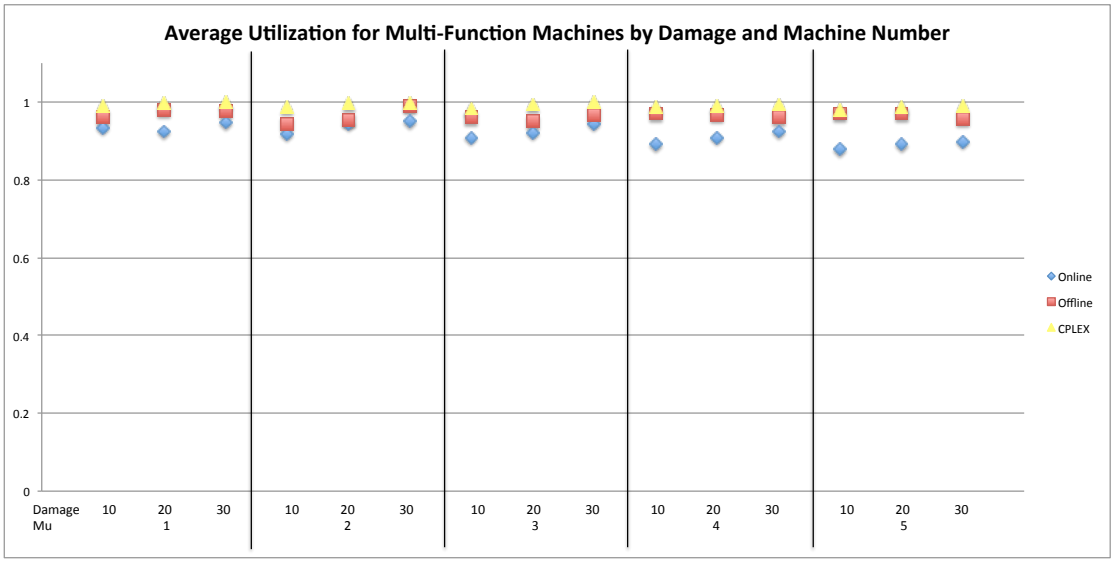


Figure 5.19: Average utilization for $\boldsymbol{MFM - Pm|MF, flex - r_{ij}|Cumulative}$ by damage and machine.

Table 5.11: $\boldsymbol{MFM - Pm|MF, flex - r_{ij}|Cumulative}$ average utilization by machine.

					Offline					Online					IP				
m	L	Mu	Damage	δ	$Mu1$	$Mu2$	$Mu3$	$Mu4$	$Mu5$	$Mu1$	$Mu2$	$Mu3$	$Mu4$	$Mu5$	$Mu1$	$Mu2$	$Mu3$	$Mu4$	$Mu5$
0	0	1	10%	0.1	97.00%	-	-	-	-	96.17%	-	-	-	-	99.33%	-	-	-	-
				0.2	95.67%	-	-	-	-	94.50%	-	-	-	-	98.83%	-	-	-	-
				0.3	95.83%	-	-	-	-	89.33%	-	-	-	-	99.33%	-	-	-	-
			20%	0.1	97.83%	-	-	-	-	96.50%	-	-	-	-	99.83%	-	-	-	-
				0.2	98.67%	-	-	-	-	92.67%	-	-	-	-	99.83%	-	-	-	-
				0.3	97.17%	-	-	-	-	88.50%	-	-	-	-	99.67%	-	-	-	-
			30%	0.1	98.83%	-	-	-	-	96.67%	-	-	-	-	99.83%	-	-	-	-
				0.2	97.83%	-	-	-	-	95.67%	-	-	-	-	100.00%	-	-	-	-
				0.3	96.00%	-	-	-	-	91.83%	-	-	-	-	99.83%	-	-	-	-
0	0	2	10%	0.1	98.67%	97.50%	-	-	-	93.50%	94.50%	-	-	-	98.83%	99.17%	-	-	-
				0.2	93.33%	93.00%	-	-	-	91.83%	92.00%	-	-	-	99.00%	98.33%	-	-	-
				0.3	90.50%	93.00%	-	-	-	89.67%	89.00%	-	-	-	98.50%	98.67%	-	-	-
			20%	0.1	97.00%	97.33%	-	-	-	98.17%	95.67%	-	-	-	99.67%	99.67%	-	-	-
				0.2	96.50%	96.50%	-	-	-	94.83%	93.83%	-	-	-	99.67%	99.83%	-	-	-
				0.3	92.17%	92.67%	-	-	-	91.83%	92.17%	-	-	-	99.00%	99.83%	-	-	-
			30%	0.1	99.83%	99.00%	-	-	-	98.33%	96.67%	-	-	-	100.00%	99.67%	-	-	-
				0.2	98.50%	99.33%	-	-	-	95.33%	94.67%	-	-	-	100.00%	99.67%	-	-	-
				0.3	98.50%	98.83%	-	-	-	93.33%	92.67%	-	-	-	99.83%	99.83%	-	-	-
0	0	3	10%	0.1	97.67%	97.67%	97.50%	-	-	98.00%	96.50%	95.83%	-	-	99.00%	99.00%	98.83%	-	-
				0.2	95.83%	95.83%	95.17%	-	-	90.50%	87.50%	92.00%	-	-	98.67%	97.83%	98.83%	-	-
				0.3	94.50%	95.83%	93.00%	-	-	86.17%	85.17%	84.33%	-	-	98.17%	98.33%	98.00%	-	-
			20%	0.1	97.33%	98.17%	97.50%	-	-	97.33%	97.00%	95.83%	-	-	99.67%	99.33%	99.83%	-	-
				0.2	94.67%	95.00%	95.67%	-	-	89.67%	91.67%	90.33%	-	-	99.50%	99.33%	99.17%	-	-
				0.3	92.33%	91.33%	94.83%	-	-	88.50%	87.33%	89.83%	-	-	99.17%	98.67%	99.83%	-	-
			30%	0.1	97.33%	97.83%	98.83%	-	-	97.33%	97.33%	96.33%	-	-	99.67%	99.83%	100.00%	-	-
				0.2	93.83%	96.67%	98.00%	-	-	95.83%	95.17%	93.67%	-	-	99.67%	100.00%	100.00%	-	-
				0.3	95.33%	96.00%	97.50%	-	-	91.33%	91.17%	90.17%	-	-	99.83%	99.67%	100.00%	-	-
0	0	4	10%	0.1	98.00%	97.67%	98.50%	98.67%	-	94.83%	94.17%	93.33%	93.00%	-	98.67%	98.67%	98.67%	98.50%	-
				0.2	96.50%	95.67%	96.83%	97.50%	-	91.33%	88.00%	89.17%	89.00%	-	99.00%	98.17%	98.83%	99.00%	-
				0.3	96.83%	96.67%	97.00%	94.17%	-	86.17%	84.33%	85.67%	82.50%	-	98.67%	98.33%	98.33%	97.50%	-
			20%	0.1	98.50%	99.00%	99.17%	98.00%	-	94.33%	96.17%	93.50%	94.00%	-	98.83%	99.33%	99.17%	99.67%	-
				0.2	96.50%	97.67%	96.17%	97.83%	-	91.67%	92.83%	91.67%	90.67%	-	99.83%	99.50%	98.67%	98.50%	-
				0.3	94.33%	94.33%	94.67%	94.00%	-	87.33%	86.50%	87.17%	85.17%	-	99.00%	99.17%	98.67%	99.33%	-
			30%	0.1	98.17%	98.67%	99.00%	98.50%	-	94.50%	95.33%	93.67%	91.50%	-	99.50%	99.50%	98.83%	99.50%	-
				0.2	96.00%	96.33%	94.33%	93.83%	-	93.83%	92.67%	91.50%	91.83%	-	99.17%	99.33%	99.67%	99.83%	-
				0.3	94.50%	95.67%	91.50%	95.67%	-	92.00%	92.67%	90.83%	90.67%	-	99.83%	99.17%	99.83%	99.67%	-
0	0	5	10%	0.1	97.67%	97.83%	98.17%	97.50%	98.00%	96.67%	95.33%	93.17%	94.33%	91.83%	97.83%	98.67%	98.33%	98.33%	97.33%
				0.2	96.67%	97.67%	98.17%	96.00%	97.50%	92.83%	88.17%	88.83%	89.33%	87.67%	98.50%	98.50%	97.50%	97.67%	98.50%
				0.3	95.33%	96.67%	94.00%	96.17%	97.17%	79.67%	82.17%	74.50%	77.50%	84.83%	97.67%	98.33%	97.83%	98.00%	97.83%
			20%	0.1	98.50%	99.00%	99.00%	98.00%	98.33%	94.83%	94.17%	95.33%	93.00%	94.00%	98.33%	98.67%	98.50%	99.33%	99.17%
				0.2	98.00%	95.17%	97.50%	96.33%	97.17%	88.67%	87.67%	87.50%	85.17%	86.33%	98.17%	97.83%	98.83%	98.33%	99.00%
				0.3	94.50%	96.83%	94.00%	97.83%	97.33%	84.83%	87.50%	86.17%	84.67%	85.50%	98.83%	98.33%	99.17%	99.00%	99.00%
			30%	0.1	98.83%	98.33%	99.83%	98.17%	98.67%	92.50%	95.33%	93.50%	91.50%	92.17%	99.17%	98.17%	99.50%	99.00%	98.33%
				0.2	94.00%	95.17%	95.50%	95.83%	96.50%	91.83%	90.00%	89.50%	90.00%	92.17%	98.67%	99.17%	98.67%	98.50%	98.83%
				0.3	92.83%	93.33%	94.17%	90.67%	94.67%	85.00%	89.00%	87.00%	83.00%	86.83%	98.67%	99.50%	99.17%	98.83%	99.00%

Chapter 6

Conclusions

The core of this research is the examination of integrated network design and scheduling problems. We formally define this class of problems and demonstrate how they can be used for many application areas, with a specific focus on infrastructure restoration. INDS problems involve three main decisions: (i) the selection of arcs to make operational in the network, (ii) the allocation of machines to process the selected arcs, and (iii) the sequencing of the selected arcs on the machines.

We establish the complexity of many INDS problems to be *NP*-Hard. We create a dispatching rule framework that iteratively calls a selection routine and scheduling routine which can be easily customized to any INDS problem. The basis of the selection routine is selecting a *set* of tasks that interact within the network and maximizes the ratio of improvement in the network performance to the objective function to time necessary for processing the set. The specific set of tasks is determined by examining the optimality conditions of the network performance used for the INDS problem. The scheduling routine uses combinations of existing scheduling rules, such as the longest processing time rule and earliest release date rule.

INDS problems are extended to incorporate specific network design and novel scheduling considerations. Six performance metrics are used to evaluate the network performance, in combination with two different objective functions. Extensions with regards to scheduling are considered by incorporating arc release dates, flexible release dates, and multi-function machines. Further, we extend the problem to incorporate an online optimization environment with regards to release dates. In this environment, we assume that not all information is known at the start of the problem. Instead, information about when arcs are released and available for processing arrive throughout the horizon of the problem. The dispatching rule framework is used and customized to incorporate all of these different extensions of INDS problems.

INDS problems are computationally tested to devise restoration plans for damaged realistic power networks. We determine that the dispatching rules are a robust solution method that arrive at near optimal solutions quickly for many problem instances. Computational tests focusing on the scheduling extensions of INDS problems and the online optimization environment indicate that the dispatching rules arrive at high quality, near optimal solutions in a matter of seconds. Different work group compositions are considered and we observe a phenomenon of diminishing return with more specialized work groups. Benefit is seen when incorporating the same number of traditional and specialized work groups. Work groups that are multi-function lead to the greatest benefit.

In conclusion, we have thoroughly examined INDS problems in many avenues. Areas of future work include considering INDS with three types of machines: traditional, specialized, and multi-function. The online optimization environment should be extended to incorporate lack of information about other aspects of the problem. Interdependencies between infrastructure systems should also be considered. For example, the availability of the road network needed for traveling between sequential restoration tasks, should impact the sequencing of selected restoration tasks.

Funding Statement: This work was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories, a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energys National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

[1] R. Agarwal and O. Ergun. Ship scheduling and network design for cargo routing in liner shipping. *Transportation Science*, 42(2):175–196, May 2008.

[2] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs, NJ, 1993.

[3] E.J. Anderson and C.N. Potts. Online scheduling of a single machine to minimize total weighted completion time. *Mathematics of Operations Research*, 29(3):686–697, Aug. 2004.

[4] C.C. Ang. Optimized recovery of damaged electrical power grids. Master's thesis, Naval Postgraduate School, Monterey, CA, Mar. 2006.

[5] I. Averbakh. Emergency path restoration problems. *Discrete Optimization*, 9(1):58–64, 2012.

[6] I. Averbakh and J. Pereira. The flowtime network construction problem. *IIE Transactions*, 44(8):681–694, Jun. 2012.

[7] N. Baumann and M. Skutella. Earliest arrival flows with multiple sources. *Mathematics of Operations Research*, 34(2):499–512, May 2009.

[8] D. Bienstock and A. Verma. The $N - k$ problem in power grids: New models, formulations, and numerical experiments. *SIAM Journal on Optimization*, 20(5):2352–2380, Apr. 2010.

[9] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, New York, New York, first edition, 1998.

[10] B. Cavdaroglu, E. Hammel, J.E. Mitchell, T.C. Sharkey, and W.A. Wallace. Integrating restoration and scheduling decisions for disrupted interdependent infrastructure systems. *Annals of Operations Research*, 203(1):279–294, Mar. 2013.

[11] S.E. Chang. Infrastructure resilience to disasters. *The Bridge*, 39(4):36–41, 2009.

[12] S.E. Chang, T.L. McDaniels, J. Mikawoz, and K. Peterson. Infrastructure failure interdependencies in extreme events: Power outage consequences in the 1998 ice storm. *Natural Hazards*, 41(2):337–358, 2007.

[13] M.C. Chou, M. Queyranne, and D. Simchi-Levi. The asymptotic performance ratio of an on-line algorithm for uniform parallel machine scheduling with release dates. *Mathematical Programming*, 106(1):137–157, Mar. 2006.

[14] C. Clastres. Smart grids: Another step towards competition, energy security and climate change objectives. *Energy Policy*, 39(9):5399–5408, Sep. 2011.

[15] J.R. Correa and M.R. Wagner. LP-based online scheduling: From single to parallel machines. *Mathematical Programming*, 119(1):109–136, Feb. 2009.

[16] T.G. Crainic. Service network design in freight transportation. *European Journal of Operational Research*, 122(2):272–288, Apr. 2000.

[17] M.S. Daskin. *Network and Discrete Location: Models, Algorithms, and Applications*. John Wiley and Sons, Inc., New York, NY, 1995.

[18] R. DeBlasio and C. Tom. Standards for the smart grid. In *Proceedings of the 2008 IEEE Energy 2030 Conference*, pages 1–7, Atlanta, GA, Nov. 2008.

[19] Z. Drezner. *Facility Location: A Survey of Applications and Methods*. Springer Verlag, 1995.

[20] D. Dubois, H. Fargier, and P. Fortemps. Fuzzy scheduling: Modelling flexible constraints vs. coping with incomplete knowledge. *European Journal of Operational Research*, 147(2):231–252, Jun. 2003.

[21] L. Dueñas-Osorio, J.I. Craig, B.J. Goodno, and A. Bostrom. Interdependent response to networked systems. *Journal of Infrastructure Systems*, 13(3):185–194, Sep. 2007.

[22] T. Elgindy, A. T. Ernst, M. Baxter, M.W.P. Savelsbergh, and T. Kalinowski. Incremental network design with shortest paths. Technical report, CSIRO Mathematics Informatics and Statistics, Australia, 2013. `http://www.optimization-online.org/DB\_FILE/2013/01/3752.pdf` (Date Last Accessed, Jun. 12, 2013).

[23] J. Ellis, D. Fisher, T. Longstaff, L. Pesante, and R. Pethia. Report to President's commission on critical infrastructure protection. Technical Report CMU/SEI-97-SR-003, CERT Coordination Center, Carnegie Mellon University, Pittsburgh, PA, Jan. 1997.

[24] O. Ergun, G. Karakus, P. Keskinocak, J. Swann, and M. Villarreal. Operations research to improve disaster supply chain management. In J. J. Cochran, editor, *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley and Sons, Inc., New York, NY, 2010.

[25] H. Farhangi. The path of the smart grid. *IEEE Power and Energy Magazine*, 8(1):18–28, Jan./Feb. 2010.

[26] A. Fiat and G. Woeginger. *Online Algorithms: The State of the Art*. Springer-Verlag, New York, NY, 1998.

[27] M. Glinkowski, J. Hou, and G. Rackliffe. Advances in wind energy technologies in the context of smart grid. *Proceedings of the IEEE*, 99(6):1083–1097, Jun. 2011.

[28] J. Gong, E.E. Lee, J.E. Mitchell, and W.A. Wallace. Logic-based multi-objective optimization for restoration planning. In W. Chaovalitwongse, K.C. Furman, and P.M. Pardalos, editors, *Optimization and Logistics Challenges in the Enterprise*, chapter 11, pages 305–324. Springer, New York, NY, 2009.

[29] R.L. Graham. Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*, 17(2):416–429, Mar. 1969.

[30] R.L. Graham, E.L. Lawler, J.K. Lenstra, and A.H.G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: A survey. *Annals of Discrete Mathematics*, 5(2):287–326, 1979.

[31] S. Guha, A. Moss, J.S. Naor, and B. Schieber. Efficient recovery from power outage. In *Proceedings of the thirty-first annual ACM Symposium on Theory of Computing*, pages 574–582, May 1999.

[32] V. Guihaire and J.K. Hao. Transit network design and scheduling: A global review. *Transportation Research Part A: Policy and Practice*, 42(10):1251–1273, Dec. 2008.

[33] Y.Y. Haimes, B.M. Horowitz, J.H. Lambert, J.R. Santos, K.G. Crowther, and C. Lian. Inoperability input-output model for interdependent infrastructure sectors. II: Case studies. *Journal of Infrastructure Systems*, 11(2):80–92, 2005.

[34] Y.Y. Haimes, B.M. Horowitz, J.H. Lambert, J.R. Santos, C. Lian, and K.G. Crowther. Inoperability input-output model for interdependent infrastructure sectors. I: Theory and methodology. *Journal of Infrastructure Systems*, 11(2):67–79, 2005.

[35] Y.Y. Haimes and P. Jiang. Leontief-based model of risk in complex interconnected infrastructures. *Journal of Infrastructure Systems*, 7(1):1–12, Mar. 2001.

[36] L.A. Hall, A.S. Schulz, D.B. Shmoys, and J. Wein. Scheduling to minimize average completion time: Off-line and on-line approximation algorithms. *Mathematics of Operations Research*, 22(3):513–544, Aug. 1997.

[37] J.A. Hoogeveen and A.P.A. Vestjens. Optimal on-line algorithms for single-machine scheduling. In *Proceedings of the Fifth Conference On Integer Programming and Combinatorial Optimization*, pages 404–414, 1996.

[38] P. Jaillet and M. Stafford. Online searching. *Operations Research*, 49(4):501–515, Jul.-Aug. 2001.

[39] P. Jaillet and M.R. Wagner. Online routing problems: Value of advanced information as improved competitive ratios. *Transportation Science*, 40(2):200–210, May 2006.

[40] P. Jaillet and M.R. Wagner. Generalized online routing: New competitive ratios, resource augmentation, and asymptotic analyses. *Operations Research*, 56(3):745–757, May 2008.

[41] P. Jaillet and M.R. Wagner. Almost sure asymptotic optimality for online routing and machine scheduling problems. *Networks*, 55(1):2–12, Jan. 2010.

[42] P. Jaillet and M.R. Wagner. Online optimization-An introduction. In J. J. Hasenbein, editor, *TutORials in Operations Research, Risk and Optimization in an Uncertain World*, INFORMS TutORials in Operations Research Series, chapter 6, pages 142–152. INFORMS, Austin, TX, Nov. 2010.

[43] G. Kalafatas and S. Peeta. Planning for evacuation: Insights from an efficient network design model. *Journal of Infrastructure Systems*, 15(1):21–30, Mar. 2009.

[44] A.R. Karlin, M.S. Manasse, L. Rudolph, and D.D. Sleator. Competitive snoopy caching. *Algorithmica*, 3:79–119, Nov. 1988.

[45] M.F. Lai and H.K. Lo. Ferry service network design: Optimal fleet size, routing, and scheduling. *Transportation Research Part A: Policy and Practice*, 38(4):305–328, May 2004.

[46] R.C. Larson, M.D. Metzger, and M.F. Cahn. Responding to emergencies: Lessons learned and the need for analysis. *Interfaces*, 36(6):486–501, Nov./Dec. 2006.

[47] E.E. Lee, J.E. Mitchell, and W.A. Wallace. Restoration of services in interdependent infrastructure systems: A network flows approach. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1303–1317, Nov. 2007.

[48] M. Liserre, T. Sauter, and J.Y. Hung. Future energy systems: Integrating renewable energy sources into the smart power grid through industrial electronics. *IEEE Industrial Electronics Magazine*, 4(1):18–37, Mar. 2010.

[49] R.G. Little. Controlling cascading failure: Understanding the vulnerabilities of interconnected infrastructures. *Journal of Urban Technology*, 9(1):109–123, 2002.

[50] R. Loggins, J. Taylor, S. Brehm, and W.A. Wallace. *CLARC County*. Rensselaer Polytechnic Institute, 2013.

[51] A. Mahmood, M. Aamir, and M.I. Anis. Design and implementation of AMR smart grid system. In *Proceedings of the 2008 IEEE Electrical Power and Energy Conference*, pages 1–6, Oct. 2008.

[52] T.C. Matisziw, A.T. Murray, and T.H. Grubesic. Exploring the vulnerability of network infrastructure to disruption. *Annals of Regional Science*, 43(2):307–321, 2009.

[53] T.C. Matisziw, A.T Murray, and T.H. Grubesic. Strategic network restoration. *Networks and Spatial Economics*, 10(3):345–361, 2010.

[54] N. Megow and A.S. Schulz. On-line scheduling to minimize average completion time revisited. *Operations Research Letters*, 32(5):485–490, Sep. 2004.

[55] D. Mendonça and W.A. Wallace. Impacts of the 2001 World Trade Center attack on New York City critical infrastructures. *Journal of Infrastructure Systems*, 12(4):260–270, Dec. 2006.

[56] J.A. Momoh. Smart grid design for efficient and flexible power networks operation and control. In *Proceedings of the 2009 IEEE/PES Power Systems Conference and Exposition*, pages 1–8, Seattle, WA, 2009.

[57] K. Moslehi and R. Kumar. A reliability perspective of the smart grid. *IEEE Transactions on Smart Grid*, 1(1):57–64, Jun. 2010.

[58] G. Mulder, F. De Ridder, and D. Six. Electricity storage for grid-connected household dwellings with PV panels. *Solar Energy*, 84(7):1284–1293, Jul. 2010.

[59] A.T. Murray, T.C. Matisziw, and T.H. Grubesic. Critical network infrastructure analysis: Interdiction and system flow. *Journal of Geographical Systems*, 9(2):103–117, Jun. 2007.

[60] A. Nagurney and Q. Qiang. A network efficiency measure with application to critical infrastructure networks. *Journal of Global Optimization*, 40(1-3):261–275, 2008.

[61] New York Independent System Operator. *Real-Time Actual Load*, 2012. `http://www.nyiso.com/public/markets_operations/market_data/custom_report/index.jsp?report=rt_actual_load` (Date Last Accessed, Jun. 12, 2013).

[62] S.G. Nurre, B. Cavdaroglu, J.E. Mitchell, T.C. Sharkey, and W.A. Wallace. Restoring infrastructure systems: An integrated network design and scheduling problem. *European Journal of Operational Research*, 223(3):794–806, Dec. 2012.

[63] S.G. Nurre and T.C. Sharkey. Integrated network design and scheduling problems with parallel identical machines: Complexity results and dispatching rules. Technical report, Rensselaer Polytechnic Instittue, 2013. Submitted for publication.

[64] T.D. O'Rourke. Critical infrastructure, interdependencies, and resilience. *The Bridge*, 37(1):22–29, 2007.

[65] M.L. Pinedo. *Scheduling: Theory, Algorithms, and Systems*. Springer, New York, NY, fourth edition, 2012.

[66] S.M. Rinaldi, J.P. Peerenboom, and T.K. Kelly. Identifying, understanding, and analyzing critical infrastructure interdependencies. *IEEE Control Systems Magazine*, 21(6):11–25, Dec. 2001.

[67] T.C. Sharkey, S.G. Nurre, H. Nguyen, J.H. Chow, J.E. Mitchell, and W.A. Wallace. Identification and clasification of restoration interdependencies in the wake of Hurricane Sandy. Rensselaer Polytechnic Institute, 2013.

[68] D.D. Sleator and R.E. Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28(2):202–208, Feb. 1985.

[69] W.E. Smith. Various optimizers for single-stage production. *Naval Research Logistics Quarterly*, 3(1-2):59–66, Jun. 1956.

[70] A.D. Sonmez and G.J. Lim. A decomposition approach for facility location and relocation problem with demand change under uncertainty. *European Journal of Operational Research*, 218(2):327–338, Apr. 2012.

[71] M. Stilp, A. Carbajal, O. Ergun, P. Keskinocak, and M. Villarreal. Debris clearance operations. Technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 2012.

[72] L. Stougie and A.P.A. Vestjens. Randomized algorithms for on-line scheduling problems: How low can't you go? *Operations Research Letters*, 30(2):89–96, Apr. 2002.

[73] K. Tierney and M. Bruneau. Conceptualizing and measuring resilience: A key to disaster loss reduction. *TR News*, 250:14–17, May-Jun. 2007.

[74] W.A. Wallace, D. Mendonça, E. Lee, J. Mitchell, and J. Chow. Managing disruptions to critical interdependent infrastructures in the context of the 2001 World Trade Center attack. In M.F. Myers, editor, *Beyond September 11: An account of post-disaster research*, pages 165–198. Natural Hazards Research and Applications Information Center, University of Colorado, Boulder, CO, 2003.

[75] N. Xu, S.D. Guikema, R.A. Davidson, L.K. Nozick, Z. Çağnan, and K. Vaziri. Optimizing scheduling of post-earthquake electric power restoration tasks. *Earthquake Engineering and Structural Dynamics*, 36(2):265–284, Feb. 2007.

DISTRIBUTION:

1	MS 0899	Technical Library, 9536 (electronic copy)
1	MS 0359	D. Chavez, LDRD Office, 1911

Sandia National Laboratories