COLLABORATIVE RESPONSE TO DISRUPTION PROPAGATION (CRDP)

by

Win P. V. Nguyen

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

School of Industrial Engineering

West Lafayette, Indiana

May 2020

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Shimon Y. Nof, Chair
School of Industrial Engineering

Dr. Mario Ventresca
School of Industrial Engineering

Dr. Seokcheon Lee
School of Industrial Engineering

Dr. David Cappelleri
School of Mechanical Engineering

Approved by:
Dr. Abhijit Deshmukh

For my beloved family, treasured friends, respected mentors, and cherished students.

ACKNOWLEDGMENTS

My deepest gratitude:

For my respected advisor, Professor Shimon Y. Nof, for taking the unprepared me and guiding me on this challenging yet meaningful journey, and for being supportive of me through many obstacles.

For my respected mentor, Professor Arden L. Bement Jr., for inspiring me to overcome myself, for his tremendous patience, and for teaching me many of his great wisdom.

For my respected mentor, Professor Duane D. Dunlap, for his sympathy, support, and guidance throughout my Ph.D. journey.

For my respected committee members:

For Professor Mario Ventresca, for his sympathy to students, for his enthusiasm for teaching, and for being a great example of a young faculty.

For Professor Seokcheon Lee, for his support and the many classes I learned from him.

For Professor David Cappelleri, for his insights and support towards this dissertation.

For my treasured friends:

For Tyler P. Throop, my best friend, for his continuous support through my difficult times and for the many things I learned with and from him.

For Professor Mary E. Johnson and her husband Drew Casani, for their great advice and support on many things: research, career, and life.

For Professor Phillip A. Sanger and his wife Terri Sanger, for being great friends and for his support during my time at Purdue University.

For Ashwin S. Nair and Oak Puwadol Dusadeeringsikul, who worked together with me on many research projects and assignments throughout this journey.

For Dr. Victor Chenyu Huang and his wife Olivia Ciying Yang, for being great friends to my wife and I, and for their continuous support.

For Dr. Mohsen Moghaddam, for being a great example of a Ph.D. researcher and for providing great inspiration for my research and dissertation. To Dr. Hao Zhong and Dr. Rodrigo Reyes Levalle for their Ph.D. research, upon which I built this foundation.

For Maitreya Sreeram, Billy Xiang He, Dr. Radhika Bhargava, Jawahar Gogigeni, Praditya Ajidarma, my friends and colleagues at the PRISM Center.

For my beloved family: To my beloved wife, for being with me and for supporting me throughout this challenging journey. To my parents, and brothers for supporting me in many ways.

For my cherished students, for respecting my words (most of them, I sincerely hope), for providing great reasons to keep improving myself, and for allowing me to experience the joy of teaching and mentoring.

TABLE OF CONTENTS

LIST OF TABLES 9
LIST OF FIGURES 11
LIST OF ABBREVIATIONS 13
GLOSSARY 15
LIST OF SYMBOLS 17
ABSTRACT 25
CHAPTER 1. INTRODUCTION 27
1.1 Motivations 27
1.2 Research Problem 29
1.3 Research Questions 30
1.4 General Definitions and Assumptions 31
1.5 Naming, Typesetting and Conventions 34
1.6 Dissertation Structure 36
CHAPTER 2. LITERATURE REVIEW 38
2.1 The RDP Problem in Different Domains 38
2.1.1 The fire spreading problem 38
2.1.2 The infectious plant disease problem 41
2.1.3 The propagating computer malware problem 43
2.1.4 The supply network disruption problem 46
2.2 Formulation of RDP problems 49
2.3 Collaborative Control Theory 55
2.4 Concluding Remarks 58
CHAPTER 3. METHODOLOGY – THE CRDP FRAMEWORK AND THE CLOC DESIGN AND CONTROL PRINCIPLE 59
3.1 The CRDP Framework 60
3.1.1 The CRDP components 61
3.1.2 The CRDP interactions 67
3.1.3 The CRDP decision spaces 70
3.1.4 The CRDP system performance metrics 71

3.2 The Covering Lines of Collaboration (CLOC) Principle 73
3.2.1 The first CLOC guideline – network modeling of disruption propagation 74
3.2.2 The second CLOC guideline – restraining disruption propagation 78
3.2.3 The third CLOC guideline – collaboration between response decisions to ensure coverage 80
3.3 Case Studies – A Synopsis 83
CHAPTER 4. CASE 1 – COLLABORATIVE DETECTION OF UNKNOWN DISRUPTION PROPAGATION 87
4.1 CDUD Description 87
4.2 CDUD Formulation 90
4.3 CDUD Analytics and Protocols 96
4.3.1 CDUD analytics 98
4.3.2 CDUD protocols 101
4.4 Numerical Experiments and Results 105
4.4.1 Comparison by CDUD protocols 106
4.4.2 Comparison by CDUD protocols and network types 108
4.4.3 Comparison by CDUD protocols and response/disruption scenarios 111
4.4.4 Set of experiments on an enterprise's internal email network 113
4.5 Concluding Remarks 118
CHAPTER 5. CASE 2 – COLLABORATIVE STRATEGIC PREVENTION OF DISRUPTION PROPAGATION 119
5.1 CSPD Description 119
5.2 CSPD Formulation 122
5.3 CSPD Analytics and Protocols 127
5.3.1 CSPD analytics 129
5.3.2 CSPD protocols 131
5.4 Numerical Experiments and Results 135
5.4.1 Comparison by CSPD protocols 136
5.4.2 Comparison by CSPD protocols and network types 138
5.4.3 Comparison by CSPD protocol groups and response/disruption scenarios 142
5.4.4 Set of experiments on an enterprise's internal email network 144

5.5 Concluding Remarks 150
CHAPTER 6. CASE 3 – COLLABORATIVE TEAMING AND COORDINATION OF DYNAMIC REPAIR AGENTS 152
6.1 CTCD Description 152
6.2 CTCD Formulation 155
6.3 CTCD Analytics and Protocols 162
6.3.1 CTCD teaming decisions 163
6.3.2 CTCD dynamic coordination decisions 168
6.4 Numerical Experiments and Results 171
6.4.1 The first set of CTCD numerical experiments 172
6.4.2 The second set of CTCD numerical experiments 174
6.5 Concluding Remarks 178
CHAPTER 7. CONCLUSIONS 180
7.1 Summary of Design Recommendations 180
7.2 Summary of Original Contributions 184
7.3 Limitations and Future Research Directions 187
APPENDIX A. TIE/CRDP SOFTWARE 191
REFERENCES 197
VITA 215
PUBLICATIONS 218

LIST OF TABLES

Table 1.1. Summary of background examples with respect to definitions 33
Table 1.2. Mapping between Research Questions and dissertation structure 37
Table 2.1. Summary of literature on fire spreading 41
Table 2.2. Summary of literature on infectious plant diseases 43
Table 2.3. Summary of literature on propagating computer malware 46
Table 2.4. Summary of literature on supply network disruptions 49
Table 2.5. Summary of literature on the formulation of RDP problems 53
Table 2.6. Summary of literature on the formulation of RDP problems 54
Table 2.7. Summary of CCT research relevant to the RDP problem 58
Table 3.1. Examples of response mechanisms 63
Table 3.2. Summary of CRDP components 67
Table 3.3. Summary of CRDP interactions 70
Table 3.4. Summary of CRDP formulation categories 72
Table 3.5. Summary of CLOC guidelines 83
Table 3.6. Comparison of formulations of the seven case studies 86
Table 4.1. Summary of CDUD description 89
Table 4.2. Entities and attributes of the CDUD model 91
Table 4.3. System performance metrics of the CDUD model 92
Table 4.4. Simulation pseudocode of the CDUD model 94
Table 4.5. Summary of the CDUD analytics 101
Table 4.6. Summary of the CDUD protocols 104
Table 4.7. CDUD experiment results grouped by CDUD protocols 107
Table 4.8. CDUD experiment results grouped by network types 110
Table 4.9. CDUD experiment results grouped by response/disruption scenarios 112
Table 4.10. CDUD email network experiment results grouped by CDUD protocols 115
Table 4.11. CDUD email network experiment results grouped by response/disruption scenarios 117
Table 5.1. Summary of CSPD description 121

Table 5.2. Entities and attributes of the CSPD model 122
Table 5.3. System performance metrics of the CSPD model 124
Table 5.4. Simulation pseudocode of the CSPD model 126
Table 5.5. Summary of the CSPD analytics 131
Table 5.6. Protocol pseudocode of the CSPD model 134
Table 5.7. Summary of the CSPD protocols 135
Table 5.8. CSPD experiment results grouped by CSPD protocols 137
Table 5.9. CSPD experiment results grouped by network types 141
Table 5.10. CSPD experiment results grouped by response/disruption scenarios and CSPD protocol groups 144
Table 5.11. CSPD email network experiment results grouped by CSPD protocols 146
Table 5.12. CSPD email network experiment results grouped by response/disruption scenarios and CSPD protocol groups 150
Table 6.1. Summary of CTCD description 154
Table 6.2. Entities and attributes of the CTCD model 155
Table 6.3. System performance metrics of the CTCD model 158
Table 6.4. Discrete events in CTCD model 159
Table 6.5. Simulation pseudocode of the CTCD model 161
Table 6.6. Summary of the CTCD teaming decisions analytics and protocols 168
Table 6.7. Summary of the CTCD coordination analytics and protocols 171
Table 6.8. CTCD first set of experiments – disruption scenarios 172
Table 6.9. Comparison table of strategic compatibility levels 176
Table 6.10. Comparison table of CTCD teaming protocols and CTCD coordination protocols 177
Table 7.1. Summary of CRDP applications and modeling adaptations 184
Table 7.2. Mapping between Research Questions and concepts 187

LIST OF FIGURES

Figure 1.1. Disruption propagation examples 32

Figure 3.1. The CRDP framework 60

Figure 3.2. Illustration of client system examples 62

Figure 3.3. Illustration of disruption examples 64

Figure 3.4. Brief RDP example illustration 66

Figure 3.5. Network modeling of disruption propagation examples 76

Figure 3.6. Different centrality measures: degree, closeness, and in-between 78

Figure 3.7. Disruption propagation restraining effect 80

Figure 3.8. Example of the CLOC sub-guideline on coverage 82

Figure 4.1. CDUD example 93

Figure 4.2. Neighboring disruption propagation example with no response 97

Figure 4.3. CDUD experiment results grouped by CDUD protocols with 95% confidence interval bars 106

Figure 4.4. CDUD experiment results grouped by CDUD protocols and network types, with 95% confidence interval bars 109

Figure 4.5. CDUD experiment results grouped by CDUD protocols and response/disruption scenarios, with 95% confidence interval bars 111

Figure 4.6. CDUD email network experiment results grouped by CDUD protocols with 95% confidence interval bars 114

Figure 4.7. CDUD email network experiment results grouped by CDUD protocols with 95% confidence interval bars 116

Figure 5.1. CSPD example 125

Figure 5.2. CSPD experiment results grouped by CSPD protocols with 95% confidence interval bars 137

Figure 5.3. CSPD experiment results grouped by CSPD protocols and network types GO and BA, with 95% confidence interval bars 139

Figure 5.4. CSPD experiment results grouped by CSPD protocols and network types ER and WS, with 95% confidence interval bars 140

Figure 5.5. CSPD experiment results grouped by CSPD protocol groups and response/disruption scenarios with 10 initial disruptions, with 95% confidence interval bars 142

Figure 5.6. CSPD experiment results grouped by CSPD protocol groups and response/disruption scenarios with 20 initial disruptions, with 95% confidence interval bars..... 143

Figure 5.7. CSPD email network experiment results grouped by CSPD protocols with 95% confidence interval bars 146

Figure 5.8. CSPD email network experiment results grouped by CSPD protocol groups and response/disruption scenarios with 10 initial disruptions, with 95% confidence interval bars .. 148

Figure 5.9. CSPD email network experiment results grouped by CSPD protocol groups and response/disruption scenarios with 10 initial disruptions, with 95% confidence interval bars .. 149

Figure 6.1. CTCD disruption propagation example..... 159

Figure 6.2. CTCD response mechanisms example 160

Figure 6.3. Example of NRDP(n) and NMDP(n) 166

Figure 6.4. MNDP protocol illustration..... 169

Figure 6.5 CCTD first set of experiments grouped by CCTD coordination protocols and response/disruption scenarios, with 95% confidence interval bars 173

Figure 6.6. CCTD experiment results grouped by CCTD teaming protocols, with 95% confidence interval bars..... 176

Figure 6.7. Comparison chart of CTCD teaming protocols and CTCD coordination protocols (with 95% confidence intervals)..... 177

LIST OF ABBREVIATIONS

		Page where first defined
ARS	Agricultural robotic systems	56
BA	Barabasi-Albert	105
BMP	Best Matching Protocol	56
CCT	Collaborative Control Theory	55
CDUD	Collaborative Detection of Unknown Disruption Propagation	84
CFT	Collaborative fault tolerance	56
CLOC	Covering Lines of Collaboration	25, 31
CPS	Cyber-physical systems	27
CRDP	Collaborative Response to Disruption Propagation	25, 29
CRP	Cooperation requirement planning	55
CSPD	Collaborative Strategic Prevention of Disruption Propagation	84
CTCD	Collaborative Teaming and Coordination of Dynamic Repair Agents	84
DLOC	Dynamic Lines of Collaboration	57
ELOCC	Emergent Lines of Collaboration and Command	57
EPCR	Error prevention and conflict resolution	55
ER	Erdos-Renyi	105
FCFS	First-come-first-serve	169
GD	Grid diagonal	105
GO	Grid orthogonal	105
MATW	Minimizing additional task workload	170
MDP	Maximum disruption propagation	92
MNDP	Minimizing neighboring disruption propagation	169
MPL	Maximum performance loss	92
RBT	Resilience by Teaming	56
RDP	Response to Disruption Propagation	25, 28

		Page where first defined
SPT	Shortest-processing-time	169
TIE	Teamwork Integration Evaluator	2626
TPL	Total performance loss	92
WS	Watts-Strogatz	105

GLOSSARY

This list briefly explains the important definitions and terminologies.

Detailed explanation in page

Response to disruption propagation (RDP) problem	A specific problem instance involving disruptions and their propagations negatively affecting a client system, and response mechanisms exist to mitigate or eliminate the disruptions.	28
Response to disruption propagation (RDP) model	A theoretical model that can simulate and/or emulate the entities, attributes, and events of a corresponding RDP problem.	33
Collaborative Response to Disruption Propagation (CRDP) framework	The unifying framework to characterize and categorize the important components, interactions, decisions, and metrics of RDP problems.	60
Client system	The client system consists of entities that are subjected to harmful disruptions and their propagation. The client system plays the role of the victim.	61
Response mechanisms	The response mechanisms consist of the entities that can strictly reduce or eliminate the existences and/or impacts of the disruptions and their propagation. The response mechanisms play the role of the rescuers/protectors.	62
Disruption propagation	The disruptions and their propagations consist of the entities that can strictly cause negative impacts and/or propagate on the client system. Disruption propagation plays the role of the aggressors/attackers.	63
CRDP interaction	Complex dynamics and relations that involve two or more CRDP components.	67
CRDP design decision	The decisions made off-line, cannot be changed during real-time.	70
CRDP control decision	The decisions made on-line can be changed during real-time.	71
CRDP system performance metrics	The optimization objective function(s) and/or binary (yes/no, true/false) goals.	71
Covering Lines of Collaboration (CLOC) principle	The collaborative control principle that guides and supports the analysis and decision-making process of the response mechanisms against disruption propagation.	73

		Detailed explanation in page
Analytics	(In the context of this work) The analysis/analyses of the state of the system and the modeling elements, and returns a set of quantifiable variables and/or conjectures that can guide the development of the protocols.	73
Protocol	(In the context of this work) The workflow pre-defined, agreed-upon, decision-making set(s) of rules, procedures, and possibly algorithms for multiple interacting agents.	73
Disruption propagation direction	The direction in which a disruption affecting one node can potentially and directly propagate to another node. This can be modeled as an edge.	75

LIST OF SYMBOLS

Naming conventions and variable typesetting are defined in Section 1.5.

		Page where first defined
Defined in CHAPTER 3, also applies to CHAPTERS 4-6		
$\boldsymbol{\mathcal{C}}$	Client system	61
$\boldsymbol{\mathcal{R}}$	Response mechanisms	62
$\boldsymbol{\mathcal{D}}$	Disruptions	63
$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$	Client-response interaction	68
$\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$	Response-disruption interaction	69
$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$	Disruption-client interaction	68
$\boldsymbol{\mathcal{S}}_{\#}$	Design decisions	70
$\boldsymbol{\mathcal{S}}_{\#}^{t}$	Control decisions	71
$\boldsymbol{\mathcal{M}}$	System performance metrics	71
$\boldsymbol{\mathcal{A}}$	Analytic	73
$\boldsymbol{\mathcal{P}}$	Protocol	73
n	A node, representing a component of the client system $\boldsymbol{\mathcal{C}}$	74
NL	Set of nodes of the client system $\boldsymbol{\mathcal{C}}$	74
e	An edge, representing a potential disruption propagation direction from one node to another.	75
EL	Set of edges	75
$e = (n_i, n_j)$	A directed/unidirectional edge, representing a potential disruption propagation direction from node n_i to node n_j.	75
$e = \{n_i, n_j\}$	An undirected/bidirectional edge, representing a potential disruption propagation direction from node n_i to node n_j and from node n_j to node n_i.	75
Defined in and only specific to CHAPTER 3		
$d(n)$	Denoting a disruption affecting node n	78

		Page where first defined
$\mathrm{DP}(n)$	Set of potential disruption propagation from node n	78
$\mathrm{EDPV}(e)$	Edge e's disruption potential value	77
$f(\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{R}})$	The total harmful impact on $\boldsymbol{\mathcal{C}}$ from the existence of $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{R}}$.	64
$\mathrm{OD}(n)$	Out-degree of node n	77
$\mathrm{NDPP}(n)$	Node n's disruption propagation potential value	77
$r(n)$	Denoting a response affecting node n	79
$W(e)$	Edge e's weight	77
Defined in and only specific to CHAPTER 4		
a	Agent	91
AL	Set of agents	91
$\mathrm{ASN}(a, t)$	Agent a's selected node to perform response activity at time t	91
DPID	Probability of initial disruption	91
$\boldsymbol{\mathcal{M}}$	Consists of $\boldsymbol{\mathcal{M}}_1, \boldsymbol{\mathcal{M}}_2, \boldsymbol{\mathcal{M}}_3$	92
$\boldsymbol{\mathcal{M}}_1$	Refers to TPL	92
$\boldsymbol{\mathcal{M}}_2$	Refers to MPL	92
$\boldsymbol{\mathcal{M}}_3$	Refers to MDP	92
MDP	Maximum disruption propagation	92
MPL	Maximum performance loss	92
$\mathrm{NDS}(n, t)$	Node n's disruption status at time t	91
$\mathrm{NOS}(n, t)$	Node n's observed status at time t	91
$\mathrm{NPNL}(n)$	Node n's set of preceding nodes	91
$\mathrm{NSNL}(n)$	Node n's set of succeeding nodes	91
$\mathrm{PL}(t)$	Performance loss at time t	92
t	Timestep	90
TPL	Total performance loss	92
CDUD Analytics and Protocols		

		Page where first defined
$\boldsymbol{\mathcal{A}}_1$	CDUD Analytic 1	98
$\boldsymbol{\mathcal{A}}_2$	CDUD Analytic 2	98
$\boldsymbol{\mathcal{A}}_3$	CDUD Analytic 3	99
$\boldsymbol{\mathcal{A}}_4$	CDUD Analytic 4	99
$\boldsymbol{\mathcal{A}}_5$	CDUD Analytic 5	100
$\boldsymbol{\mathcal{A}}_6$	CDUD Analytic 6	100
$D(n,t)$	Event of node n being disrupted at time t	96
$\mathrm{DEL}(t)$	The dynamic set of edges at time t	100
DIST	Distance matrix based on NL and EL	99
$\mathrm{DIST}(n_i, n_j)$	Shortest-path distance from n_i to n_j	99
k	Equivalent to DPID	96
$\mathrm{NAHC}(n)$	Node n's advanced harmonic centrality	103
$\mathrm{NHC}(n)$	Node n's harmonic centrality	102
$\mathrm{NIHC}(n)$	Node n's intermediate harmonic centrality	103
$O(n,t)$	Event of node n being observed at time t	97
$\mathrm{OL}(t)$	Set of observations at time t	100
$\boldsymbol{\mathcal{P}}_1$	CDUD protocol 1: Random allocation protocol	101
$\boldsymbol{\mathcal{P}}_2$	CDUD protocol 2: Basic degree centrality allocation protocol	102
$\boldsymbol{\mathcal{P}}_3$	CDUD protocol 3: Basic harmonic centrality allocation protocol	102
$\boldsymbol{\mathcal{P}}_4$	CDUD protocol 4: Basic expanded centrality allocation protocol	102
$\boldsymbol{\mathcal{P}}_5$	CDUD protocol 5: Intermediate degree centrality allocation protocol	102
$\boldsymbol{\mathcal{P}}_6$	CDUD protocol 6: Intermediate harmonic centrality allocation protocol	102
$\boldsymbol{\mathcal{P}}_7$	CDUD protocol 7: Intermediate multi-order degree centrality allocation protocol	103
$\boldsymbol{\mathcal{P}}_8$	CDUD protocol 8: Advanced degree centrality allocation protocol	103

		Page where first defined
$\boldsymbol{\mathcal{P}}_9$	CDUD protocol 9: Advanced harmonic centrality allocation protocol	103
$\boldsymbol{\mathcal{P}}_{10}$	CDUD protocol 10: Advanced multi-order degree centrality allocation protocol	103
$P(n,t)$	Probability of node $D(n,t)$	96
$\mathrm{SP}(n_i, n_j)$	Shortest path from n_i to n_j	99
Defined in and only specific to CHAPTER 5		
a	Strategic allocation	122
AL	Set of strategic allocations	122
APL(t)	Accumulative performance loss from time 0 to time t	124
APP	Strategic allocation primary protection amount	122
$\mathrm{ASN}(a)$	Strategic allocation a's selected node to protect	123
ASP	Strategic allocation secondary protection amount	122
DCD	Disruption-client diameter	122
DL	Set of disruptions	122122
$\boldsymbol{\mathcal{M}}$	CSPD system performance metrics, consists of $\boldsymbol{\mathcal{M}}_1, \boldsymbol{\mathcal{M}}_2, \boldsymbol{\mathcal{M}}_3, \boldsymbol{\mathcal{M}}_4$	124
$\boldsymbol{\mathcal{M}}_1$	Refers to TPL_1	124
$\boldsymbol{\mathcal{M}}_2$	Refers to TPL_2	124
$\boldsymbol{\mathcal{M}}_3$	Refers to TPL_3	124
$\boldsymbol{\mathcal{M}}_4$	Refers to TPL_4	124
$\mathrm{NDS}(n,t)$	Node n's disruption status at time t	123
$\mathrm{NNL}(n)$	Node n's set of neighboring nodes	123
$\mathrm{NPS}(n)$	Node n's protection status	123
$\mathrm{PL}(t)$	Performance loss at time t	124
t	Timestep	
TPL_1	Total performance loss at one-fourth of the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter	124
TPL_2	Total performance loss at one-half of the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter	124

		Page where first defined
TPL_3	Total performance loss at three-fourth of the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter.	124
TPL_4	Total performance loss at maximum $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter.	124
CSPD Analytics and Protocols		
$\boldsymbol{\mathcal{A}}_1$	CSPD analytic 1	129
$\boldsymbol{\mathcal{A}}_2$	CSPD analytic 2	130
$\boldsymbol{\mathcal{A}}_3$	CSPD analytic 3	131
$\boldsymbol{\mathcal{A}}_4$	CSPD analytic 4	131
DIST	Distance matrix	128
$\mathrm{DIST}(n_i, n_j)$	Shortest-path distance between n_i and n_j	128
$\mathrm{NHC}(n)$	Node n's harmonic centrality	130
$\mathrm{NHCI}(n)$	Node n's harmonic-based coverage index	133
$\boldsymbol{\mathcal{P}}_1$	CSPD protocol 1: Random allocation protocol	132
$\boldsymbol{\mathcal{P}}_2$	CSPD protocol 2: Degree centrality allocation protocol	132
$\boldsymbol{\mathcal{P}}_3$	CSPD protocol 3: Harmonic centrality allocation protocol	132
$\boldsymbol{\mathcal{P}}_4$	CSPD protocol 4: CLOC – local coverage allocation protocol	132
$\boldsymbol{\mathcal{P}}_5$	CSPD protocol 5: CLOC – harmonic coverage allocation protocol	133
$\boldsymbol{\mathcal{P}}_6$	CSPD protocol 6: CLOC – global coverage allocation protocol	133
$\mathrm{PAL}(n_i, n_j)$	Set of paths between n_i and n_j	129
$\mathrm{PATH}(n_i, n_j)$	A path between n_i and n_j	128
$\mathrm{RPA}(n)$	Redundant prevention resources allocated to node n	131
TPM	Temporary performance metric value	134
Defined in and only specific to CHAPTER 6		
a	Repair agent	155
$\mathrm{ABS}(a, t)$	Agent a's busy status at time t	156

		Page where first defined
$\mathrm{AERN}(t, a, n)$	Agent ends responding node event	159
AL	A response team, which is a set of repair agents	155
d	Disruption	155
DL	Set of disruptions	155
$\mathrm{EDPS}(e, t)$	Edge e's disruption propagation status	156
$\mathrm{EDPT}(e)$	Edge e's disruption propagation time	156
$\boldsymbol{\mathcal{M}}$	CTCD system performance metrics, consists of $\boldsymbol{\mathcal{M}}_1, \boldsymbol{\mathcal{M}}_2, \boldsymbol{\mathcal{M}}_3, \boldsymbol{\mathcal{M}}_4$	158
$\boldsymbol{\mathcal{M}}_1$	Refers to RF	158
$\boldsymbol{\mathcal{M}}_2$	Refers to RT	158
$\boldsymbol{\mathcal{M}}_3$	Refers to TPL	158
$\boldsymbol{\mathcal{M}}_4$	Refers to MDPF	158
MDPF	Maximum disruption propagation fraction	158
$\mathrm{NAA}(n, t)$	Node n's assigned agent at time t	156
$\mathrm{NDP}(t, e)$	Node disruption propagates event	159
$\mathrm{NDS}(n, t)$	Node n's disruption status at time t	155
$\mathrm{NLDT}(n, t)$	Node n's latest disrupted time	156
$\mathrm{NPEL}(n)$	Node n's set of incoming/preceding edges	156
$\mathrm{NSEL}(n)$	Node n's set of outgoing/succeeding edges	156
$\mathrm{PL}(t)$	Performance loss at time t	158
$\mathrm{RRM}(a, n)$	Response requirement matrix	155
RT	Recovery time	158
SRT	Selected response team	156
TL	Set of response teams	155
TPL	Total performance loss	158
CTCD Analytics and Protocols		
$\boldsymbol{\mathcal{A}}_1$	CTCD analytic 1: Neighboring disruption analytic $\mathrm{NNDA}(n)$	163

		Page where first defined
$\boldsymbol{\mathcal{A}}_2$	CTCD analytic 2: Harmonic centrality analytic $\mathrm{NHCA}(n)$	164
$\boldsymbol{\mathcal{A}}_3$	CTCD analytic 3: The rate of disruption propagation analytic $\mathrm{NRDP}(n)$	165
$\boldsymbol{\mathcal{A}}_4$	CTCD analytic 4: Maximum disruption propagation analytic $\mathrm{NMDP}(n)$	165
$\boldsymbol{\mathcal{A}}_5$	CTCD analytic 5: Strategic compatibility index $\mathrm{TSCI}(\mathrm{AL})$	167
$\boldsymbol{\mathcal{A}}_6$	CTCD analytic 6: Total disruption strength analytic $\mathrm{TDS}(t)$	168
$\boldsymbol{\mathcal{A}}_7$	CTCD analytic 7: Response task analytic $\mathrm{NRTA}(n)$	168
$\boldsymbol{\mathcal{A}}_8$	CTCD analytic 8: Total response workload $\mathrm{TRW}(t)$	168
$\mathrm{AAVI}(\mathrm{AL})$	Team AL's across-agent-variation index	174
$\mathrm{AEI}(a)$	Agent a's estimated effectiveness index	166
$\mathrm{ANVI}(a)$	Agent a's across-node-variation index	175
DIST	Distance matrix	164
$\mathrm{DIST}(n_i, n_j)$	Shortest-path distance from n_i to n_j	164
$\mathrm{NHCA}(n)$	Node n's harmonic centrality analytic	164
$\mathrm{NMAT}(n)$	Node n's MATW selection index	171
$\mathrm{NMDP}(n)$	Node n's maximum disruption propagation analytic	170
$\mathrm{NMND}(n)$	Node n's MNDP selection index	170
$\mathrm{NNDA}(n)$	Node n's neighboring disruption analytic	163
$\mathrm{NPNL}(n)$	Node n's set of incoming/preceding neighboring nodes	163
$\mathrm{NRTA}(n)$	Node n's response task analytic	168
$\mathrm{NRDP}(n)$	Node n's rate of disruption propagation analytic	165
$\mathrm{NSEL}_{\mathrm{MNDP}}(n)$	Node n's set of outgoing/succeeding, disrupted, and unrepaired nodes.	170
$\mathrm{NSNL}(n)$	Node n's set of outgoing/preceding neighboring nodes	163
$\mathrm{NSV}(n)$	Node n's strategic value	166
$\boldsymbol{\mathcal{P}}_1$	CTCD teaming protocol 1: Random team selection protocol	167

		Page where first defined
$\boldsymbol{\mathcal{P}}_2$	CTCD teaming protocol 2: Low-compatibility team selection protocol	167
$\boldsymbol{\mathcal{P}}_3$	CTCD teaming protocol 3: Medium-compatibility team selection protocol	167
$\boldsymbol{\mathcal{P}}_4$	CTCD teaming protocol 4: High-compatibility team selection protocol	167
$\boldsymbol{\mathcal{P}}_5$	CTCD coordination protocol 1: First-come-first-serve (FCFS) protocol	169
$\boldsymbol{\mathcal{P}}_6$	CTCD coordination protocol 2: Shortest-processing-time (SPT) protocol	169
$\boldsymbol{\mathcal{P}}_7$	CTCD coordination protocol 3: Minimizing neighboring disruption propagation (MNDP) protocol	169
$\boldsymbol{\mathcal{P}}_8$	CTCD coordination protocol 4: Minimizing additional task workload (MATW) protocol	170
$\mathrm{TDS}(t)$	Total disruption strength at time t	168
$\mathrm{TRW}(t)$	Total response workload at time t	168
$\mathrm{TSCI}(\mathrm{AL})$	Team AL's strategic compatibility index	167
$\mathrm{URT}(a, n)$	Unnormalized response time of agent a to node n	175

ABSTRACT

Disruptive events during recent decades have highlighted the vulnerabilities of complex systems of systems to disruption propagation: Disruptions that start in one part of a system and can propagate to other parts. Such examples include: Fire spreading in building complexes and forests; plant/crop diseases in agricultural production systems; propagating malware in computer networks and cyber-physical systems; and disruptions in supply networks. The impacts of disruption propagation are devastating, with fire causing annual US$23 billion loss in the US alone, plant diseases/crop reducing agricultural productivity 20% to 40% annually, and computer malware causing up to US$2.3 billion loss per event (as a conservative estimate). These problems, the response to disruption propagation (RDP) problems, are challenging due to the involvement of different problem aspects and their complex dynamics. To better design and control the responses to disruption propagation, a general framework and problem-solving guideline for the RDP problems is necessary.

To address the aforementioned challenge, this research develops the Collaborative Response to Disruption Propagation (CRDP) unifying framework to classify, categorize, and characterize the different aspects of the RDP problems. The CRDP framework allows analogical reasoning across the different problem contexts, such as the examples mentioned above. Three main components applicable to the investigate RDP problems are identified and characterized: (1) The client system as the victims; (2) The response mechanisms as the rescuers/protectors; and (3) The disruption propagation as the aggressors/attackers. This allows further characterization of the complex interactions between the components, which augments the design and control decisions for the response mechanisms to better respond to the disruptions. The new Covering Lines of Collaboration (CLOC) principle, consisting of three guidelines, is developed to analyze the system state and guide the response decisions. The first CLOC guideline recommends the network modeling of potential disruption propagation directions, creating a complex network for better situation awareness and analysis. The second CLOC guideline recommends the analysis of the propagation-restraining effects due to the existence of the response mechanisms, and utilizing this interaction in optimizing response decisions. The third CLOC guideline recommends the

development of collaboration protocols between the response decisions to maximize the coverage of response against disruption propagation.

The CRDP framework and the CLOC principle are validated with three RDP case studies: (1) Detection of unknown disruptions; (2) Strategic prevention of unexpected disruptions; (3) Teaming and coordination of repair agents against recurring disruptions. Formulations, analytics, and protocols specific to each case are developed. TIE/CRDP, a new version of the Teamwork Integration Evaluator (TIE) software, is developed to simulate the complex interactions and dynamics of the CRDP components, the response decision protocols, and their performance. The evaluator is capable of simulating and evaluating the complex interactions and dynamics of the CRDP components and the response decision protocols. Experiment results indicate that advanced CLOC-based decisions significantly outperform the baseline and less advanced protocols for all three cases, with performance superiority of 9.7-32.8% in case 1; 31.1%-56.6% in case 2; 2.1%-12.1% for teaming protocols, and at least 50% for team coordination protocols in case 3.

CHAPTER 1. INTRODUCTION

1.1 Motivations

Disruptive events during recent decades have inspired greater interest in the concept of disruption propagation and disruption response in complex systems of systems (called client systems): cyber-physical systems (CPS), building complexes, networks of supply, manufacturing, computers, transportation, utility and other infrastructure (Nguyen & Nof, 2019a). Due to the complex interactions and interdependencies within a client system, undesirable disruptions occurring in one part of the client system can propagate to multiple other parts of the system, affecting the performance and viability of the entire system. The existence of disruptions and their propagation can be catastrophic:

- Fire spreading in buildings and forests can result in losses of life and economic damages .
- Infectious stresses and diseases in agricultural plants/crops can lead to food supply shortage and economic losses ().
- Hostile and propagating malware in computer networks, sensor networks, and cyber-physical systems can severe denial-of-service and loss of sensitive information (Snediker, Murray, & Matisziw, 2008; Hao Zhong, Nof, & Filip, 2014; Y. Kim, Chen, & Linderman, 2015; Hao Zhong & Nof, 2015; W. P. Liu et al., 2016; Hao Zhong, 2016; Dusadeerungsikul & Nof, 2019).
- Sudden, unexpected changes to supply/demand in one tier of a supply network can affect the firms of succeeding/preceding network tiers, resulting in economic losses (Arora & Ventresca, 2018; H. Zhong & Nof, 2020).

Against the harmful disruptions and their propagation, response mechanisms are often prepared and/or deployed to reduce, mitigate, or eliminate the disruptions' harmful existences and impacts. Response mechanisms exist in many different forms. Against fire in building complexes: insulating building materials, water sprinklers, responsive firefighting resources such as firetrucks and firefighters. Against agricultural diseases: pesticides, herbicides, immunization, active detection through robots and sensors. Against computer malware: firewalls, security protocols,

active detection and scanning. Against supply network disruptions: network design, backup inventory, backup sourcing, multi-sourcing, and supply/demand reconfiguration. The design and control of response mechanisms allocations and activities are critical to protect the client system from disruptions. Accordingly, the response to disruption propagation problem is defined as follows:

> **Definition 1.** *Response to disruption propagation (RDP) problem.* It refers to a specific problem instance involving disruptions and their propagations negatively affecting a client system, and response mechanisms exist to mitigate or eliminate the disruptions.

In the context of this work, the client system is the "victim" of the harmful disruptions. The disruptions and their propagation are the "aggressors/attackers" that can harm the components and/or subsystems of the client system. The response mechanisms are the "rescuers/protectors" by detecting, removing, and/or preventing (amongst other response actions) the disruptions and their propagation. More details on these aforementioned terminologies are discussed in CHAPTER 3.

The results of a selective literature review indicate that different disciplines and research areas have different modeling approaches and design/control principles towards solving domain-specific RDP problems. These research silos have limited interaction and exchange of knowledge between each other. Thus, the analogies of modeling, reasoning, design, and control between the different problem instances are not fully utilized. Furthermore, there is no "big picture above the different RDP problems": no modeling framework to characterize and categorize the different components of the problem; and no general design and control principles and philosophies that can be applied to different domain-specific RDP problems. The goal of this research is to provide insights into this "big picture above the different RDP problems" by developing solid research frameworks for

1. Systematic characterization and classification of the different components of a general, non-domain-specific RDP problem.
2. Identification of the interactions between the RDP components to provide a basis for further analysis.

3. Development of collaborative design and control principles to improve the performance of the response mechanisms against disruptions and their propagation.
4. Validation of the developed framework and design and control principles through the specification and analysis of different RDP problems, accompanied by extensive numerical experiments.

The union of the aforementioned items 1-3 above consists of the Collaborative Response to Disruption Propagation (CRDP) framework and the Covering Lines of Collaboration (CLOC) principle. The CRDP framework and the CLOC principle are the original contributions of this dissertation.

1.2 Research Problem

Disruption propagation in systems is the challenging problem addressed in this research. Disruptions and their propagations can bring about serious damages to the client system if left unchecked. Against disruptions, response mechanisms can be prepared and deployed, forming the RDP problem. The design and control of response mechanisms must be supported by insights, collaboration, and intelligence to achieve better detection, prevention, and recovery. Therefore, the key challenges include (a) the identification and characterization of the important components of the RDP problem, as well as their interactions; and (b) the development of appropriate collaborative design and control principles to support the response decisions in overcoming the disruptions and their propagation.

There is a need for systematic identification and characterization of the different components of the RDP problems across different research disciplines. Identifying the components allows further characterization and analysis of their interactions, leading to further insights and understandings of their complex interactions. These insights and understandings can then become the foundations of the development of collaborative design strategies and collaborative control protocols for the response mechanisms, with the objective of improving the detection, prevention, and recovery capabilities of the client system. Furthermore, this systematic framework can trigger analogical

reasoning across different research disciplines and subject areas, further increasing the potential for knowledge gains.

1.3 Research Questions

Systematic classification of different RDP problems and their corresponding models enables analogical reasoning and comparison, allowing ideas and insights sharing between the different research disciplines and research areas. Thus, the first research question is stated as follows.

> **Research Question 1:** What is a good framework for systematic identification and characterization of different RDP problems and their corresponding models?

Various research disciplines and subject areas have studied different RDP problems, but the connection between the different disciplines has not been established. Due to the lack of the "big picture above the different RDP problems", separate research silos are unavoidable. The framework developed to address Research Question 1 can be utilized to enable analogical reasoning and knowledge sharing between the different research disciplines. To support the answer to Research Question 1, the second research question is stated as follows.

> **Research Question 2:** What are the necessary components and interactions to be identified and characterized to enable systematic formulation of different RDP problems and their corresponding RDP models?

The identification and characterization of the different components and their interactions enable analogical comparison between different RDP problems and models, but not solving the problems and/or improving the performance of the response mechanisms. To address this part of the research problem, the third research question is stated as follows.

> **Research Question 3:** Based on the answers to Research Questions 1 and 2, what collaborative design and control principles can be developed to provide better response against disruptions and their propagation?

The remainder of this dissertation is to present the Collaborative Response to Disruption Propagation (CRDP) formulation framework to address Research Questions 1 and 2, as well as the Covering Lines of Collaboration (CLOC) design and control principles to address Research Question 3. Three case studies are presented to support the validation of the three Research Questions, together with the summary of four case studies that have been published in the literature.

1.4 General Definitions and Assumptions

By definition, every RDP problem has three types of entities (relevant to RDP): client system, response mechanisms, and disruption propagation. Even though specifying the full scope of RDP problems is not feasible at this stage of research, providing real-life examples of RDP problems can help researchers and practitioners understand the RDP problem and its components. Several real-life examples of RDP problems include:

- Fire spreading: building complexes *(client system)* are subjected to fire occurrences *(disruption)* that can spread *(propagation)*, and water sprinklers, fire extinguishers, as well as active firefighting resources such as firefighters and firetrucks *(response mechanisms)* can be prepared and deploy to tackle the fire.
- Plant/crop infectious diseases: agricultural plants/crops *(client system)* can be infected by diseases *(disruption)* that if left undetected, can infect nearby plants *(propagation)*. Immunization and active detection *(response mechanisms)* can be employed to prevent and mitigate the impacts of diseases.
- Propagating computer malware: computers of a network or a cyber-physical system *(client system)* can be attacked by intelligent malware *(disruptions)*, with malware having the capability to *propagate* to connected computers. Possible *response mechanisms* include firewalls, security protocols, active scanning, and active quarantines.
- Supply network disruption: firms of a supply network *(client system)* can suffer from *disruptions* such as machine breakdowns, failures and disturbances, communication errors/conflicts. A disrupted firm will affect *(propagation)* its immediate suppliers and customers, who in turn affect their suppliers and customers, propagating the impacts of the disruptions beyond the initially disrupted firms. Possible *response mechanisms* include backup inventory, negotiation, and resilient supply network design.

Figure 1.1 illustrates the examples of fire spreading, plant/crop infectious diseases, and propagating computer malware.

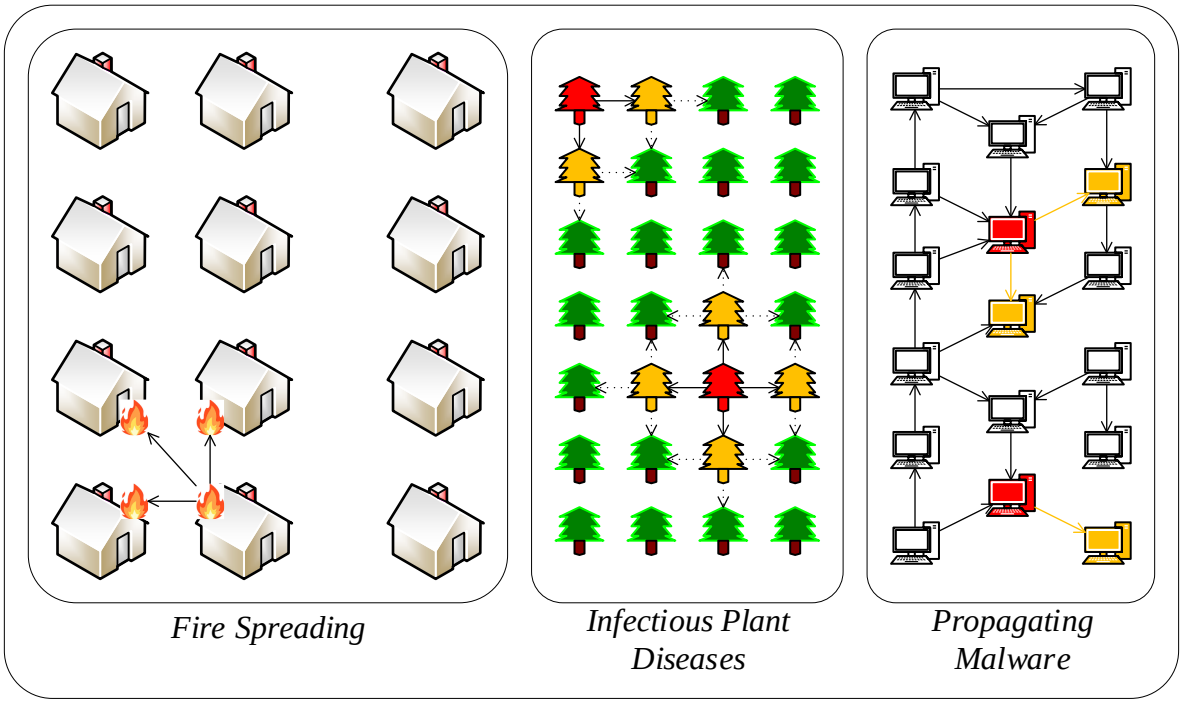

Figure 1.1. Disruption propagation examples

All the aforementioned and similar examples have several components that can be formalized and modeled in a standard and general way. Thus, the following definitions are stated:

Definition 2. *Client system.* In the context of this work, it refers to the system affected or to be affected by disruptions.

Definition 3. *Disruption*. It refers to a negative, harmful, and undesirable existence and/or impact.

Definition 4. *Disruption propagation*. It refers to the spreading of disruptions and/or their impacts beyond the initial occurrence of disruptions.

Definition 5. *Response mechanisms*. It refers to any phenomena or activities that limit or reduce the negative effects of disruptions and their propagation.

Definition 6. *Response to disruption propagation (RDP) model.* It refers to a theoretical model that can simulate and/or emulate the entities, attributes, and events of a corresponding RDP problem.

The aforementioned examples are summarized in Table 1.1, with respect to the aforementioned definitions 2, 3, 4, and 5.

Table 1.1. Summary of background examples with respect to definitions

Case	Client System	Response Mechanism	Disruptions	Disruption Propagation
Fire spreading	Building complexes; Forests	Sprinklers; Fire extinguisher; Firefighters; Firetrucks	Fire	Ongoing fire spreading to nearby objects
Infectious plant disease	Agricultural plants	Immunization; Pesticides; Herbicides; Active detection	Disease/ Stress	Disease/Stress infect nearby plants
Propagating computer malware	Computers; Devices; Sensors	Firewalls; Security protocols; Active scanning and quarantine	Malware: viruses, worms, trojans	Propagation to connected, vulnerable computers
Supply network disruption	Enterprises; Firms; Departments	Backup inventory; Outsourcing; Negotiation	Breakdowns; Communication errors and conflicts	Disruptions affect connected nodes and further

The following assumptions are considered throughout this work.

General Assumption 1. Within the scope of this work, disruptions are assumed to strictly affect the client system negatively. Unexpected and positive events are not within the scope of this research.

General Assumption 2. Within the scope of this work, disruptions are assumed to propagate their existences and/or their impacts. In this purview, disruptions that cannot propagate are typically not as threatening to the client system, because they cannot grow in strength.

General Assumption 3. Within the scope of this work, the response mechanisms are assumed to provide strictly positive effects on the client system. Response mechanisms that can harm the client system (such as a farmer detecting plant diseases can inadvertently spread such diseases to other plants) are not considered at this stage of research.

1.5 Naming, Typesetting and Conventions

Throughout the dissertation, the following naming and typesetting conventions are observed to ensure readability and uniformity.

CRDP framework classification terms: bolded script capital alphabetical single letter, with optional alpha-numeric subscript and/or superscript. Examples include $\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{S}}_{\boldsymbol{R}}^{\boldsymbol{t}}, \boldsymbol{\mathcal{P}}$. This naming and typesetting choice provides distinctions between abstract concepts (specific to the CRDP framework) and other mathematical/programming definitions.

Mathematical naming conventions and typesetting:

- Multiplication sign × is explicitly used, instead of placing variables next to each other. $y = ax + b$ will instead be written as $y = a \times x + b$.
- An object of a set is represented by lowercase italicized single-letter. For example, $n \in \mathrm{NL}$ is used to denote a node, $e \in \mathrm{EL}$ is used to denote an edge.
- Preference of multi-letter variables is observed throughout this dissertation. This naming convention is selected to reduce the understanding gap between mathematical modeling and numerical simulation through programming, and to facilitate analogous reasoning across different research areas and disciplines.
- A multi-letter variable is formatted as uppercase roman abbreviation of the full variable name, and has 4 letters or fewer. For example, node's operational status $\mathrm{NOS}(n)$, agent's busy status $\mathrm{ABS}(a)$.
- A set of objects is represented by an uppercase roman multi-letter variable, beginning with the first letter of the object, and followed by the letter L (short for List, which is a common programming collection type, notably with Python and C#). Examples include the set of nodes NL, the set of edges EL.

- |set_of_objects| or |{objects}| denotes the size/cardinality of a set of objects.
- A mathematical variable is initially defined together with its domain. For example, if a node's operational status takes a real number between 0 and 1, it is defined as $\mathrm{NOS}(n) \in [0,1]$. Other examples include node $n \in \mathrm{NL}$, edge $e \in \mathrm{EL}$.
- For any numerical quantity, if an operation or statement increases/decreases the quantity outside the defined range, the quantity is automatically adjusted to the nearest limit. With the above example $\mathrm{NOS}(n) \in [0,1]$, $\mathrm{NOS}(n) \leftarrow 1.5$ becomes $\mathrm{NOS}(n) \leftarrow 1$.
- $\Rightarrow$ is equivalent to the "if" logical statement.
- $\Leftrightarrow$ is equivalent to the "if and only if" logical statement.
- $\leftarrow$ is equivalent to the assignment statement mainly used for programming and numerical simulation. $x \leftarrow y$ means the value or the reference of y is assigned to variable x at simulation or programming runtime. It is notably different from the notation $=$.

Analytical notations:

- $\nearrow$ or $\searrow$ is employed to describe that certain quantity is conjectured and reasoned to increase or decrease, based on the given incomplete information and/or complex circumstances. While mathematical analysis is employed when possible and useful in this dissertation, many circumstances and interactions are too complex for complete mathematical analysis. In such cases, these analytical notations are employed to present reasonable conjectures in a succinct manner, and to summarize the descriptive explanations. Examples include:

  $a \nearrow \Rightarrow b \searrow$ means if a increases, b is likely to decrease in the near future.

  $x \nearrow \Leftrightarrow y \searrow$ means x is like to increase if y decreases, and vice versus.

The use of chapter-specific notations (for CHAPTER 4, CHAPTER 5, and CHAPTER 6):

- These chapters involve significantly different simulation logic and modeling, with
  - CHAPTER 4 discussing disruptions with unknown status;
  - CHAPTER 5 discussing static strategic resources that be allocated once, and cannot be relocated after disruptions occur;

    - and CHAPTER 6 discussing dynamic repair agents that can be relocated after disruptions occur.
- The significant modeling differences necessitate the use of different mathematical analysis, thus requiring different chapter-specific notations.
- For each chapter, the chapter-specific notations are only effective within the scope of the mentioned chapter, and are specified in the LIST OF SYMBOLS above.

1.6 Dissertation Structure

The remainder of this dissertation is organized as follows:

CHAPTER 2 reviews the background and previous work on the RDP problem in different research domains, on the formulation of RDP problems, and on the Collaborative Control Theory.

CHAPTER 3 presents the CRDP framework and the CLOC principle.

CHAPTER 4, CHAPTER 5, and CHAPTER 6 present three case studies: (1) Detection of unknown disruptions; (2) Strategic prevention of unexpected disruptions; (3) Teaming and coordination of repair agents against recurring disruptions. Each chapter presents its corresponding case's description, formulation, analytics, protocols, and numerical experiments.

CHAPTER 7 summarizes the applications and significance of this dissertation, outlines the answers to the Research Questions, and discusses the recommendations for future research.

The mapping between the Research Questions, their summaries, and the chapters/sections in this dissertation is summarized in Table 1.2.

Table 1.2. Mapping between Research Questions and dissertation structure

Research Question	Research Question Summary	Relevant Chapters/Sections
Research Question 1 – Framework	What is a good framework to classify RDP problems?	Section 3.1
Research Question 2 – Formulation	What are the necessary components and interactions for the formulation of RDP problems?	Subsections 3.1.1-4 Section 4.2 (supporting case) Section 5.2 (supporting case) Section 6.2 (supporting case)
Research Question 3 – Design and Control Principles	What collaborative design and control principles can be developed to provide better response to disruption propagation?	Section 3.2 (main answer) Section 4.3 (supporting case) Section 5.3 (supporting case) Section 6.3 (supporting case)

CHAPTER 2. LITERATURE REVIEW

In this chapter, two theoretical knowledge themes relevant to this research are reviewed. The themes are:

Theme 1: The RDP problem in different domains. This theme reviews the different RDP problems in different domains and research disciplines and presents the similarities between the different problems.

Subtheme 1.1: The fire spreading problem.
Subtheme 1.2: The infectious plant disease problem.
Subtheme 1.3: The propagating computer malware problem.
Subtheme 1.4: The supply network disruption problem.

Theme 2: Formulation of RDP problems. This theme reviews the different approaches for formulating RDP problems.

Theme 3: Collaborative Control Theory. This theme reviews the previous work of Collaborative Control Theory related to the RDP problems.

2.1 The RDP Problem in Different Domains

2.1.1 The fire spreading problem

Fire spreading is a recognizable and prominent RDP problem. Notably, the local fire departments of the United States respond to an average of 1.3 million fires a year (Reyes Levalle & Nof, 2017), roughly 23% of which are wildfires (U.S. Fire Administration, 2017). The impacts of fire are devastating. Fire incurs a yearly economic cost of US$23 billion and loss of life of around 3,400 as well as around 14,000 injuries (Ahrens, 2011). Fire locations include residential, nonresidential, vehicle, and outside. The resources allocated for firefighting activities are significant, with

estimates of 1.16 million firefighters (around 30% career and 70% volunteer) in more than 29,000 fire departments, in the United States (U.S. Fire Administration, 2017).

Fire spreading modeling has been studied extensively in the literature. Fire spreading can be modeled in cellular automata, which are discrete dynamic systems consisting of cells (square-shaped or hexagon-shaped) arranged in a two-dimensional space (Georgoudas, Sirakoulis, & a, 2007; U.S. Fire Administration, 2017). An alternative modeling method is particle system simulation (Hernández Encinas, Hoya White, Martín del Rey, & Rodríguez Sánchez, 2007). Response surface modeling is also employed in analyzing life safety, particularly in smoke spread (Zhou, Zhang, & Qin, 2008). Propagation of both fire and smoke can be considered and modeled in different environments. Fires in forests can be devastating economically, ecologically, and environmentally, and are typically difficult to respond to due to the large scale and distances involved (Van Weyenberge, Criel, Deckers, Caspeele, & Merci, 2017). Fires in building complexes are also important subjects to the potential loss of lives involved, as well as economic damages, family dislocations, and reduced livelihood (Cencerrado, Cortés, & Margalef, 2014; Caton, Hakes, Gorham, Zhou, & Gollner, 2017). Also relevant are smoke and fire spreading in subways, which are studied to ensure subway passenger safety (J. Kim, Dietz, & Matson, 2016). Other environments include fires in ships (Giachetti, Couton, & Plourde, 2017) and fires in construction sites (Jiao, Wang, Xiao, Xu, & Chen, 2014). Notably, numerical simulations are widely employed in the modeling and analysis of fire spreading, due to the complexity and dynamicity of the problems involved. The main observation is that the spreading of fire is mainly influenced by proximity, flammability of objects, and in some cases, environmental conditions such as wind and humidity. Fire spreading can also be modeled using graph theory and network theory, with the nodes representing components and the edges representing fire spreading directions (Tsai, 2016).

Response to fire and fire spreading are also explored in this literature review. Firefighting activities allocation can be supported by scheduling algorithms (Floderus, Lingas, & Persson, 2013). Allocation of both firefighters and firetrucks/fire engines are considered in the literature. The resource allocation and vehicle routing of firefighting and fire rescue vehicles can be critical to response time (Floderus et al., 2013). The resource allocation and scheduling problem can be

supported by predictive analysis, studies of past data, and risk evaluation (F. He, He, Sun, & Chen, 2014; L. He, Fan, Liu, Chen, & Li, 2014). These techniques and methodologies are applicable to the different fire spreading environments: buildings, forests, power infrastructure, etc. (F. L. Liu & Wang, 2012; J. Kim et al., 2016). Local firefighters are also investigated as a potential approach for immediate firefighting, which can reduce the fire spreading potential and prevent further losses, although there exist risks to the less professional local firefighters (Lu, Guo, Jian, & Xu, 2018). Firefighting activities are highly dangerous activities, requiring proper training and equipment preparation (Himoto & Tanaka, 2012). Usage of predictive analysis and past data can be critical in achieving timely allocations of response activities (Osorio, Fernandez-Pello, Urban, & Ruff, 2013; Cencerrado et al., 2014). Another important subject in this research area is the study of fire emergency evacuation activities: the study of crowd behavior and fire exits (Georgoudas et al., 2007; Manes & Rush, 2019). Emerging fire response technologies include the autonomous firefighting mobile robots (Zheng, Jia, Li, & Jiang, 2017) and similar unmanned autonomous vehicles (flying drones) (Anantha Raj & Srivani, 2018; Sherstjuk, Zharikova, & Sokol, 2018). These new technologies are significant because of the reduced risks of loss of lives and the potentials of autonomous having the capability to tackle more risky situations. The robots and vehicles can be autonomous and/or controlled remotely, and can tolerate more hazardous environments of high temperature and/or toxicity.

Table 2.1 provides a summary of the literature on fire spreading on four aspects: impacts, client system, response mechanisms, and modeling approaches.

Table 2.1. Summary of literature on fire spreading

Aspect	Details
Impacts	Annually (data in 2017), all roughly: 1.3 million fires a year, US$23 billion economic loss, roughly 3,400 deaths, 14,000 injuries (Zharikova & Sherstjuk, 2018)
Client system	Buildings, building complexes, forests/wildfires, ships, construction sites, subway, ships.
Response mechanisms	Static: fire sprinklers, local fire extinguisher. Dynamic and active: firefighters, firetrucks/fire engines
Modeling approaches	Cellular automata, particle simulation, response surface modeling, graph/network theory. Fire spreading from proximity, flammability, environmental conditions.

2.1.2 The infectious plant disease problem

Infectious plant/crop disease is another recognizable and prominent RDP problem. Henceforth, this dissertation refers to both plants and crops as plants for brevity. Modern agricultural systems typically involve tightly packed communities of plants; thus, stresses and diseases can propagate from one plant to nearby ones. The impact of infectious plant disease can be devastating. Agriculture plant yield loss ranges from 20% to 40%, due to direct causes of pathogens/diseases, animals, and weeds (U.S. Fire Administration, 2017). This loss is not the true cost to society, however, because food security (the challenge relevant to this problem) involves food availability, physical and economic access to food, and food utilization (Savary, Ficke, Aubertot, & Hollier, 2012). The plant diseases worsen the global food security outlook, given that at least 800 million people are inadequately fed in the 2000s (Ingram, 2011). This is especially catastrophic for developing countries and countries with high populations (Strange & Scott, 2005). Plant protection and response to diseases are becoming increasingly important, due to the reduction of important natural resources to agricultural production: water, arable land, energy, fertilizers, etc.

Crop losses are not necessarily caused only by plant diseases, but also by harmful pests and insects (countered by pesticides), fungi (Vurro, Bonciani, & Vannacci, 2010), competitive weeds (countered by herbicides) (Ducrot & Matano, 2016). While these disruption mechanisms differ in nature, they all spread and propagate in similar manners: through the plant close proximity enabled

by the necessity of large scale and size of agricultural production. The population of these pathogens can spatially expand and temporally (over-time) increase (Diggle, Salam, & Monjardino, 2006). Understanding the propagation mechanisms is critical to deploy control methods effectively (Gilligan, 2008; Donatelli et al., 2017). While lower in scale and productivity, plant trades and plant nurseries also face the infectious disease problem (Estrada, Meloni, Sheerin, & Moreno, 2016). A general framework specific to crop losses was developed by Bate et al. (2016). This framework identifies four key components: epidemic, injuries, crop loss, and economic loss. Epidemics may lead to crop injuries, which in turn may lead to crop loss and economic loss. Based on this framework, two plant protection decision types can be characterized: strategic decisions and tactical decisions. Strategic decisions are made before the crops are established, and include crop types and gene/breeding decisions. Tactical decisions are made during the cropping seasons and include fertilizing, pesticide, and herbicide decisions.

Certain prevention and control mechanisms can be deployed in such cases to mitigate the impacts of infectious plant diseases. Due to the large scale and size of agricultural activities, detections of plant diseases and stresses can be very difficult, requiring advanced automation (Savary et al., 2012). Even in the smaller greenhouse environment, manual disease detection and monitoring are often hampered by human resources, low sampling rate, and high monitoring costs (Shanmugam, Adline, Aishwarya, & Krithika, 2017). Automated disease detection often employs the use of advanced vision cameras, hyperspectral imaging (Schor et al., 2016), pattern recognition techniques, and machine learning techniques to determine whether a plant is infected or not (D. Wang et al., 2018). When a part of a plant (such as the leaves) is affected by diseases, the color of the part changes, which can be captured by machine vision. This image-based approach is often enabled and augmented through the use of neural networks, particularly convolutional neural networks (Ocampo & Dadios, 2018; Fu, Wang, & Ji, 2019). These machine learning techniques often enable accurate and reliable detection of diseases without relying on large amounts of existing data.

For both manual and automated detections, the detection decisions (where and when to detect) can be supported and advised by the use of advanced statistical inference techniques, such as Bayesian inference, Markov chain analysis (Singh, 2018), and advanced data mining techniques (Gibson et

al., 2006). Other advanced collaborative control and task administration methods include adaptive search and routing optimization (Shah, Shah, Malensek, Pallickara, & Pallickara, 2016). This is necessitated by the large scale of agricultural production activities, which make exhaustive searching infeasible and economically prohibitive (Dusadeerungsikul & Nof, 2019). Involvement of human operators and experts through remote control and telerobotic can also improve detection accuracy and speed (P. Guo, Dusadeerungsikul, & Nof, 2018; Dusadeerungsikul & Nof, 2019). This is necessary due to the technical challenges in robotics and machine vision, limiting the reliability, accuracy, and speed of the machine detection activities. Early detection is also highly emphasized, due to its capability to detect pathogens and worsening conditions before they can incur serious and irrecoverable damages (Dusadeerungsikul, Nof, & Bechar, 2018).

Table 2.2 provides a summary of the literature on plant diseases on five aspects: impacts, client system, response mechanisms, disruptions, and control approaches.

Table 2.2. Summary of literature on infectious plant diseases

Aspect	Details
Impacts	Globally 20% to 40% reduction in crop yield, with some countries suffering from more devastating impacts.
Client system	Farms, crop fields, greenhouse, plant nursery.
Response mechanisms	Active disease detection, disease monitoring, immunization (herbicides, pesticides), quarantine.
Disruptions	(collectively called disease in this work) Pathogens, fungi, pests, insects, weeds.
Control approaches	For detection: machine vision, robotics, pattern recognition. Task administration and coordination: routing optimization, adaptive search, statistical inference.

2.1.3 The propagating computer malware problem

Propagating computer malware is another prominent RDP problem. A single computer worm attack can cause economic damages as much as US$2.6 billion (Orozco-Fuentes et al., 2019). Worldwide financial losses amounted up to US$110 billion, as of 2012 (Tidy & Woodhead, 2018). Such computer malware can infect as many as 90% of the vulnerable computers within 10 minutes, too quickly for human administrators to timely intervene. The malware can be even more

dangerous with collaboration capabilities (H. Guo, Cheng, & Kelley, 2016). Propagating malware can spread stealthily through computer networks by using zero-day exploits, causing significant damage before they can be detected and removed (Y. Zhang, Bhargava, & Hurni, 2009). Against the stealthy malware, defenders generally deploy intrusion detection systems (which looks out for suspicious activities) and intrusion prevention system (which blocks off malicious activities) (Thompson, Morris-King, & Cam, 2016).

Traditionally, the primary targets of propagating malware are computer networks, such as the networks of large enterprises, utility companies, the government, and military systems. These computer networks often have classical and standardized defensive measures: firewalls, virtual private networks, and intrusion detections (Ahmad, Woodhead, & Gan, 2016). As information and communication technologies become more mobile and ubiquitous, the concept of computer network also evolves. Mobile phones (with each smart device having sufficient computing power and functionalities), for example, are increasingly becoming targets of propagating malware through due to the prevalence of Bluetooth and Wi-Fi technologies (Zyba, Voelker, Liljenstam, Mehes, & Johansson, 2009; Eder-Neuhauser, Zseby, & Fabini, 2018). This problem context is different from the computer networks with centralized defenses in that the malware can propagate in a distributed manner and through proximity of location. This problem is important due to the increasing tendency of mobile phone users to store their personal, financial, and sometimes work-related information on their phones (W. Liu et al., 2016). Proposed strategies for this type of malware include local detection, proximity signature dissemination, and broadcast signature dissemination, but these strategies largely depend on each device's detection capabilities.

Critical infrastructures such as the smart power grids are also potential targets of propagating malware (S. Peng, Wang, & Yu, 2013; Eder-Neuhauser et al., 2018). Due to the large scale of the grids, utility companies often implement device (i.e. smart meters, controllers) standardization to decrease operational costs and maintenance costs. The standardization, however, also creates favorable conditions for the malware to propagate quickly (Park, Nicol, Zhu, & Lee, 2013). Wireless sensor networks are also prevalent targets of propagating malware, due to the limited computing capabilities of the sensor nodes (S. Shen et al., 2014; Eder-Neuhauser et al., 2018). The limited computing power of the sensors means they have very limited capabilities to defend

themselves against malware (del Rey, Guillén, & Sánchez, 2016). Updates through physical contacts are possible, but inconvenient, whereas remote and wireless updates also open the doors for malware to infect the sensor nodes. Dynamic game theory-based control strategies have been proposed for this problem context (Queiruga-Dios, Hernández Encinas, Martín-Vaquero, & Hernández Encinas, 2017).

The increasing popularity of social media platforms such as Facebook and Twitter also provide a prominent target for propagating malware, because the users are not well-informed in good security and privacy practices (Faghani & Saidi, 2009; Cheng, Ao, Chen, & Chen, 2011; S. Shen et al., 2014). Social media users are susceptible to phishing attacks and spam messages, and the acquaintances/friends of the compromised users can become more vulnerable due to the established "trusts" and connections between the users. Similar malware propagation behavior can be observed in peer-to-peer networks and wireless ad-hoc networks, where the trusts between the peers can lead to more severe damages from malware (Jia, Liu, Fang, Liu, & Liu, 2018). Possible damages include identity theft, loss of sensitive information, loss of financial information on the end-user side, as well as long term loss of customer trust on the platform developer side.

Widely-used approaches in modeling propagating malware include the Susceptible-Infectious, Susceptible-Infectious-Susceptible, and Susceptible-Infective-Recovered model and/or a variation of such models (Yurong, Guo-Ping, & Yiran, 2008; C. Wang, Fu, Bai, & Bai, 2009; S. Shen et al., 2014; Thompson et al., 2016; Queiruga-Dios et al., 2017; B. Liu et al., 2018; W. Liu & Zhong, 2018; Musa, Almohannadi, & Alhamar, 2018; Valizadeh & van Dijk, 2019). Agent-based modeling, in which the malware is modeled as intelligent agents with awareness and reasoning capabilities, is also employed (Yu, Gu, Barnawi, Guo, & Stojmenovic, 2015). Important factors that affect the propagation include preexisting knowledge/preferences (of the malware), communication protocols, computer network topology/structure (Batista, Martín del Rey, & Queiruga-Dios, 2018). Particularly, the certain network structure types, such as the scale-free and small-world types, are prevalent in propagating malware research (Cooke, Mao, & Jahanian, 2006; Yurong et al., 2008; H. Guo et al., 2016). The work of Faghani and Saidi (2009) indicates that the malware propagation growth could follow the exponential distribution, which is confirmed with real-world global scale malware data.

It is noted that most of the contemporary research on propagating malware focuses on the client system, the network topology/structure of the client system, the malware (disruptions), but clearly not as much on the design and control of response mechanisms. Table 2.3 provides a summary of the literature on propagating malware on four aspects: impacts, client system, response mechanisms, and notable observations.

Table 2.3. Summary of literature on propagating computer malware

Aspect	Details
Impacts	A single computer worm attack can cause up to US$ 2.6 billion. Worldwide financial losses amounted up to US$110 billion, as of 2012.
Client system	Computer networks in large enterprises; Mobile devices; Critical infrastructures and cyber-physical systems; Wireless sensor networks; Social networks;
Response mechanisms	Firewalls; Security protocols; Communication protocols; Anomaly detection;
Notable observations	Unlike the case of fire spreading and infectious disease, computer malware can possess significant intelligence. Contemporary research focuses on malware propagating mechanisms, infection/attack attributes, and network topology. There is a gap in the study of control and response to propagating malware.

2.1.4 The supply network disruption problem

Supply chain/network disruption is another prominent RDP problem. Notable contemporary examples include the Thailand Floods in 2011 disrupting the global production of computer hard drives and a 2007 earthquake in Japan paralyzed 70% of automotive production in Japan for at least a week (H. Guo et al., 2016). The capability of supply networks to cope with disruptions is often termed supply network resilience (Chozick, 2007). Demand/supply disruptions can affect almost any industry with physical production: electronics, automotive, medical supplies, food, home appliances, etc (Nof, 2013; Scheibe & Blackhurst, 2018). The supply network paradigm also includes other form of supply/demand: inter-firm manufacturing networks (Schmitt & Singh, 2012); information in sensor network (Zhan, Qingbo, & Tingxin, 2014; Reyes Levalle, 2018); water in water supply network (S.-P. Zhang et al., 2015). Manufacturing networks can be treated

as a special case of supply network, with the manufacturing networks being more contained in control and less focused on pricing (Simão, Coutinho-Rodrigues, & Current, 2004; Chi-Yu Huang, Cheng, & Holt, 2007; Chi-Yu Huang, Holt, Monk, & Cheng, 2007; Firmansyah & Amer, 2013; Gu, Jin, & Ni, 2014). It is noted that supply network disruptions are notably different from disruption propagation in the fire spreading case, infectious plant disease, and propagating computer malware:

1. In the cases of fire spreading, infectious plant disease, and propagating computer malware, the disruptions (fire, disease, malware) propagate their existences to other parts of the client system (building/forest, plants, computer networks). Fire causes more fire, diseases spread more diseases, and malware creates more malware. In this case, the existences of the disruptions are propagated.
2. In the case of supply network disruption, the supply/demand disruptions (production breakdowns, natural disasters) do not necessarily propagate themselves, i.e. a strike in General Motors does not necessarily lead to strikes for their customers. In this case, the impacts of the disruptions are propagated.

Contemporary modeling approaches include the use of network science and/or complex networks, particularly node removal analysis, centrality analysis and clustering analysis (Ismail, Poolton, & Sharifi, 2011; Dixit, Seshadrinath, & Tiwari, 2016; Han & Shin, 2016; Arora & Ventresca, 2017; Datta, 2017; K. Zhao, Scheibe, Blackhurst, & Kumar, 2019; Kang Zhao, Zuo, & Blackhurst, 2019). The complex network approach can also be combined with agent-based modeling (Behdani, Dam, & Lukszo, 2011; Reyes Levalle & Nof, 2015b; C. S. Tan, Tan, & Lee, 2015; Bhargava, Levalle, & Nof, 2016; P. S. Tan, Lee, & Tan, 2016; Reyes Levalle & Nof, 2017; Arora & Ventresca, 2018; Reyes Levalle, 2018), although the investigated supply networks using agent-based modeling tend to be smaller in size. The limited network sizes are also observed with the researchers employing mathematical optimization and programming approaches (Mari, Young Hae, & Memon, 2014; Reyes Levalle & Nof, 2015a; Ghavamifar, Makui, & Taleizadeh, 2018; Jabbarzadeh, Haughton, & Khosrojerdi, 2018; Diabat, Jabbarzadeh, & Khosrojerdi, 2019; Sawik, 2019). The limited size is possibly caused by the large number (often more than five) of variable types involved. The smaller network size, however, also allows more detailed analysis and solution methods to be

performed, such as genetic/memetic algorithms (Fattahi, Govindan, & Keyvanshokooh, 2017), decision tree (Hasani & Khosrojerdi, 2016), inventory models (Ponnambalam et al., 2013). The detailed analysis can consider a complex and dynamic supply/demand interactions between the nodes (firms): negotiation, pricing, inventory, and transportation (Xanthopoulos, Vlachos, & Iakovou, 2012), whereas only basic node/edge attributes can be considered in the complex networks approach. Supply network risks are also highly related to disruptions, with similar behaviors and characteristics (Basole, 2016; Paul, Sarker, & Essam, 2017).

Several researchers also investigate supply network resilience from the network topology/structure perspective. Supply network resilience can be measured by removing nodes (firms) from the network, and resilience measures such as supply availability and functional network size can be quantified (Basole & Bellamy, 2014; Arora & Ventresca, 2018). Certain supply network types can be generalized, such as block-diagonal, scale-free, centralized, and diagonal (Nair & Vidal, 2011), with scale-free networks experience higher resilience against node/edge removal disruptions. In such cases, centrality measures can be applied and computed, with notable examples include betweenness centrality and harmonic centrality (Y. Kim et al., 2015). The main observed advantage of this approach is that large network sizes (up to thousands of nodes) can be simulated, with the main drawback being the lack of consideration of complex interaction mechanisms between the nodes, edges, and disruptions. Several supply network resilience measures have been proposed: reliability, response time, recoverability (Y. Kim et al., 2015; Mohapatra, Nanda, & Adhikari, 2015).

Design and control approaches for supply network resilience against disruptions mainly focus on backup inventory and emergency/contingency planning (Gu et al., 2014; Reyes Levalle & Nof, 2015b; Parajuli, Kuzgunkaya, & Vidyarthi, 2017; Reyes Levalle & Nof, 2017; Reyes Levalle, 2018). The backup inventory approach requires limited collaboration and information sharing, but its impacts are mainly at the local level. On the other hand, collaboration and teaming approaches can improve resilience at the network level, but require cooperative information sharing and rigorous planning (Reyes Levalle & Nof, 2015a, 2015b). Backup/contingency planning is not limited to inventory, and could include contingency distribution and sourcing to mitigate the impacts of disruptions (Seok, Kim, & Nof, 2016; Yavari & Zaker, 2019). These design and control

approaches are not only shown to be satisfactory with numerical experiments, but also with industrial surveys and empirical studies (Xu et al., 2015). Facility locations, distribution centers location, and transportations are also important considerations of supply networks (Azad, Davoudpour, Saharidis, & Shiripour, 2014; Topal & Sahin, 2018). Even when physical production is not affected, disrupted logistics and transportation capabilities can negatively impact the supply network viability (Gong, Mitchell, Krishnamurthy, & Wallace, 2014).

Table 2.4 provides a summary of the literature on supply network disruptions on four aspects: impacts, client system, response mechanisms, and notable observations.

Table 2.4. Summary of literature on supply network disruptions

Aspect	Details
Impacts	Supply network disruptions general belong to the category of propagating disruption impacts, not disruption existences. Impacts include reduced production capability and economic losses.
Client system	Networks of firms; Water supply network; Information network; Manufacturing networks
Response mechanisms	Backup inventory; Information sharing; Contingency/emergency sourcing and distribution
Notable observations	Contemporary research investigates this problem from both the complex networks perspective and the agent-based perspective.

2.2 Formulation of RDP problems

Despite the different research domains and approaches, the different aforementioned RDP problems share several common elements: the client system, the response mechanisms, and disruption propagation. The client system (“the victims”) consists of the components and subsystems that can be affected by harmful disruptions. The disruptions (“the aggressors/attackers”) have the capability to attack and incur harmful effects on the client system, and can propagate the disruptions existence/impacts to other parts of the client system. The response mechanisms (“the rescuers/protectors”) can remove the disruptions and/or limit their effects.

The findings from the four aforementioned problem contexts indicate that different disciplines and research areas (fire spreading, infectious diseases, malware propagation, supply network disruption) have different disruption modeling approaches. For instance, a fire occurred in one part of a building complex or a forest can readily spread to near parts, causing catastrophic damages if left un-responded to (Day, 2014; Anantha Raj & Srivani, 2018). Stresses and diseases in agricultural plants can propagate to other nearby plants (W. Peng, Feng, Che, & MengChu, 2018). Hostile malware in computer networks and sensor networks can propagate to connected nodes, compromising the performance of a network and possibly forcing an entire network to shut down (Y. Kim et al., 2015; W. P. Liu et al., 2016; Dusadeerungsikul & Nof, 2019). Sudden changes to supply/demand in one tier of a supply network can affect the firms of succeeding/preceding network tiers (Snediker et al., 2008; Arora & Ventresca, 2018). Disruptions can be both external, i.e. natural disasters, malware attacks, supply/demand changes; or internal, i.e. human errors, communication and assignment conflicts, and physical equipment wear and tear. The impacts of disruptions and their propagation can be devastating. Undetected plant diseases can lead to lower productivity and food shortages. Supply disruptions in supply networks can lead to raw materials and intermediate components shortage, resulting in severe revenue losses for enterprises (Day, 2014; Reyes Levalle & Nof, 2017). Cyber-attacks on computer networks and information networks can lead to the immediate compromises of sensitive information and service denials, as well as the long-term equipment damage, loss of customer's trust and strategic advantages. External disruptions include natural disasters and security issues, which can disrupt the production of certain raw materials and intermediate production steps; damage or destroy infrastructure, which can negatively impact manufacturing processes and product quality (Day, 2014; Gong et al., 2014). Internal disruptions include uncertainties, human errors, communication conflicts, equipment and machinery breakdowns in supply networks and manufacturing networks (Gong et al., 2014).

Among the plethora of research on disruption propagation, the nature and mechanisms of the disruptions are highly diverse and dependent on the network defined by the respective authors. Network disruptions are defined as the removal of nodes and edges from the network in several articles (Barabasi & Albert, 1999; Sajadi, Esfahani, & Sorensen, 2011; S. Q. Shen, J. C. Smith, & R. Goli, 2012; S. Q. Shen, 2013; T. Y. Wang, Zhang, Sun, & Wandelt, 2017). This disruption type is widely investigated together with an important class of networks, called the scale-free networks,

which includes the Internet, cells and metabolic networks (R. Albert, Jeong, & Barabasi, 2000). This class of networks exhibits a very high degree of resistance against nodes and edges removal. Due to the low characteristics path length of these scale-free networks, however, they are more vulnerable to disruptions that propagate through the networks (as opposed to those that remove the nodes and edges). Algorithms and strategies to allocate node/edge removal disruptions in networks to optimize certain objectives are also discussed (Réka Albert, Jeong, & Barabási, 2000). Examples include the maximization of the number of graph components and the minimization of the largest component size after node removal. The design and operations of networks subjected to edge removal are also discussed (S. Shen, J. C. Smith, & R. Goli, 2012). It is noted that the research surrounding node/edge removal disruptions is highly related to graph theory and network theory, particular to the concepts of node degree, degree distribution.

Other research focuses on the disruptions concerning the attributes of the nodes and edges of the network. From a modeling perspective, the attributes of the nodes and edges can be freely designed, giving researchers the capability to model the actual physical networks and their operations more accurately. In supply networks, disruptions mainly concern the attributes of production capabilities of the nodes, which in turn affect the supply/demand relationships (attributes of edges). Disruptions in the S. Q. Shen (2013); Reyes Levalle and Nof (2015b, 2017) reduce the quality of service and the outputs of the nodes, affecting succeeding nodes. Disruptions in Reyes Levalle and Nof (2015a) reduce output flows of the nodes, requiring succeeding nodes to have contingent sources or face reduced production rate and affecting other succeeding nodes. Traffic disruptions in road networks n the work of Seok et al. (2016) are concerned with the attribute traffic density of the edges and. The disruptions in L. Zhang, Gier, and Garoni (2014); Hao Zhong (2016) are concerned with the attribute failure status of the nodes and edges. The effects of choosing different disruption targets in CPSs are discussed in the work of Hao Zhong and Nof (2015).

Due to the complex interactions and interdependencies within a network, disruptions can propagate from a node/edge to other nodes/edges. For example, a seminal work by S. L. Wang, Hong, Ouyang, Zhang, and Chen (2013) investigates a disruption propagation mechanism by load-based mechanism, in which disrupted nodes reduce the load of their connected nodes. In Motter and Lai (2002); Hao Zhong and Nof (2015) the disruptions occur initially and target nodes, and can

propagate to the connected edges, which in turn propagate the disruptions to other nodes. In Hao Zhong (2016), disruptions (which are node removals) affect the loads of the nodes, and when a node fails due to insufficient load, it is removed from the network, which reduces the load of its neighbor nodes. In the work of Yin, Liu, Liu, and Li (2016), disruptions can propagate the impacts to other nodes through relationships and functions that can be customized to individual nodes. Other research that investigate disruption propagation include (Buzna, Peters, Ammoser, Kuhnert, & Helbing, 2007; Buldyrev, Parshani, Paul, Stanley, & Havlin, 2010; Chaoqi, Ying, & Xiaoyang, 2017; Chaoqi, Ying, Yangjun, & Xiaoyang, 2017; Guariniello & DeLaurentis, 2017; Chaoqi, Ying, Kun, & Yangjun, 2018). In general, the mechanisms of propagation of disruptions are specific to the networks modeled by the researchers. Two main observations are made: (1) for unweighted networks, the disruption propagations are generally related to the total degree or the out-degree of the nodes, and (2) for weighted networks, the disruption propagations are generally related to both the degree of the nodes and the attributes of the nodes and edges as defined by the networks of interest.

The response mechanisms and strategies investigated in the literature are also specific to the networks modeled by the researchers. Crucitti, Latora, and Marchiori (2004) employ a response mechanism with a gradually increasing amount of response capability which is determined by a function, with the response strategies considering the status of the nodes (undisrupted, moderately disrupted, or fully disrupted). Buzna et al. (2007); Chaoqi, Ying, and Xiaoyang (2017); Chaoqi et al. (2018) investigate response mechanisms as balancing energy loads of nodes, and the response decisions are concerned with which nodes, edges, and the amount. Reyes Levalle and Nof (2015b); Chaoqi, Ying, Yangjun, et al. (2017); Reyes Levalle and Nof (2017) employ agent-based and semi-centralized decision making to re-route supply/demand flows to sustain network performance during disruptions. Reyes Levalle and Nof (2015a) investigate the initial allocation of repair agents to contain disruption propagation, and Hao Zhong and Nof (2015) further improves by introducing better online scheduling protocols. Hao Zhong (2016) investigate centralized and decentralized algorithms to prevent errors and conflicts in complex networks, which can propagate throughout the network if left undetected and un-responded to. X. W. Chen and S. Y. Nof (2012) investigate various repair team and equipment configurations in electrical networks, but the disruptions concerned do not propagate.

From the literature survey, it is apparent that there exists no general framework to unify the different RDP problems between the different problem domains and research disciplines. This knowledge gap is addressed by the CRDP framework. A summary of the literature findings surrounding the formulation of RDP problems is provided in Table 2.5.

Table 2.5. Summary of literature on the formulation of RDP problems

Aspect	Details
Client system	Components/subsystems can be modeled as nodes. Each node can include attributes representing node's characteristics.
Response mechanisms	Response mechanisms can be inherent in the client system (supply network structure). Response mechanisms can be passive/static or active/dynamic. Response mechanisms are often guided with intelligence through either design or control.
Disruptions	Disruptions can be modeled as separate entities or as node/edge removals. Disruptions can be modeled as node/edge attributes representing disruptions. Disruptions can be modeled with disruptions attribute representing strengths, targets, etc.
Disruption propagation	Disruption propagation directions can be modeled as undirected/bidirectional or directed/unidirectional edges. Disruption propagation characteristics can be defined as edge attributes.

Several selected research articles for comparison are summarized in Table 2.6.

Table 2.6. Summary of literature on the formulation of RDP problems

Disruption	Propagation	Response mechanisms	Response protocol	Response vs propagation	Work
Node and/or edge removal	Yes	No	No	No	a
Node, IN	Yes	No	No	No	b
Node	Yes	Yes	Yes, by attribute	Yes	c
Node, load-based	Yes	No	No	No	d
Node, general function	Yes	No	No	No	e
Node, load-based	Yes	Undisrupted node re-distribution	Yes, by target	Yes	f
Node, edge, binary	Node to edge, edge to node	Response agents	Yes, by distance	Yes	g
Node, edge	Yes	No	No	No	h
Node	Yes	Rewiring edges	Yes	No	i
Node, 0 to 1	Yes	Repair agents	Yes, by informatics	Yes	j
Node, binary	Yes, by edge direction and weight	Response agents	Yes, supported with analytics	Yes	k
Node, binary, unknown	Yes, by direction	Detection agents	Yes	Yes, based on CLOC	CHAPTER 4 – Case 1
Node, 0 to 1, unexpected	Yes, bidirectional	Strategic resources	Yes	Yes, based on CLOC	CHAPTER 5 – Case 2
Node, binary	Yes, by edge direction and weight	Team of repair agents	Yes, teaming and coordination	Yes, based on CLOC	CHAPTER 6 – Case 3

Table 2.6 continued
IN: interdependent network a: (Landegren, Johansson, & Samuelsson, 2016) b: (R. Albert et al., 2000) c: (Buldyrev et al., 2010) d: (Crucitti et al., 2004; Buzna et al., 2007) e: (Motter & Lai, 2002; Guariniello & DeLaurentis, 2017) f: (W. Liu et al., 2016; Chaoqi, Ying, & Xiaoyang, 2017; Chaoqi et al., 2018) g: (Hao Zhong et al., 2014; Hao Zhong & Nof, 2015; Hao Zhong, 2016; Chaoqi, Ying, Yangjun, et al., 2017) h: (S. Q. Shen, 2013; Hao Zhong, Wachs, & Nof, 2013) i: (S. Q. Shen et al., 2012; Reyes Levalle & Nof, 2015b, 2017; Reyes Levalle, 2018) j: (Reyes Levalle & Nof, 2015a) k: (Nguyen & Nof, 2018)

2.3 Collaborative Control Theory

The dynamics and relations between the client systems, response mechanisms, and disruption propagation in RDP problems are complex and thus, require the use of appropriate design and control principles. The Collaborative Control Theory (CCT) consists of various principles for the design and control of collaboration between systems and agents (Nof, 2007; Nguyen & Nof, 2019a), and is thus selected for in-depth review. The CCT principles focus on the design and control of the sharing of information, resources, and tasks. Collaboration in this context is classified into mandatory collaboration, optional collaboration, and concurrent collaboration (Nof, Ceroni, Jeong, & Moghaddam, 2015). Once the relevant collaboration mechanisms are defined for a certain CPS, the first principle to be applied is the cooperation requirement planning (CRP) principle. This principle involves identifying the resources required to finish the possible tasks in the CPS. The principle CRP also involves real-time planning/execution of tasks and revision of the plan. Another CCT principle, e-Work parallelism, highlights the importance of utilizing parallelism in CPS. The e-Work parallelism principle involves the analysis of task dependencies in both the cyber (data) and physical dimensions to find opportunities to parallelize tasks (Nof et al., 2015).

An important related CCT principle to the RDP problem is the error prevention and conflict resolution (EPCR) principle (X. W. Chen & S. Y. Nof, 2012; Xin W. Chen & S. Y. Nof, 2012;

Nof et al., 2015; Chen & Kockelman, 2016). This principle involves detecting errors and conflicts in a CPS, then dispatch agents to resolve the errors and conflicts. The EPCR principle defines errors and conflicts as the violations of specifications or characteristics of the client system. The EPCR principle is strongly related to the RDP problem in that errors and conflicts can be disruptions that are harmful to the client. Earlier EPCR research focuses on the detection and resolution of conflicts and errors, but more recent research focuses more on early detection and prevention. In the context of EPCR, errors and conflicts could indeed have propagating effects, but propagation was not the focus of the EPCR principle. Also related is the CCT research on agricultural robotic systems (ARS), which focus on the early detection of agricultural epidemic stresses and diseases (Xin W. Chen & Shimon Y. Nof, 2012; Dusadeerungsikul et al., 2018; Dusadeerungsikul & Nof, 2019). The ARS research focuses more on adaptive searching and routing of autonomous robots and human-in-the-loop operations. Both the EPCR principle and the ARS research are highly relevant to this dissertation.

Another important CCT principle is the collaborative fault tolerance (CFT) principle, which highlights the higher efficiency and reliability from having numerous weaker agents that collaborate with each other, as opposed to having few stronger agents. One interesting CCT principle is the association/dissociation principle, which looks at the decisions of agents to join/leave/remain in teams. This principle considers the selfish/local interests of the agents and models the decisions of the agents based on the perceived benefits of joining, remaining in, and leaving a team. One important derivative work of the CFT principle is the Resilience by Teaming (RBT) principle of (Reyes Levalle & Nof, 2015b, 2017; P. Guo et al., 2018; Reyes Levalle, 2018). The RBT principle focuses on information sharing between intelligent agents, as well as situation awareness at the local-level and network-level of the supply network to mitigate the impacts of disruptions. The RBT principle is an important inspiration for this work.

A possibly relevant CCT to the RDP problems is the Best Matching Protocol (BMP) principle. This principle involves developing efficient matching protocols that find the best matches between two or more sets of agents or entities (Velasquez & Nof, 2008a, 2008b; Reyes Levalle & Nof, 2015a). The simplest case of BMP is the classical one-to-one matching problem, which can be solved optimally using the Hungarian Algorithm. The BMP principle finds many applications in

recent CCT research, including collaborative tool sharing, demand and capacity sharing, reconfigurable supply networks, as well as task administration protocols (Velasquez & Nof, 2009; M. Moghaddam & Nof, 2015; Bhargava et al., 2016). With respect to the RDP problems, agents could have different response times and disruptions/nodes could have different response requirements (requiring different agents or different agent types), requiring BMP to achieve higher performance. A notable CCT work is the PRISM Best Matching Taxonomy, which presents a taxonomic framework to classify best matching problems and characteristics (M. Moghaddam & Nof, 2014). The Best Matching Taxonomy is an important inspiration for the development of the CRDP framework.

The most relevant CCT principles to the RDP problems are the Emergent Lines of Collaboration and Command (ELOCC) principle and the Dynamic Lines of Collaboration (DLOC) principle. The ELOCC principle enables CPSs to make effective decisions when the CPSs are being challenged and/or forced to change (Velasquez, Yoon, & Nof, 2010; Mohsen Moghaddam & Nof, 2016). The ELOCC principle also emphasizes the exchange and creation of information and knowledge despite the emergency/evolution. An important derivative work of ELOCC was the Dynamic Lines of Collaboration (DLOC) principle, done by Yoon, Velasquez, Partridge, and Nof (2008), Hao Zhong and Nof (2015), Hao Zhong (2016), and Ferialdy (2016). The DLOC principle focuses more on the general concept of propagating services in CPSs, and on the configuration, allocation, and scheduling of traveling agents to fulfill the services of the CPSs. The ELOCC and DLOC principles are important inspirations for this work. It is noted, however, that the CCT previous research has not studied in-depth the complex interactions between the client system, response mechanisms, and disruption propagation, which is specifically addressed in this work.

A summary of CCT research relevant to the RDP problem is provided in Table 2.7.

Table 2.7. Summary of CCT research relevant to the RDP problem

CCT Research and Principle(s)	Relevant aspects	Details
ECPR	Detection and prevention	Error and conflicts can be harmful disruptions to the client systems, with the possibility to propagate. Detection and prevention are important response activities.
ARS	Important problem domain	Agricultural plant diseases can be infectious/epidemical and is an important RDP problem. Detection and prevention are important response activities.
RBT	Important problem domain	Supply network disruption is an important RDP problem. Information sharing and situation awareness at the local-level and global-level are critical for resilience.
BMP	Taxonomic framework	A taxonomic framework provides a systematic foundation for identification and characterization of problem components and interactions.
ELOCC	Emergency response	Emergency responses have similar characteristics to disruption response.
DLOC	Response coordination	Configuration, coordination, and scheduling are important design and control strategies.

2.4 Concluding Remarks

CHAPTER 2 reviews the different research articles on the different RDP problem domains, problem formulation, as well as potential design and control principles for the RDP problems. From the literature survey, it is observed that recent research on the RDP problems in different research domains is well-established and diverse, but there exists no framework to connect the different domains and characterize the different components and interactions. There is indeed no "big picture above the different RDP problems": no modeling framework to characterize and categorize the different components of the problem. This justifies the development of a systematic framework to unify the different problem domains, per Research Questions 1 and 2. Furthermore, there is a dearth of general design and control principles and philosophies that can be applied to different domain-specific RDP problems. This justifies the development of the CLOC principle, per Research Question 3. The CRDP framework and the CLOC principle are presented in the next chapter.

CHAPTER 3. METHODOLOGY – THE CRDP FRAMEWORK AND THE CLOC DESIGN AND CONTROL PRINCIPLE

This chapter presents the new Collaborative Response to Disruption Propagation (CRDP) framework for the characterization and the categorization of the aforementioned RDP problems and models. The framework, which is one important original contribution of this dissertation, is developed at the PRISM (Production, Robotics, and Integration Software for Manufacturing and Management) Center of Purdue University. The CRDP framework culminates in the development of the Teamwork Integration Evaluation TIE/CRDP software (presented in APPENDIX A). With respect to the CCT research, the CRDP framework was inspired by the PRISM Best Matching Taxonomy, the RBT principle, and the ECPR principle.

This chapter also presents the Covering Lines of Collaboration (CLOC) principle, which is developed to guide and support the analysis and decision-making process against disruption propagation. The CLOC principle, which is another important original contribution of this dissertation, is also developed at the PRISM Center of Purdue University. With respect to the CCT research, the CLOC principle was inspired by and is a continuation of the ECPR principle, the RBT principle, the ELOCC principle, and the DLOC principle.

3.1 The CRDP Framework

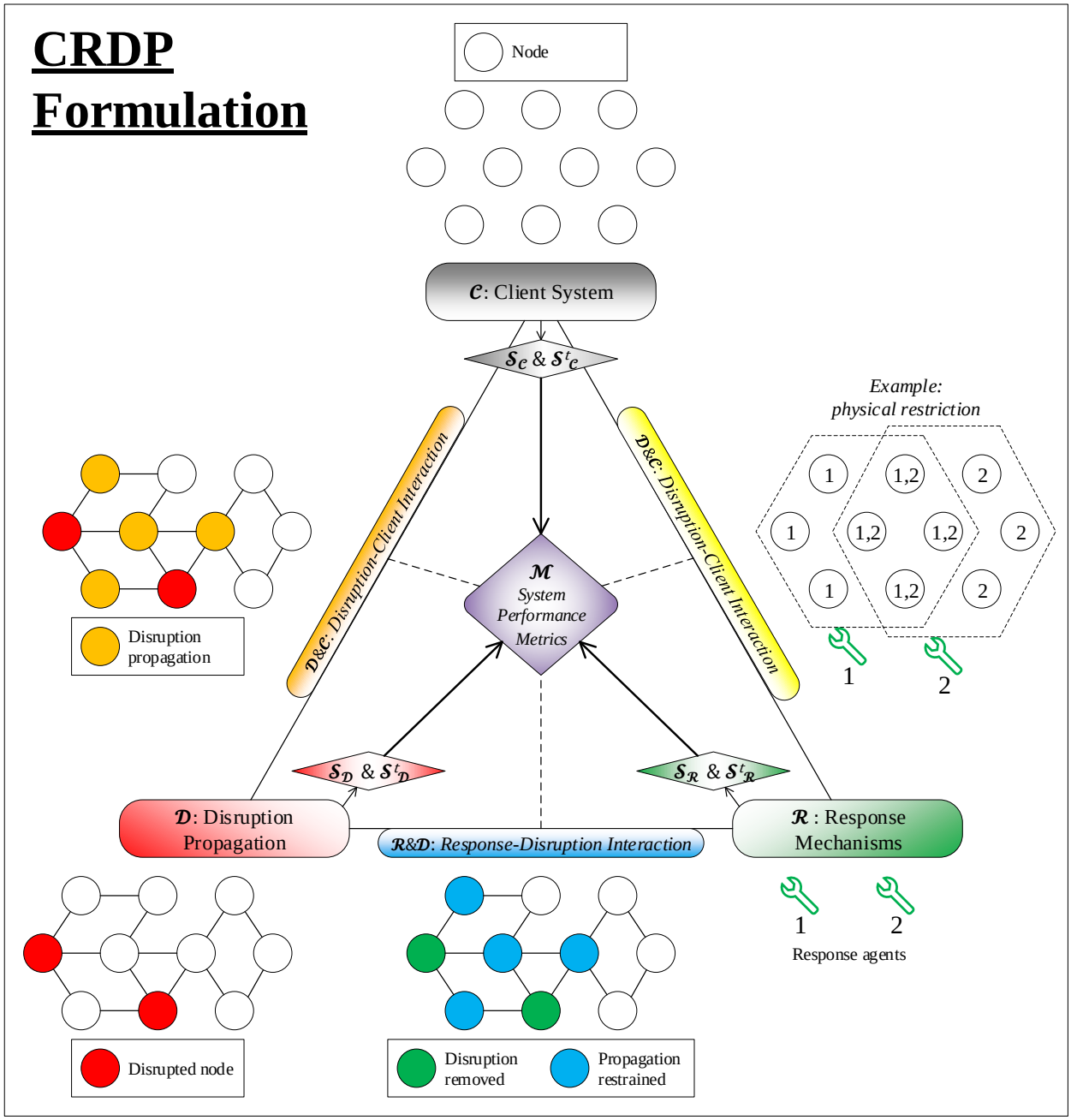

Figure 3.1. The CRDP framework

The Collaborative Response to Disruption Propagation (CRDP) framework is a unifying framework for the characterization and the categorization of different RDP problems and models. The CRDP framework consists of:

- 3 components: the client system $\boldsymbol{\mathcal{C}}$, the response mechanisms $\boldsymbol{\mathcal{R}}$, and the disruption propagation $\boldsymbol{\mathcal{D}}$.
- 3 interactions between the components: the client-response interaction $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$, the response-disruption interaction $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$, and the disruption-client interaction $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$.
- 2 types of decision spaces for each component: the design decision $\boldsymbol{\mathcal{S}}_{\#}$ and the control decision $\boldsymbol{\mathcal{S}}_{\#}^{t}$.
- And the set of system performance metrics $\boldsymbol{\mathcal{M}}$.

By employing the CRDP framework, an RDP problem can be systematically formulated into an RDP model, with each modeling element being characterized and classified into components, interaction, decision spaces, and system performance metrics.

Each part of the CRDP framework is discussed in the subsections below.

3.1.1 The CRDP components

To accurately reflect the characteristics of an RDP problem, the corresponding RDP model needs to contain the appropriate modeling elements: entities/objects, attributes, relations, and events. Entities refer to the physical and/or virtual objects of the model, and each entity can have zero, one, or many attributes. The relations refer to the connections and/or interactions between different entities and/or attributes of the same types or different types. The events refer to the important additions, removals, and changes to the model's aforementioned modeling elements. For the remaining of the chapter, the term "modeling element" refers to all the aforementioned types of modeling elements.

One important observation of this work is that certain modeling elements of an RDP model can be characterized into three distinct CRDP components that have different roles in the corresponding RDP model.

The first CRDP component is $\boldsymbol{\mathcal{C}}$: Client system. The client system (illustrated in Figure 3.2) consists of entities that are subjected to harmful disruptions and their propagation, making the

client system and its entities the victims of the disruptions. Any entity that fits entirely into this description and not exhibiting characteristics fitting $\mathcal{R}$ or $\mathcal{D}$ (as explained below) should be classified in $\mathcal{C}$. Examples include the building complexes ($\mathcal{C}$) on fire ($\mathcal{D}$) which can spread, plants ($\mathcal{C}$) affected by diseases ($\mathcal{C}$) that are contagious, computer networks ($\mathcal{C}$) attacked by malware ($\mathcal{D}$) that can propagate within the network.

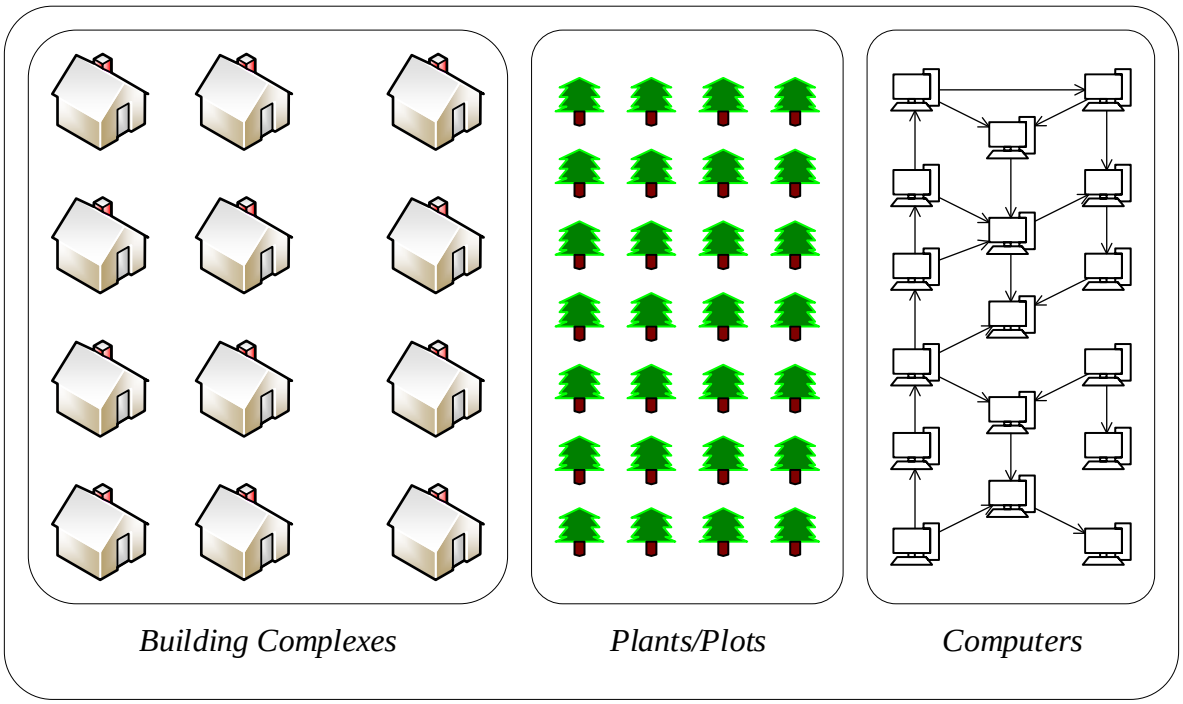

Figure 3.2. Illustration of client system examples

The second CRDP component is $\mathcal{R}$: Response mechanisms. The response mechanisms consist of the entities that can strictly reduce or eliminate the existences and/or impacts of the disruptions and their propagation. The response mechanisms are the rescuers and/or protectors of the client system. Any entity and attribute that fits entirely into this description should be classified in $\mathcal{R}$. The response types include, but are not limited to, disruption detection, disruption prevention, disruption removal, disruption quarantine, client system repair, and a combination thereof. Examples of response mechanisms include firefighting and sprinkler ($\mathcal{R}$) against fire ($\mathcal{D}$) affecting building complexes ($\mathcal{C}$), detection and quarantine ($\mathcal{R}$) against plant disease ($\mathcal{D}$) potentially affecting plants ($\mathcal{C}$), firewall ($\mathcal{R}$) against computer malware ($\mathcal{D}$) attacking computer

networks ($\mathcal{C}$). Examples of response mechanisms in different problem contexts are given in Table 3.1.

Table 3.1. Examples of response mechanisms

Problem context	**Response mechanisms**	
	Static/Reactive	**Dynamic/Active**
Fire spreading	Water sprinkler (*removal*); Insulation (*prevention*); Smoke detector (*detection*)	Firefighters (*removal*); Fire engines (*removal*); Helicopters (*removal*)
Infectious plant disease	Immunization (*prevention*); Static sensor (*detection*); Pesticides (*prevention*); Herbicides (*prevention*)	Active detection (*detection*); Disease cure (*client repair*) Quarantine (*removal*)
Propagating computer malware	Firewalls (*detection + prevention*); Security protocols (*combination*)	Active scanning (*detection*); Quarantine (*removal*)
Supply network disruption	Backup inventory (*mitigation*); Network topology (*mitigation*)	Negotiation (*mitigation*); Alternative supply/distribution (*mitigation*); Breakdown repair (*client repair*)

The third CRDP component is $\mathcal{D}$: Disruption propagation. This term refers to both the disruptions and their propagation. $\mathcal{D}$ consists of the entities that can cause negative impacts and/or propagate on the client system. The disruptions are the aggressors/attackers that are harmful to the client systems. Any such entity and attribute exhibiting such characteristics should be classified in $\mathcal{D}$. Examples (illustrated in Figure 3.3) include spreading fire, infectious diseases, propagating malware, supply/demand drastic changes. Disruption propagation includes two types of propagation: disruption existences and disruption impacts. The cases of fire spreading and infectious plant disease belong to the category of disruption existence propagation, because the fire spreads and creates additional fires, and diseases infect nearby plants and create additional diseases. Certain types of supply network disruptions, e.g. production breakdowns and worker strikes, belong to the category of propagation of disruption impacts, which does not propagate the breakdowns and strikes to the suppliers and customers.

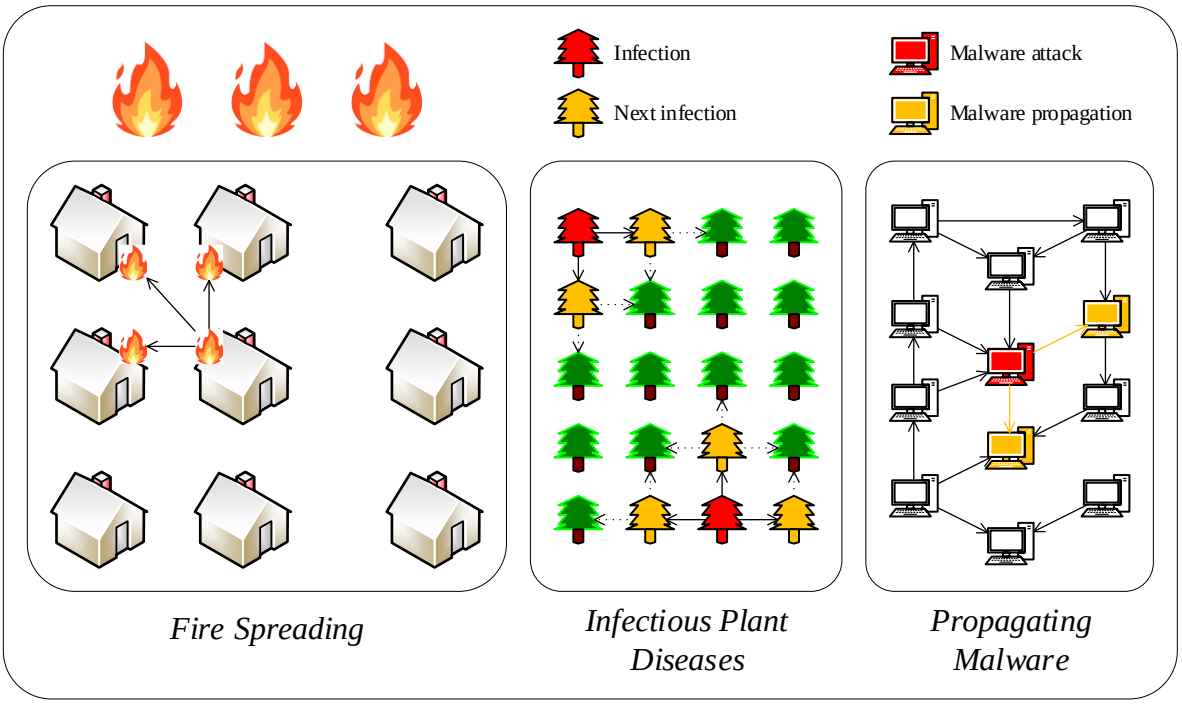

Figure 3.3. Illustration of disruption examples

The components $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ are separated (even though both $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ are antithetical to the disruptions $\boldsymbol{\mathcal{D}}$) due to the distinct functions that each component has. Furthermore, the existence of $\boldsymbol{\mathcal{R}}$ is strictly beneficial to $\boldsymbol{\mathcal{C}}$, while certain conditions and/or configurations of $\boldsymbol{\mathcal{C}}$ (i.e. plants placed in closer proximity) can worsen the damages from $\boldsymbol{\mathcal{D}}$. Furthermore, there are cases where $\boldsymbol{\mathcal{C}}$ is passive (such as the plants) and $\boldsymbol{\mathcal{R}}$ does not necessarily know everything about $\boldsymbol{\mathcal{C}}$, which necessitates the distinction between $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$.

Without loss of generality, given any $\boldsymbol{\mathcal{C}}$, the negative impact of having $\boldsymbol{\mathcal{D}}$ without $\boldsymbol{\mathcal{R}}$ is defined as $f(\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{R}}) \in \mathbb{R}_{\geq 0}$. The CRDP framework states that:

- The harmful impact from $\boldsymbol{\mathcal{D}}$ when $\boldsymbol{\mathcal{R}}$ is not available is always positive, or $f(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} = \emptyset) > 0$.
- Furthermore, due to disruption propagation, the rate of increase (with respect to time) of harmful impact is also non-negative $f'(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} = \emptyset) \geq 0$, with the equal sign occurring when propagation is saturated.

- With available response $\boldsymbol{\mathcal{R}} \neq \emptyset$, the harmful impact is non-negative, or $f(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} \neq \emptyset) \geq 0$, and the equal sign occurs when the responses $\boldsymbol{\mathcal{R}}$ fully prevent $\boldsymbol{\mathcal{D}}$ from affecting $\boldsymbol{\mathcal{C}}$.
- Furthermore, the harmful impact with $\boldsymbol{\mathcal{R}}$ is less than or equal to the harmful impact without $\boldsymbol{\mathcal{R}}$, or $f(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} \neq \emptyset) \leq f(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} = \emptyset)$, and the equal sign occurs when the responses $\boldsymbol{\mathcal{R}}$ are entirely ineffective.
- Also, the rate of increase (with respect to time) of harmful impact with $\boldsymbol{\mathcal{R}}$ is less than or equal the case without $\boldsymbol{\mathcal{R}}$, or $f'(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} \neq \emptyset) \leq f'(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} = \emptyset)$, and the equal sign occurs when the responses $\boldsymbol{\mathcal{R}}$ are entirely ineffective. It is noted that $f'(\boldsymbol{\mathcal{D}} \neq \emptyset, \boldsymbol{\mathcal{R}} \neq \emptyset)$ can be negative, equal to zero, or positive.

A brief illustration of a simple RDP model classified in accordance with the CRDP framework is provided in Figure 3.4.

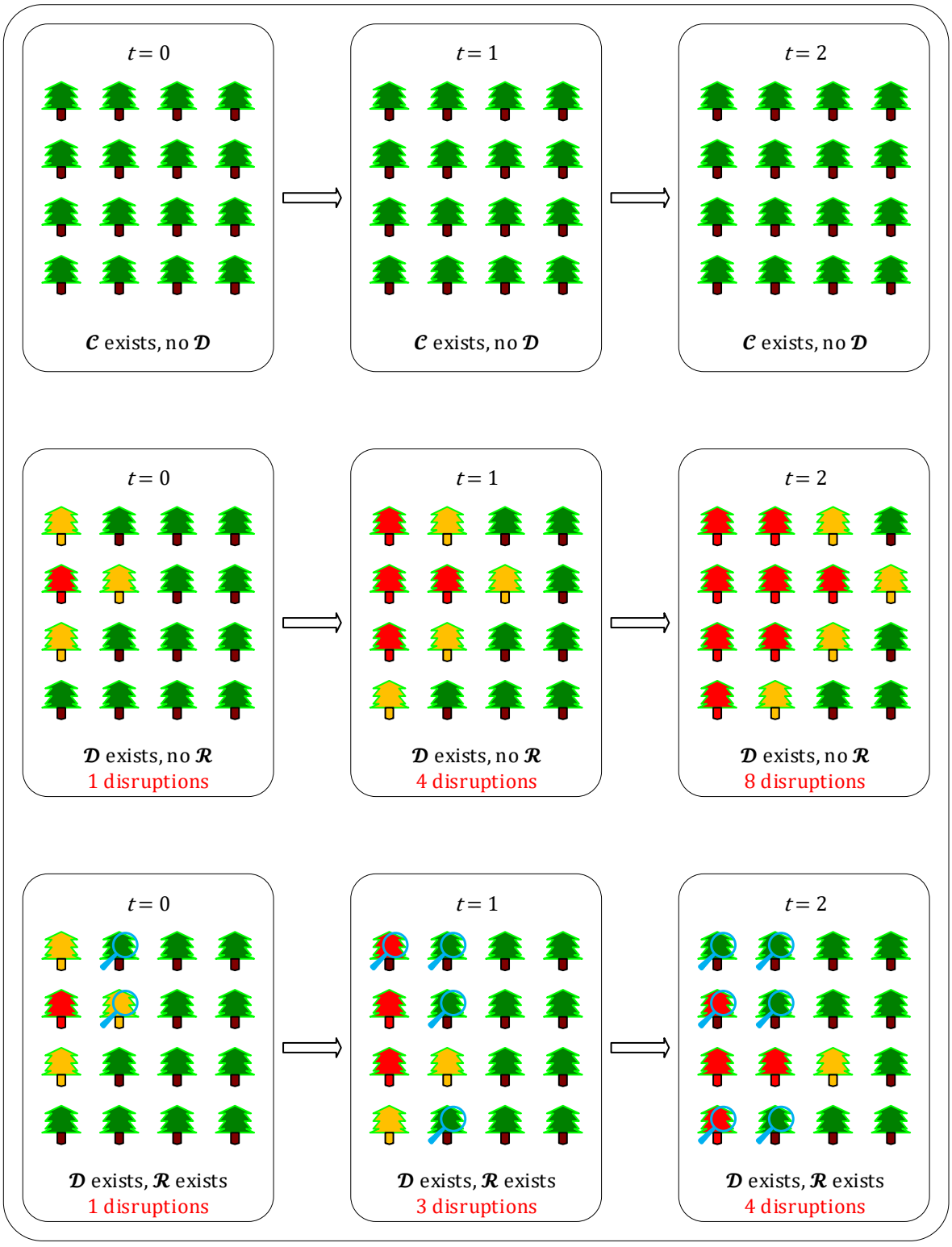

Figure 3.4. Brief RDP example illustration

The modeling entities belonging to $\boldsymbol{\mathcal{C}}$ but exhibiting behaviors suitable for $\boldsymbol{\mathcal{R}}$ or $\boldsymbol{\mathcal{D}}$ should instead be assigned to $\boldsymbol{\mathcal{R}}$ or $\boldsymbol{\mathcal{D}}$, respectively. Such examples include fire sprinklers (which physically belong to the building complex) that can extinguish fire; the anti-disease characteristics of the plants (which physically belong to the plants) that prevents diseases; and internal errors/conflicts of a computer network (which originated from the computer network) that can cause deadlock within the network. This reassignment is necessary because the functionality and purpose of the modeling element are of interest to the corresponding RDP model. The CRDP components are summarized in Table 3.2, which lists the components and their accompanying details and examples.

Table 3.2. Summary of CRDP components

CRDP Components	Details	Examples
$\boldsymbol{\mathcal{C}}$: client system	The system subjected to harmful disruptions and their propagation.	Buildings; plants; computer networks; supply firms…
$\boldsymbol{\mathcal{R}}$: response mechanisms	The entities that can reduce and/or mitigate the existences and/or impacts of disruption propagation. Types include (but not limited to): detection, prevention, removal, repair	Fire sprinkler, firefighters, firetrucks; disease immunization, disease detection; firewall against malware…
$\boldsymbol{\mathcal{D}}$: disruption propagation	The entities that can cause harmful impacts to the client system and can propagate their existences/impacts. Types include: propagation of disruption existences, propagation of disruption impacts.	Fire; plant disease; propagating computer malware; supply network supply/demand disruption…

3.1.2 The CRDP interactions

The three components $\boldsymbol{\mathcal{C}, \mathcal{R}, \mathcal{D}}$ do not exist in isolation, and can exhibit complex dynamics and interactions with each other. Any such relationship can be further classified into three CRDP interactions. The complex dynamics and relations that affect more than one CRDP component ($\boldsymbol{\mathcal{C}, \mathcal{R}, \mathcal{D}}$) are classified as a CRDP interaction.

The client-response interaction: $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$. Modeling elements that fit into both $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ should be classified into $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$. This type of interaction includes (but is not limited to) physical access restrictions/limitations, location familiarity, system compatibility, etc. The interaction $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$ is relevant to both the design/redesign decisions of $\boldsymbol{\mathcal{C}}$ and the resource allocation and configuration decisions of $\boldsymbol{\mathcal{R}}$. Any modeling element that is directly related to both $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ should be classified into $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$. Examples of this type of interaction include:

1. Fire spreading case: firefighting activities ($\boldsymbol{\mathcal{R}}$) involves physical traveling to different building complexes ($\boldsymbol{\mathcal{C}}$). Certain buildings could be further away and take more time to travel to ($\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$) , and/or could require different firefighting mechanisms ($\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$) (firefighter vs firetruck vs helicopters) to address.
2. Plant disease detection case: detection activities ($\boldsymbol{\mathcal{R}}$) could be hindered by the arrangements ($\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$) of the plants ($\boldsymbol{\mathcal{C}}$). Certain plants could be surrounded by other plants, thus are harder to investigate.
3. Malware in computer network case: malware detection and prevention ($\boldsymbol{\mathcal{R}}$) could be affected by different operating system configurations ($\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$) of different computers ($\boldsymbol{\mathcal{C}}$). sections.

The disruption-client interaction: $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$. Modeling elements that fit into both $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$ should be classified into $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$. This type of interaction includes (but is not limited to) the propagation of disruption through the connections and/or proximities between the components of the client system; different disruption propagation speed/intensity. The interaction $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$ is relevant to both the design/redesign decisions of $\boldsymbol{\mathcal{C}}$ and possibly the targeting decisions of $\boldsymbol{\mathcal{D}}$ (in the case the disruptions are supported by intelligence, as with the case of autonomous malware). Examples of this type of interaction include:

1. Fire spreading case: fire ($\boldsymbol{\mathcal{D}}$) can spread from one room/building ($\boldsymbol{\mathcal{C}}$) to another through the proximity ($\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$) between the rooms/buildings.
2. Plant disease detection case: undetected disease ($\boldsymbol{\mathcal{D}}$) can infect nearby ($\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$) plants ($\boldsymbol{\mathcal{C}}$).
3. Malware in computer network case: malware ($\boldsymbol{\mathcal{D}}$) can propagate between computers ($\boldsymbol{\mathcal{C}}$) that are directly connected ($\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$) with each other.

The response-disruption interaction: $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$. Modeling elements that fit into both $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{D}}$ should be classified into $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$. This type of interaction includes (but is not limited to) the response mechanisms' reduction and/or prevention of disruption propagation; compatibility of response methods towards different disruption types. The interaction $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$ is relevant to both the resource allocation/configuration decisions of $\boldsymbol{\mathcal{R}}$ and possibly the targeting decisions of $\boldsymbol{\mathcal{D}}$ (in the case the disruptions are supported by intelligence, as with the case of autonomous malware). Examples of this type of interaction include:

1. Fire spreading case: firefighting activities $(\boldsymbol{\mathcal{R}})$ can extinguish the fire $(\boldsymbol{\mathcal{D}})$ and prevent fire spreading $(\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}})$ to nearby buildings.
2. Plant disease detection case: accurate detection and quarantine $(\boldsymbol{\mathcal{R}})$ of disease $(\boldsymbol{\mathcal{D}})$ could prevent disease propagation $(\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}})$ between plants.
3. Malware in computer network case: successful firewall $(\boldsymbol{\mathcal{R}})$ would prevent both malware $(\boldsymbol{\mathcal{D}})$ and propagation of malware $(\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}})$.

The above examples' modeling elements could appear to include elements of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$, but should be classified into $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$ instead because they signify the characteristics of preventing $(\boldsymbol{\mathcal{R}})$ imminent and potential future disruption $(\boldsymbol{\mathcal{D}})$ propagation $(\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}})$. More complex modeling elements could be classified into more than one category of interaction, but this multi-classification should be employed sparingly to avoid confusion. The interactions are in Table 3.3. Summary of CRDP interactions, which lists the interactions and their accompanying details and examples.

Table 3.3. Summary of CRDP interactions

CRDP Interactions	Description	Examples
$\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$: client-response	Modeling elements and relationships that involve both $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$	Physical constraints; Location constraints; System compatibility
$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$: disruption-client	Modeling elements and relationships that involve both $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$	Disruption propagation through proximity; Disruption propagation through connections;
$\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$: response-disruption	Modeling elements and relationships that involve both $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{D}}$	Removal/mitigation of current disruptions; Prevention of disruption occurring; Prevention of potential disruption propagation;

The classification of modeling elements into the three interaction types $\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$, $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$, and $\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$ provides insights into the complex interactions between the CRDP components $\boldsymbol{\mathcal{C}}$, $\boldsymbol{\mathcal{R}}$, and $\boldsymbol{\mathcal{D}}$. These insights can be used to support analysis and decision makings to improve the client system's resilience and response mechanisms' effectiveness against disruptions.

3.1.3 The CRDP decision spaces

The RDP problem and its corresponding RDP model usually involve intelligent decision-making. In the context of this work, the term *decision space* is equivalent to the term solution space in mathematical optimization (after decision/solution constraints are taken into consideration). The plural term *decisions* is occasionally used in the place of *decision space*. A specific decision or solution is explicitly called *decision option* or *solution*. Examples include the pesticide/herbicide allocation decisions and detection activities allocation decisions in the plant disease problem. The decisions can be classified into two types of decisions:

The CRDP design decisions: $\boldsymbol{\mathcal{S}}_{\#}$. The decisions of this type are made off-line, and cannot be changed during real-time. The CRDP component(s) directed involved in a type of decision space is/are denoted in the # symbol of the $\boldsymbol{\mathcal{S}}_{\#}$ notation. Examples include:

1. Fire spreading case: location decisions ($\boldsymbol{S_{\mathcal{C}}}$) of building complexes ($\boldsymbol{\mathcal{C}}$), allocation decisions ($\boldsymbol{S_{\mathcal{R}}}$) of static firefighting mechanisms ($\boldsymbol{\mathcal{R}}$) such as sprinklers, fire extinguishers, firefighting water supply.
2. Plant disease detection case: plant ($\boldsymbol{\mathcal{C}}$) location decisions ($\boldsymbol{S_{\mathcal{C}}}$), herbicide/pesticide ($\boldsymbol{\mathcal{R}}$) amount and location decisions ($\boldsymbol{S_{\mathcal{R}}}$).
3. Malware in computer network case: firewall ($\boldsymbol{\mathcal{R}}$) configuration decisions ($\boldsymbol{S_{\mathcal{R}}}$), and if malware ($\boldsymbol{\mathcal{D}}$) is controlled by intelligence, malware off-line/initial targeting decisions ($\boldsymbol{S_{\mathcal{D}}}$).

The CRDP control decisions: $\boldsymbol{S_{\#}^{t}}$. The decisions of this type are made on-line and during real-time. The CRDP component(s) directed involved in a type of decision space is/are denoted in the # symbol of the $\boldsymbol{S_{\#}^{t}}$ notation. Examples include:

1. Fire spreading case: Allocation decisions ($\boldsymbol{S_{\mathcal{R}}^{t}}$) of dynamic firefighting mechanisms ($\boldsymbol{\mathcal{R}}$) such as firefighters, firetrucks, and helicopters.
2. Plant disease detection case: detection ($\boldsymbol{\mathcal{R}}$) decisions ($\boldsymbol{S_{\mathcal{R}}^{t}}$) to check for possible disease.
3. Malware in computer network case: active malware scanning and investigation decisions ($\boldsymbol{S_{\mathcal{R}}^{t}}$), and if malware $\boldsymbol{\mathcal{D}}$ is controlled by intelligence, malware on-line targeting/propagation decisions ($\boldsymbol{S_{\mathcal{D}}^{t}}$).

3.1.4 The CRDP system performance metrics

Both $\boldsymbol{S_{\#}}$ and $\boldsymbol{S_{\#}^{t}}$ are subjected to one or more optimization goal(s)/objective(s), which is/are classified into the CRDP system performance metrics: $\boldsymbol{\mathcal{M}}$. These refer to the optimization objective function(s) and/or binary (yes/no, true/false) goals. System performance metrics can be real-time metrics and/or aftermath metrics (after disruptions are eliminated, resources run out, or at the end of the simulation). Examples include:

1. Fire spreading case: total aftermath damage ($\boldsymbol{\mathcal{M}}$), number of buildings unharmed ($\boldsymbol{\mathcal{M}}$).
2. Plant disease detection case: total infections detected ($\boldsymbol{\mathcal{M}}$), total redundant detections (no infection detected in plants) ($\boldsymbol{\mathcal{M}}$).

3. Malware in computer network case: total system performance loss during disruptions ($\boldsymbol{\mathcal{M}}$), maximum fraction of system compromised ($\boldsymbol{\mathcal{M}}$), whether the system is fully recovered (yes/no) ($\boldsymbol{\mathcal{M}}$).

While the further classification of $\boldsymbol{\mathcal{M}}$ into $\boldsymbol{\mathcal{M}}_{\boldsymbol{\mathcal{C}\&\mathcal{R}}}$ (the client system and response mechanisms perspective) and $\boldsymbol{\mathcal{M}}_{\boldsymbol{\mathcal{D}}}$ (disruption perspective) is possible, such a classification is unnecessary due to the fact that the existence of $\boldsymbol{\mathcal{D}}$ is antithetical, and in some cases, adversarial, to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$. This means a metric $x \in \mathbb{R}$ viewed from the perspective of $\boldsymbol{\mathcal{D}}$ would be the opposite number $(-x)$ when viewed from the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ and vice versa.

The list of CRDP formulation categories is provided in Table 3.4, which lists the categories with their notations and brief details.

Table 3.4. Summary of CRDP formulation categories

CRDP Formulation Category	Notations	Details
CRDP Components	$\boldsymbol{\mathcal{C}}$: client system	Modeling elements pertaining to $\boldsymbol{\mathcal{C}}$
	$\boldsymbol{\mathcal{R}}$: response mechanisms	Modeling elements pertaining to $\boldsymbol{\mathcal{R}}$
	$\boldsymbol{\mathcal{D}}$: disruption propagation	Modeling elements pertaining to $\boldsymbol{\mathcal{D}}$
CRDP Interactions	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$: client-response interaction	Modeling elements and relationships pertaining to both $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$
	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$: disruption-client interaction	Modeling elements and relationships pertaining to both $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$
	$\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$: response-disruption interaction	Modeling elements and relationships pertaining to both $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{D}}$
CRDP Decision Space	$\boldsymbol{\mathcal{S}}_{\#}$: design decision space	Decisions made off-line, can't be changed during real-time.
	$\boldsymbol{\mathcal{S}}_{\#}^{t}$: control decision space	Decisions made on-line and during real-time.
CRDP System Performance Metrics	$\boldsymbol{\mathcal{M}}$: system performance metric	Optimization objective functions and/or binary (true/false, yes/no) goals.

3.2 The Covering Lines of Collaboration (CLOC) Principle

Making the appropriate design decisions $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and control decisions $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$ of the response mechanisms is necessary to achieve desirable outcomes of system performance metrics $\boldsymbol{\mathcal{M}}$. Therefore, the new Covering Lines of Collaboration (CLOC) principle is developed to guide and support the analysis and decision-making process of the response mechanisms against disruption propagation. The CLOC principle supports the development of CCT analytics and protocols specifically for the RDP problems. This dissertation does not consider the decisions of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{D}}$, namely $\boldsymbol{\mathcal{S}_{\mathcal{C}}, \mathcal{S}_{\mathcal{C}}^{t}, \mathcal{S}_{\mathcal{D}}, \mathcal{S}_{\mathcal{D}}^{t}}$, due to the high level of additional analysis complexity.

In the context of this work, the term *analytics* $\boldsymbol{\mathcal{A}}$ refers to the analysis/analyses of the state of the system and the modeling elements, and returns a set of quantifiable variables and/or conjectures that can guide the development of the protocols $\boldsymbol{\mathcal{P}}$. The use of analytics is necessary due to the complex dynamics and interactions involved, rendering exact mathematical analysis and proofs difficult to achieve.

In the context of this work, the term *protocol* $\boldsymbol{\mathcal{P}}$ refers to the workflow decision-making set(s) of rules, procedures, and possibly algorithms for multiple interacting agents. The protocols are pre-defined and agreed-upon, and are used to determine the decision options for $\boldsymbol{\mathcal{S}_{\#}}$ and $\boldsymbol{\mathcal{S}_{\#}^{t}}$. The protocols are different from traditional scheduling policies in that: Protocols are necessary for sophisticated workflow problems where the agents can encounter task assignment conflicts. A protocol can be different from an algorithm in that a protocol can involve interaction between different agents and processes following the protocol(s) (Nof et al., 2015). In that sense, a protocol is more general than an algorithm and involves more complex interactions between different entities and processes.

The CLOC principle consists of three guidelines. Each CLOC guideline serves as a set of instructions to be applied to a specific RDP problem and its corresponding model. After the CRDP framework is employed to formulate an RDP problem into a model, the CLOC principle can be applied to develop appropriate analytics and protocols to support the response decisions. The three CLOC guidelines are as follows.

1. The first CLOC guideline – network modeling of disruption propagation. This guideline specifies that the components/subsystems of the client system are modeled as nodes and the potential disruption propagation directions are modeled as edges. The resulting network can be further analyzed using network analysis to understand the disruption propagation behavior.
2. The second CLOC guideline – restraining disruption propagation. The existence of response mechanisms, by definition, already reduces the harmful impacts of the disruptions and prevent further disruption propagation. This guideline specifies the analysis of the propagation-restraining effect and the utilization of this knowledge to develop analytics and protocols to support the response decisions.
3. The third CLOC guideline – collaboration between response mechanisms to ensure coverage. This guideline specifies the development of collaborative analytics and protocols to support the response decisions. The response decisions do not exist in isolation because each response decision has a propagation-restraining effect that affects the direction and severity of disruption propagation. Collaboration between response mechanisms can ensure the coverage of the propagation-restraining effect, improving the performance of the response mechanisms.

3.2.1 The first CLOC guideline – network modeling of disruption propagation

The first CLOC guideline specifies the network modeling of the disruption propagation behavior. When a disruption affects an entity of the client system $\boldsymbol{\mathcal{C}}$, this disruption has the potential to propagate to other entities of $\boldsymbol{\mathcal{C}}$. The propagation, in general, is not arbitrary (per the findings of CHAPTER 2), and can be identified and characterized. This means the components/subsystems of the client system can be modeled as nodes and the potential disruption propagation directions can be modeled as edges. The resulting network can be further analyzed using network analysis to improve situation awareness and to better understand the disruption propagation behavior. The following sub-guidelines are stated:

CLOC 1a. Each entity of the client system $\boldsymbol{\mathcal{C}}$ is modeled as a node n belonging to the set of nodes $\mathrm{NL} = \{n_0, n_1, \dots\}$. Different characteristics of each node $n \in \mathrm{NL}$ is assigned an attribute relevant for that node.

CLOC 1b. An edge e connecting two nodes $n_i, n_j \in$ NL is defined as a potential disruption propagation direction between the two nodes, based on $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$. The set of edges is defined as EL $= \{e_0, e_1, \dots\}$.

CLOC 1c. An edge can either be directed (also called unidirectional) or undirected (also called bidirectional). A directed edge $e = (n_i, n_j)$ means a disruption affecting n_i can propagate to n_j, but not necessarily in the opposite direction. An undirected edge $e = \{n_i, n_j\}$ (alternative notation $e = (n_i, n_j) \cup (n_j, n_i)$) means a disruption affecting n_i can propagate to n_j and vice versa.

CLOC 1d. Edge attributes are defined to accurately model the behavior of disruption propagation.

CLOC 1e. If a disruption affecting a node n_i can result in a propagation to node n_j, an edge (n_i, n_j) must be created to reflect this propagation. Each edge represents a potential disruption propagation, which does not necessarily guarantee the propagation occurring (due to response mechanisms and/or stochasticity, for example).

The examples used in Figure 1.1 can be converted to their corresponding network models, as shown in Figure 3.5.

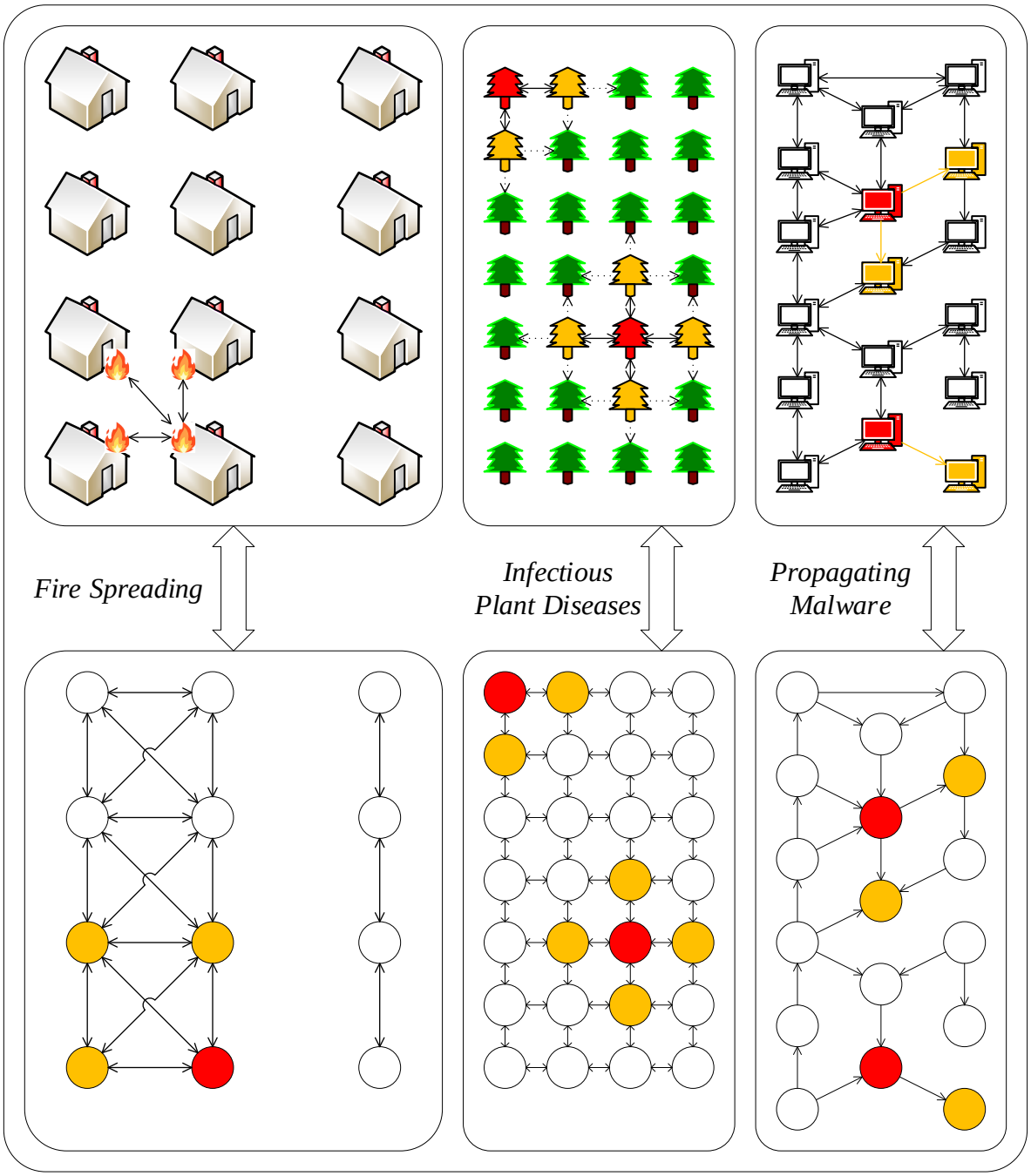

Figure 3.5. Network modeling of disruption propagation examples

Employing network modeling enables the usage of network analysis to analyze the potential impacts of disruptions and their propagation. Potential network analysis methods that can be applied include:

CLOC 1f. Degree centrality analysis: For each node $n \in \mathrm{NL}$, the node's out-degree, or $\mathrm{OD}(n) = \left|\{e = (n_i, n_j) \in \mathrm{EL}: n_i \equiv n\}\right|$, denotes the local-level disruption propagation potential of the node, if node n is disrupted. If the disruption propagation behavior is affected by the edge's attribute(s), a node's disruption propagation potential value can be defined as $\mathrm{NDPP}(n) = \sum_{e=(n_i,n_j)}^{\{e\in\mathrm{EL}:n_i\equiv n\}} \mathrm{EDPM(e)}$ with $\mathrm{EDPM}(e) \in \mathbb{R}$ defined as an attribute of the edge that quantifies this effect.

CLOC 1g. Distance analysis: Based on EL, the weights $W(e) \in \mathbb{R}_{\geq 0}$ of all $e \in \mathrm{EL}$ are available (for unweighted networks, $W(e) = 1, \forall e \in \mathrm{EL}$), a distance matrix can be computed using the Floyd-Warshall algorithm with complexity $\mathrm{O}(n^3)$ (Floyd, 1962; H. Zhong & Nof, 2020). The distances between nodes can then be analyzed.

CLOC 1h. Network centrality analysis: A node's centrality measure is one potential indicator of the node's importance with respect to the topology of the network. Each centrality measure is defined differently, and an appropriate centrality measure should be compatible with the specific disruption propagation mechanism(s) as defined by $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$. Potential network centrality measures include degree centrality (Warshall, 1962), betweenness centrality (Freeman, 1978), closeness centrality (Freeman, 1978), harmonic centrality (Bavelas, 1950), percolation centrality (Marchiori & Latora, 2000) amongst others. Figure 3.6 provides three examples of centrality measures.

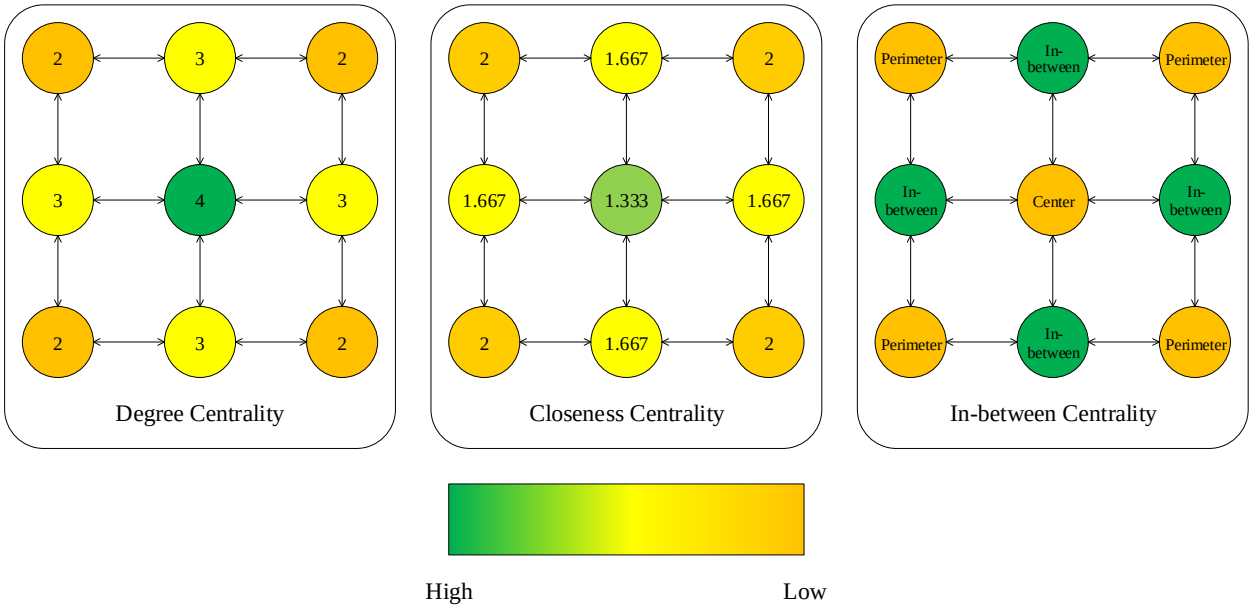

Figure 3.6. Different centrality measures: degree, closeness, and in-between

3.2.2 The second CLOC guideline – restraining disruption propagation

The second CLOC guideline specifies the analysis of one particularly significant $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$ interaction: the disruption propagation restraining effect of the response mechanisms. This guideline also specifies the utilization of this knowledge to develop analytics and protocols to support the response decisions. The existence of response mechanisms, by definition, already reduces the harmful impacts of the disruptions and prevent further disruption propagation. This means the existence and/or deployment of response mechanisms ($\boldsymbol{\mathcal{R}}$) can restrain potential disruption ($\boldsymbol{\mathcal{D}}$) propagation from affecting the nodes ($\boldsymbol{\mathcal{C}}$) near the disrupted nodes. If disruptions are not prevented, not timely detected, and/or not timely removed, propagation will occur and further damage the client system.

Without loss of generality, given set of nodes NL, set of directed edges EL, and a disruption $d(n)$ affecting a node $n \in \mathrm{NL}$. This means there exist the set of potential disruption propagations $\mathrm{DP}(n)$ to the succeeding nodes n_j of node n:

$$d(n) \Rightarrow \exists \mathrm{DP}(n) = \left\{ d(n_j) : \forall n_j \in \mathrm{NL} \middle| \exists e = (n_i \equiv n, n_j) \in \mathrm{EL} \right\} \quad (1)$$

The existence of response mechanisms on node n, denoted as $r(n)$, can prevent and/or reduce the disruption propagation (by the definition of response mechanisms), due to the removal and/or weakening of $d(n)$. This means

$$r(n) \Rightarrow d(n) \searrow \Rightarrow |DP(n)| \searrow \quad (2)$$

Thus, the second CLOC guideline is stated as:

CLOC 2. Response decisions should target the disruptions with the greatest disruption propagation potential, in order to restrain disruption propagation. This guideline has two merits: (a) reduce the rate of increase of the disruptions' harmful effects; (b) reduce the workload of the response mechanisms and/or improve response resources efficiency.

An illustration of the disruption propagation restraining effect is provided in Figure 3.7. In this example, response to the north-west node leads to the restraint of 2 propagations, whereas response to the east node leads to the restraint of 3 propagations. Selecting the east node for response not only reduces the rate of increase of disruptions, but also reduces the additional workload of the response mechanisms.

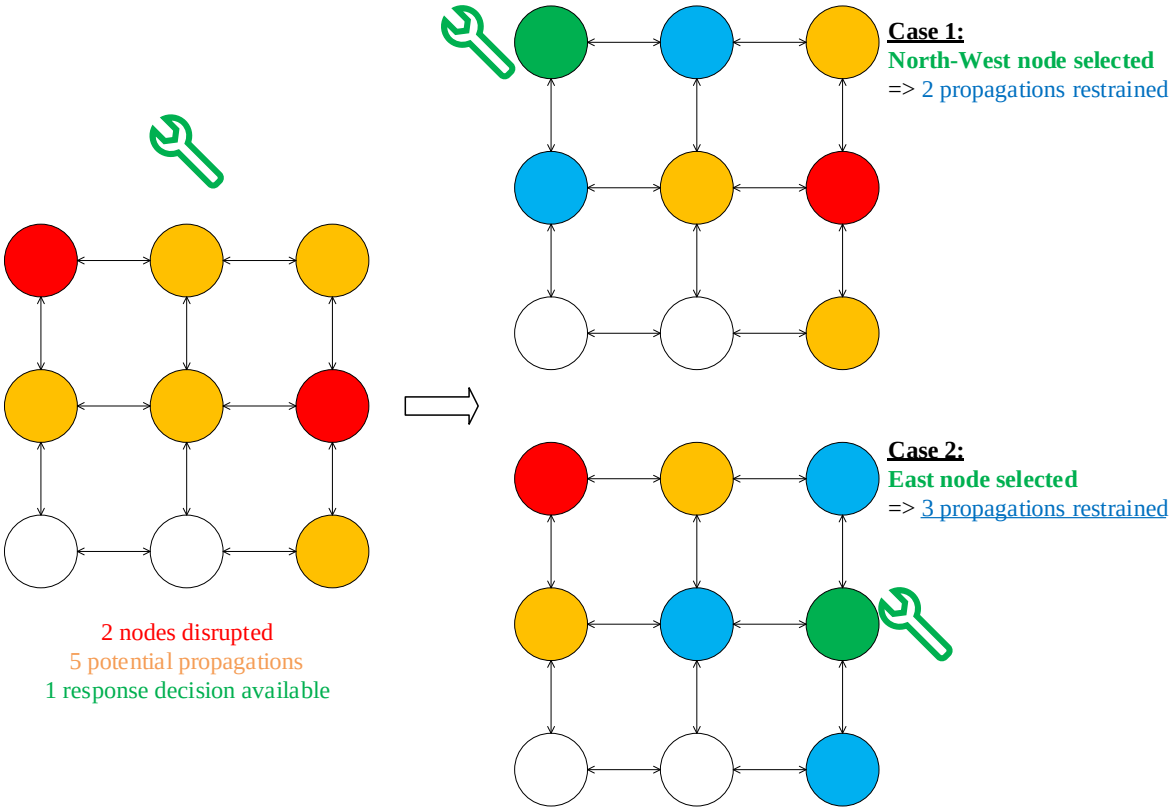

Figure 3.7. Disruption propagation restraining effect

3.2.3 The third CLOC guideline – collaboration between response decisions to ensure coverage

The third guideline of the CLOC principle specifies the development of collaborative analytics and protocols to support the response decisions $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and/or $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$. The response decisions do not exist in isolation because each response decision has a propagation-restraining effect that affects the direction and severity of disruption propagation. Collaboration between response mechanisms can ensure coverage of the propagation-restraining effect, improving the performance of the response mechanisms. The following sub-guidelines are stated:

CLOC 3a. The insights gained from analyzing the CRDP components $\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{D}}$, and the CRDP interactions $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$ with respect to $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and/or $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$ as well as $\boldsymbol{\mathcal{M}}$ can be utilized to develop the analytics $\boldsymbol{\mathcal{A}}$. The analytics $\boldsymbol{\mathcal{A}}$ refer to any formula, programming procedure/function, or quantities that provide insights and can be employed to support the decision-making processes of $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$.

CLOC 3b. The analytics $\boldsymbol{\mathcal{A}}$ can be used to support the development of the decision-making and allocation protocols $\boldsymbol{\mathcal{P}}$ to guide the response decisions $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and/or $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$. A CRDP protocol in $\boldsymbol{\mathcal{P}}$ refers to a pre-defined and agreed-upon set of steps that guides and/or selects the one or more decision spaces in $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and/or $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$. As noted before, a protocol can be different from an algorithm (and can also include algorithms and policies).

CLOC 3c. Particularly, the protocols $\boldsymbol{\mathcal{P}}$ that support $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and/or $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$ should consider ongoing decisions that are in effect, ensuring the *coverage* of the propagation-restraining effect. Possible methods include: best matching protocols (Piraveenan, Prokopenko, & Hossain, 2013), centrality-based allocation (pioneered by (Mohsen Moghaddam & Nof, 2016)), minimizing disruption propagation (Hao Zhong & Nof, 2015), and strategic location analysis.

A brief example is given in Figure 3.8, with 20 nodes, 31 bidirectional edges, 5 initial disruptions, 2 response decisions available each t. Response option 1 at nodes 6 and 14 leads to only 2 effective restraints of disruption propagation, protecting only nodes 2 and 18, with nodes 5, 7, 8, 13, and 15 being disrupted through propagation. Option 2 leads to 3 effective restraints, protecting nodes 8, 13, and 18, providing better response coverage.

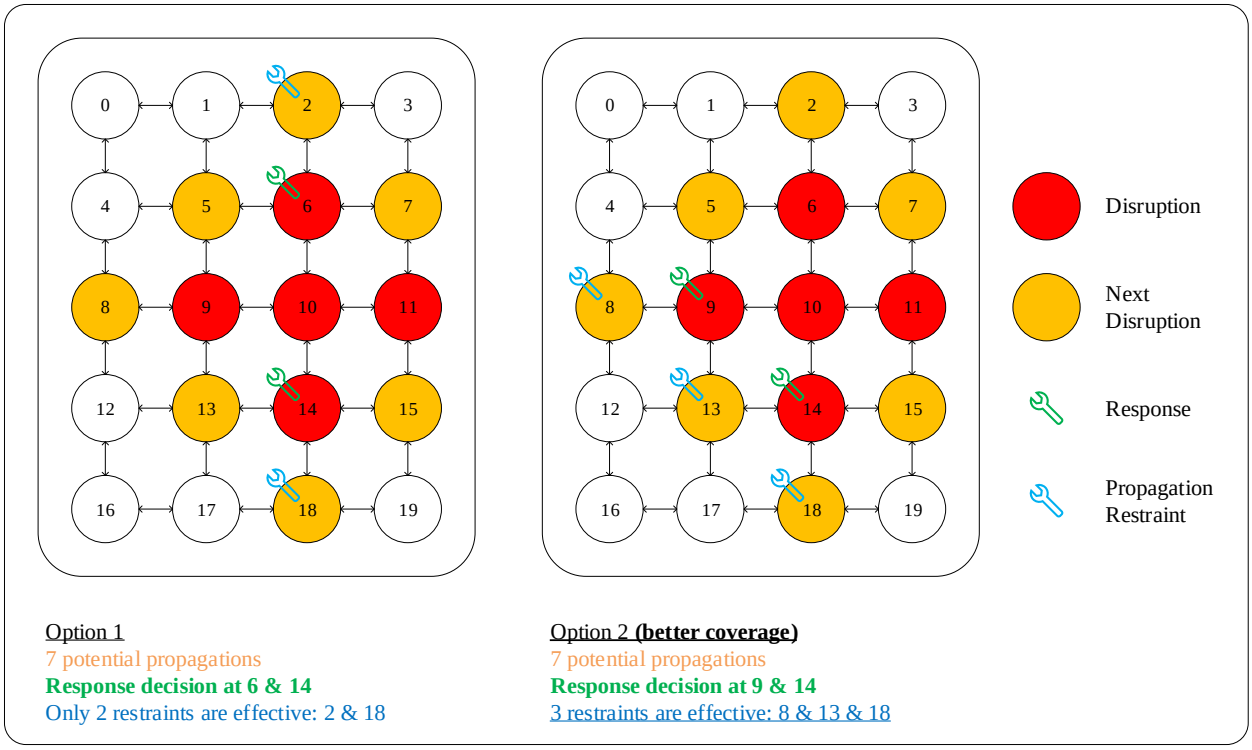

Figure 3.8. Example of the CLOC sub-guideline on coverage

The CLOC guidelines should not be executed in a purely sequential manner, and the revisiting of previous steps is highly recommended to improve the quality of the analytics $\boldsymbol{\mathcal{A}}$ and protocols $\boldsymbol{\mathcal{P}}$. The CLOC guidelines are summarized in Table 3.5.

Table 3.5. Summary of CLOC guidelines

CLOC Guideline	Details
CLOC 1a	Model components of $\boldsymbol{\mathcal{C}}$ as nodes of a network.
CLOC 1b	Model potential disruption propagation $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ directions as edges of the network.
CLOC 1c	Differentiate directed (unidirectional) edges vs undirected (bidirectional) edges if necessary.
CLOC 1d	Define edge attributes if necessary.
CLOC 1e	Include all possible potential disruption propagation as edges.
CLOC 1f	Apply degree centrality analysis.
CLOC 1g	Apply network distance analysis if necessary.
CLOC 1h	Apply appropriate network centrality measures if necessary.
CLOC 2	Restrain disruptions with the most propagation potential
CLOC 3a	Combine insights from CRDP components, CRDP interactions, CLOC 1a-h, CLOC 2 to meaningful analytics $\boldsymbol{\mathcal{A}}$ to support $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$
CLOC 3b	From the analytics $\boldsymbol{\mathcal{A}}$, design protocols $\boldsymbol{\mathcal{C}}$ to support decision-making.
CLOC 3c	The design protocols $\boldsymbol{\mathcal{C}}$ should consider collaboration and synergy between the decisions in $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ and/or $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$.

3.3 Case Studies – A Synopsis

Seven case studies, each with a corresponding RDP model, have been conducted on seven different RDP problems. Three case studies are presented and discussed in this dissertation, with the other case studies were published in the literature as journal articles and conference proceedings. By demonstrating the systematic classification and specification of the CRDP framework to each specific RDP model, the three case studies serve as validation of the CRDP framework and its accompanying robust classification procedures, addressing Research Questions 1 and 2. The case studies also demonstrate the application of the CLOC principle to each RDP problem, addressing Research Question 3. Each case study contains a generalized RDP model that retains the critical domain-specific system behaviors, which allows adaptation to more complex and domain-specific cases. A summary of each case study is provided below.

Case 1 – Collaborative Detection of Unknown Disruption Propagation (CDUD). This case study focuses on disruptions that are unknown to the client system and response mechanisms, and can also propagate. The response mechanisms are detection agents that can scan the nodes in the client system and determine whether the nodes are disrupted. Based on the CLOC principle, the CDUD analytics and protocols are developed to support the collaborative detection decisions. The advanced CDUD protocols outperform baseline protocols by 9.7% to 32.8% with statistical significance, depending on scenarios.

Case 2 – Collaborative Strategic Prevention of Disruption Propagation (CSPD). This case study focuses on the response mechanisms strategic allocation decisions that cannot be changed after disruptions occur, without knowledge of where the disruptions would attack. The response mechanisms are strategic allocations that can protect specific nodes and the neighboring nodes. Based on the CLOC principle, the CSPD analytics and protocols are developed to support the decisions of the strategic allocations. The advanced CSPD protocols outperform the baseline protocols by 31.1% to 56.6% with statistical significance, depending on scenarios.

Case 3 – Collaborative Teaming and Coordination of Dynamic Repair Agents (CTCD). This case study focuses on two types of response decisions: the off-line teaming decisions and the on-line coordination decisions. A team of repair agents must be selected to be on standby, and the selection cannot be changed when disruptions occur. This case study also focuses on the recurring nature of disruption propagation, which means disruptions can re-propagate to nodes that are no longer protected by the response mechanisms. Based on the CLOC principle, the CTCD analytics and protocols are developed to support the teaming decisions and the coordination decisions. The advanced CTCD teaming protocols outperform the baseline protocols by 2.1% to 12.1%, and the advanced CTCD coordination protocols outperform the baseline and less advanced protocols by at least 50%, all cases with statistical significance.

Case 4 – Cyber-augmented Manufacturing Networks (Nguyen & Nof, 2019a). This case study focuses on manufacturing network disruptions. The disruption propagation of interest is the propagation of disruption impacts, instead of disruption existence. The response mechanisms involved are repair agents, with their repair decisions supported by network centrality, disruption, and flow analytics. The advanced protocols developed outperform the baseline protocols by 7.6% to 33.7%, with statistical significance.

Case 5 – Collaborative Response to Disruption Propagation in Cyber-physical Systems and Complex Networks (Nguyen & Nof, 2018). This case study focuses on the recurring disruption propagation in cyber-physical systems. This case study is the predecessor of the CTCD case, and only involves the coordination decisions of the response mechanisms. The response mechanisms involved are repair agents, with their repair decisions supported by the analysis of the disruption propagation restraining effects and smart task allocation between different response agents. The advanced protocols developed outperform the baseline protocols significantly (up to 90%) with lower numbers of agents, and perform similarly with higher numbers of agents.

Case 6 – Collaborative Response to Disruption Propagation with the Established Lines of Collaboration (Nguyen & Nof, 2019a). As the predecessor of the CSPD case, this case study also focuses on the strategic allocation decisions of the response mechanisms. This case study also inspires the development of the third CLOC guideline, which recommends the coverage-based allocation of strategic resources.

Case 7 – Collaborative Response to Disruption Propagation against Evolving Disruptions (Nguyen & Nof, 2019b). This case study is a continuation of the CRDP/DSS case, using the same problem settings with Case 5. This investigates the possibility of disruptions learning the structure of the client system, and evolve their targeting protocols over time.

A comparison of the formulations of the seven cases is provided in Table 3.6.

Table 3.6. Comparison of formulations of the seven case studies

Case	**Client system $\mathcal{C}$**	**Response mechanisms $\mathcal{R}$**	**Disruption propagation $\mathcal{D}$**	**Edge - Disruption propagation behaviors (edges) $\mathcal{D}$&$\mathcal{C}$**
Case 1 – CDUD	ARS, CPSs, computer networks	Dynamic; Detection	Node attribute: binary (0 or 1)	Directed, unweighted edges
Case 2 – CSPD	ARS, building complexes	Strategic; Prevention	Node attribute: 0 to 1	Undirected, unweighted edges
Case 3 – CTCD	CPSs, computer networks	Dynamic; Repair	Node attribute: binary (0 or 1)	Directed, weighted edges
Case 4 – (Nguyen, Nair, & Nof, 2019)	Manufacturing networks	Dynamic; Repair	Node attribute: 0 to 1	Directed, weighted edges
Case 5 – (Nguyen & Nof, 2018)	CPSs, computer networks	Dynamic; Repair	Node attribute: binary (0 or 1)	Directed, weighted edges
Case 6 – (Nguyen & Nof, 2019a)	CPSs	Strategic; Prevention	Node attribute: 0 to 1	Undirected, unweighted edges
Case 7 – (Nguyen & Nof, 2019b)	CPSs, computer networks	Dynamic; Repair	Node attribute: binary (0 or 1)	Directed, weighted edges

CHAPTER 4. CASE 1 – COLLABORATIVE DETECTION OF UNKNOWN DISRUPTION PROPAGATION

4.1 CDUD Description

One important and common property of disruption is the characteristic of being unknown to the client system and/or the response mechanisms. The unknown disruptions are especially devastating to the client system because they can propagate while remaining undetected, with the disruptions having a head start in propagation until detected and responded to. This problem is relevant to the agriculture plant disease settings and the propagating malware problem in computer networks. In the agricultural setting, diseases can be difficult to detect due to the large scale of the system, and components (plants and animals) are often tightly packed, allowing diseases to spread within a population (Nguyen et al., 2019). Similarly, in computer networks, malware that successfully infiltrates through the firewalls (due to security vulnerabilities and/or backdoors) can propagate to connected computers. The two aforementioned settings inspire the formulation of an RDP model to address the aspects "unknown" and "propagation" of the disruptions.

Following the CRDP framework, the Collaborative Detection of Unknown Disruptions (CDUD) model is formulated with the components, interactions, decision space, and system performance metrics. The entities of the client system $\boldsymbol{\mathcal{C}}$ are presented by nodes, each of which can represent a plant, a group of plants, a computer, or a device. The nodes are susceptible to disruptions in $\boldsymbol{\mathcal{D}}$, with the disruptions capable of propagating to nearby (agricultural) or connected (computer) nodes. The disruption information of a node is not available to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ until detection activities are performed on the node. The response mechanisms in $\boldsymbol{\mathcal{R}}$ employed for this case are active and dynamic detection agents (detection robots, computer scanning agents) that can accurately detect the unknown disruptions. Detected disruptions are then removed, and no further propagation is possible, whereas undetected infections can continue to propagate diseases to nearby/connected nodes. With respect to $\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$, response agents in $\boldsymbol{\mathcal{R}}$ can respond to disruptions affecting any nodes in $\boldsymbol{\mathcal{C}}$. Response agents in $\boldsymbol{\mathcal{R}}$ are also aware of the locations of the nodes in $\boldsymbol{\mathcal{C}}$ and the potential disruption propagation directions $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$, which is expected in agricultural systems and computer networks. These $\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$ aspects are applicable to the problem contexts of agricultural systems and

computer networks, where cooperation between $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ is possible and necessary. With respect to $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$, a disruption in $\boldsymbol{\mathcal{D}}$ affecting a node in $\boldsymbol{\mathcal{C}}$ can propagate to the succeeding neighboring nodes, thus, network modeling can be applied, per the first CLOC guideline. Four important aspects of $\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$ are noted: (i) the response agents in $\boldsymbol{\mathcal{R}}$ are not aware of the location of the disruptions in $\boldsymbol{\mathcal{D}}$ until the disruptions are detected; (ii) the response agents in $\boldsymbol{\mathcal{R}}$ can accurately scan, detect, and remove a disruption in $\boldsymbol{\mathcal{D}}$; (iii) the response activity in $\boldsymbol{\mathcal{R}}$ protect a node from future disruptions; and (iv) response activity in $\boldsymbol{\mathcal{R}}$ prevent future disruption propagation from occurring.

Within the scope of the CDUD model, one decision type $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$ is investigated: the dynamic response activities of the response agents in $\boldsymbol{\mathcal{R}}$. Because the disruptions can propagate in real-time, the $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$ decisions compete against disruption propagation. Inaccurate and/or ineffective response can lead to more severe propagation $\boldsymbol{\mathcal{D}}$, worsening the workload of $\boldsymbol{\mathcal{R}}$ and the system viability of $\boldsymbol{\mathcal{C}}$. Three system performance metrics $\boldsymbol{\mathcal{M}}$ are of interest: total performance loss $\boldsymbol{\mathcal{M}}_1$, maximum performance loss $\boldsymbol{\mathcal{M}}_2$, and maximum disruption propagation $\boldsymbol{\mathcal{M}}_3$. The metric total performance loss $\boldsymbol{\mathcal{M}}_1$ measures the total over-time performance loss of $\boldsymbol{\mathcal{C}}$ due to disruptions. $\boldsymbol{\mathcal{M}}_1$ is relevant when the client system $\boldsymbol{\mathcal{C}}$ is still expected to be operational under disruption, such as in the case of computer networks. The metric maximum performance loss $\boldsymbol{\mathcal{M}}_2$ measures the highest level of performance loss ever occurred. The metric maximum disruption propagation $\boldsymbol{\mathcal{M}}_3$ indicates the proportion of the client system $\boldsymbol{\mathcal{C}}$ ever affected by disruptions. Both $\boldsymbol{\mathcal{M}}_2$ and $\boldsymbol{\mathcal{M}}_3$ are important to consider because certain disruption types incur long-term or permanent damages that cannot be recovered from, such as loss of sensitive information in computer networks and the loss of agricultural production in ARSs.

A summary of the CDUD model formulation is provided in Table 4.1.

Table 4.1. Summary of CDUD description

CRDP Formulation Category	Item	Details
CDUD Components	$\boldsymbol{\mathcal{C}}$: client system	Nodes representing plants, groups of plants, computers, or devices (depending on the context).
	$\boldsymbol{\mathcal{R}}$: response mechanisms	Response agents that can detect and quarantine nodes.
	$\boldsymbol{\mathcal{D}}$: disruption propagation	Disruptions are unknown to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ until detected. Disruptions can propagate if left unresponded to.
CDUD Interactions	$\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$: client-response interaction	Response agents can respond to all nodes in $\boldsymbol{\mathcal{C}}$. Response agents are aware of potential disruption propagation directions.
	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$: disruption-client interaction	A disruption affecting a node can propagate to the node's succeeding nodes, cause more disruption(s).
	$\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$: response-disruption interaction	Response agents are not aware of which nodes are disrupted until detected. Response activity to a node removes disruption. Response activity to a node prevents future disruptions from affecting this node.
CDUD Decision Space	$\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$: response dynamic decision	Response decisions are allocated in real-time. One response decision per response agent.
CDUD System Performance Metrics	$\boldsymbol{\mathcal{M}}_1$: total performance loss	Total over-time disruptions affecting the client system.
	$\boldsymbol{\mathcal{M}}_2$: maximum performance loss	Maximum number of disruptions affecting the client system at one point.
	$\boldsymbol{\mathcal{M}}_3$: maximum disruption propagation	The largest proportion of the client system affected by disruptions.

4.2 CDUD Formulation

Based on the CDUD model description, the CRDP formulation of the CDUD model is as follows. The CDUD model is simulated using the one variation of the TIE/CRDP software presented in APPENDIX A. Each timestep $t \in \mathbb{Z}_{\geq 0}$ is a discrete timestep. The entities and attributes are given in Table 4.2.

Table 4.2. Entities and attributes of the CDUD model

Type	Entity/Attribute and Explanation	CRDP domain
Input	$\boldsymbol{\mathcal{C}}$: $\mathrm{NL} = \{n_0, n_1, \dots\}$ Set of nodes, with each node $n \in \mathrm{NL}$ representing a component of the client system.	$\boldsymbol{\mathcal{C}}$
Input	$\boldsymbol{\mathcal{R}}$: $\mathrm{AL} = \{a_0, a_1, \dots\}$ Set of agents, with each agent $a \in \mathrm{AL}$ representing an active and dynamic detection agent capable of accurately detect and quarantine disruptions.	$\boldsymbol{\mathcal{R}}$
Input	$\boldsymbol{\mathcal{D}}$: $\mathrm{DPID} \in [0,1]$ Probability of initial infection/disruption affecting each node independently.	$\boldsymbol{\mathcal{D}}$
Input	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$: $\mathrm{EL} = \{e_0, e_1, \dots\}$ The set of directed edges, with each directed edge $e = (n_i, n_j)$ representing a potential disruption propagation direction from node n_i to node n_j and from node n_j to node n_i.	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
The following attributes are defined for each node $\boldsymbol{n} \in$ NL		
Dynamic	$\mathrm{NOS}(n,t) \in \{0,1\}$ Node n's observed status at time t, with value 0 denoting that node n is not observed, and 1 if otherwise. Default value of 0.	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
Dynamic	$\mathrm{NDS}(n,t) \in \{0,1\}$ Node n's disruption status at time t, with value 0 denoting that node n is not disrupted, and 1 if otherwise. Default value of 0.	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
Derived	$\mathrm{NPNL}(n) \subset \mathrm{NL}$ Node n's set of preceding nodes, which includes all nodes with an edge pointing the nodes to n. $\mathrm{NPNL}(n) = \{n_i \in \mathrm{NL}: \exists (n_i, n) \in \mathrm{EL}\}$	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
Derived	$\mathrm{NSNL}(n) \subset \mathrm{NL}$ Node n's set of succeeding nodes, which includes all nodes with an edge pointing from n to the nodes. $\mathrm{NSNL}(n) = \{n_j \in \mathrm{NL}: \exists (n, n_j) \in \mathrm{EL}\}$	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
The following attributes are defined for each agent $\boldsymbol{a} \in$ AL		
Decision	$\mathrm{ASN}(a,t) \in \mathrm{NL}$ Agent a's selected node to perform response activity at time $t > 0$. This decision type is made without information of $\mathrm{NDS}(n,t), \forall n \in \mathrm{NL}: \mathrm{NOS}(n, t-1) = 0$, and with full information of all other entities and variables.	$\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{\boldsymbol{t}}$

The nodes in the client system $\boldsymbol{\mathcal{C}}$ are represented by the set of nodes NL. The response agents in $\boldsymbol{\mathcal{R}}$ are represented by the set of agents AL. The disruptions in $\boldsymbol{\mathcal{D}}$ are represented by the attributes $\mathrm{NDS}(n,t)$, and $\mathrm{NDS}(n,0)=1$ is caused by DPID. The allocations of response activities to the nodes in NL are represented by $\mathrm{ASN}(a,t)$, and the statuses of current and past responses are represented by $\mathrm{NOS}(n,t)$. Per the first CLOC guideline, the disruption propagation directions are represented by the set of directed edges EL, which is known to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{D}}$. This leads to the derivation of the set of preceding nodes $\mathrm{NPNL}(n)$ and the set of succeeding nodes $\mathrm{NSNL}(n)$ for each node.

Following the specification of Table 4.1, the CDUD system performance metrics are given in Table 4.3.

Table 4.3. System performance metrics of the CDUD model

System Performance Metric	CRDP domain
$\mathrm{PL}(t)=\frac{\sum_{n}^{\mathrm{NL}}\max\left(0,\mathrm{NDS}(n,t)-\mathrm{NOS}(n,t)\right)}{\lvert\mathrm{NL}\rvert}$ Performance loss, which denotes the total fraction of the client system affected by undetected disruptions at time t. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, $\mathrm{PL}(t)$ is to be minimized.	$\boldsymbol{\mathcal{M}}$
$\mathrm{TPL}=\sum_{t}\mathrm{PL}(t)$ Total performance loss (TPL), which is the over-time total performance loss during a simulation replication. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, TPL is to be minimized.	$\boldsymbol{\mathcal{M}}_1$
$\mathrm{MPL}=\max_{t}\mathrm{PL}(t)$ Maximum performance loss (MPL), which is the highest level of performance loss within a simulation replication. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, MPL is to be minimized.	$\boldsymbol{\mathcal{M}}_2$
$\mathrm{MDP}=\frac{\lvert\{n\in\mathrm{NL}:\sum_t\mathrm{NDS}(n,t)\geq 1\}\rvert}{\lvert\mathrm{NL}\rvert}$ Maximum disruption propagation (MDP), which denotes the fraction of the nodes of the client system that have ever been disrupted throughout a simulation replication. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, MDP is to be minimized.	$\boldsymbol{\mathcal{M}}_3$

A small example of a CDUD case is provided in Figure 4.1.

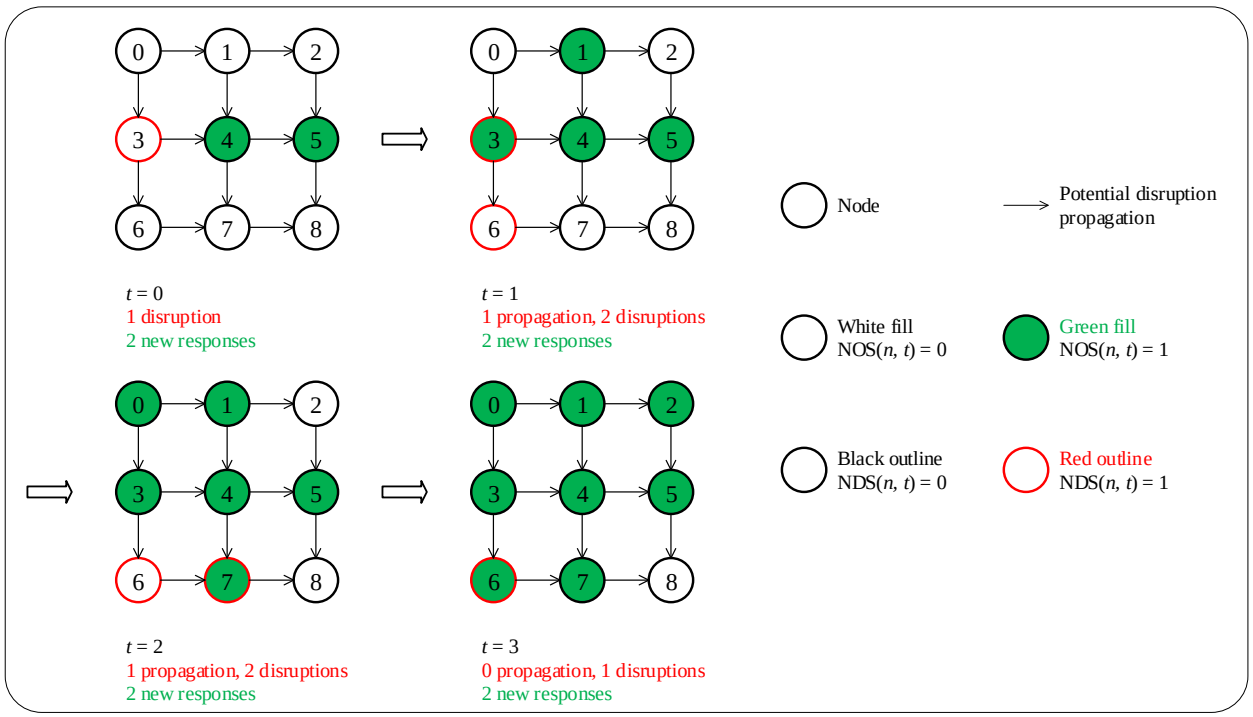

Figure 4.1. CDUD example

In this example, there are 9 nodes numbering 0 to 8 in NL, with 12 directed edges representing potential disruption propagation directions. There are two response agents in AL.

At time $t = 0$,

Node n_3 is disrupted, thus $\mathrm{NDS}(n_3, 0) = 1$.

The agents are assigned to n_4 and n_5 by $\boldsymbol{S}_{\boldsymbol{R}}^{t}$, thus $\mathrm{ASN}(a_0, 0) = n_4, \mathrm{ASN}(a_1, 0) = n_5$.

This leads to $\mathrm{NOS}(n_4, 0) = 1, \mathrm{NOS}(n_5, 0) = 1$.

At $t = 1$,

The disruption at n_3 propagates to n_6 due to $\mathrm{NDS}(n_3, 0) = 1, \mathrm{NOS}(n_3, 0) = 0$, and $\mathrm{NOS}(n_6, 0) = 0$,

The disruption at n_3 does not propagate to n_4 due to $\mathrm{NOS}(n_4, 0) = 1$.

The simulation continues until the end of $t = 3$, where all disruptions are detected and removed. In this example, $\boldsymbol{\mathcal{M}}_1 = \text{TPL} = \sum_t \text{PL}(t) = 1/3, \boldsymbol{\mathcal{M}}_2 = \text{MPL} = \max_t\big(PL(t)\big) = 1/9$, and $\boldsymbol{\mathcal{M}}_3 = \text{MDP} = 1/3$. The complete simulation pseudocode of the CDUD model is provided in Table 4.4.

Table 4.4. Simulation pseudocode of the CDUD model

Step	Pseudocode	CRDP domain
Step 1	$t \leftarrow 0$, Initialize NL, AL, DPID, EL	Simulation
Step 2	$\forall n \in \text{NL}$, if $\text{unif}(0,1) < \text{DPID}, \text{NDS}(n,t) \leftarrow 1$	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
Step 3	For $t \coloneqq 1$ to $t_{\max}$	Simulation
Step 3.1	Decide $\text{ASN}(a,t)$ for all $a \in \text{AL}$	$\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{\boldsymbol{t}}$
Step 3.2	foreach $n \in \text{NL}$ if $\exists a \in \text{AL}: \text{ASN}(a,t) \equiv n$ $\text{NOS}(n,t) \leftarrow 1$ $\text{NDS}(n,t) \leftarrow 0$ else $\text{NOS}(n,t) \leftarrow \text{NOS}(n,t-1)$ next n	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$
Step 3.3	foreach $\forall n \in \text{NL}$ if $\text{NDS}(n,t-1) = 1 \; and \; \text{NOS}(n,t) = 0$ $\forall n_j \in \text{NSNL}(n): \text{NOS}(n,t) = 0$ $\text{NDS}\big(n_j,t\big) \leftarrow 1$ else if $\text{NOS}(n,t) = 1$ $\text{NDS}(n,t) \leftarrow 1$ else $\text{NDS}(n,t) \leftarrow \text{NDS}(n,t-1)$ next n	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$
Step 3.4	$\text{PL}(t) \leftarrow \frac{\sum_n^{\text{NL}} \text{NDS}(n,t)}{\lvert\text{NL}\rvert}$	$\boldsymbol{\mathcal{M}}$
Step 4	Compute $\boldsymbol{\mathcal{M}}$	$\boldsymbol{\mathcal{M}}$

In Table 4.4, Step 1 initializes $t \leftarrow 0$, the main inputs of the CDUD model, which includes the set of nodes NL, the set of agents AL, the initial disruption probability DPID affecting each node, and the set of directed edges EL representing potential disruption propagation directions.

Step 2 initializes the disruptions based on probability DPID. Each node $n \in \text{NL}$ affected by disruptions will have the attribute $\text{NDS}(n,t) = 1$.

Step 3 begins the dynamic simulation of the system. The simulation ends when the maximum time $t_{\max}$ is reached or the system state no longer changes.

Step 3.1, which decides $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$, involves the agents $a \in \mathrm{AL}$ selecting the nodes $n \in \mathrm{NL}$ to respond to while not having information regarding $\mathrm{NDS}(n,t)$. These decisions can be supported by analytics and protocols, which are discussed below in the following section.

Step 3.2 actuates the decisions made in Step 3.1 and involves updating the attributes $\mathrm{NOS}(n,t)$ and $\mathrm{NDS}(n,t)$ according to the decisions $\mathrm{ASN}(a,t)$. If a node has been responded to at t, its $\mathrm{NOS}(n,t) \leftarrow 1, \mathrm{NDS}(n,t) \leftarrow 0$, removing the disruption affecting the node (if any). Otherwise, $\mathrm{NOS}(n,t) \leftarrow \mathrm{NOS}(n,t-1)$. This means a node that has been responded to in the past (or not) would retain the observation status.

Step 3.3 propagates the undetected disruptions to nodes that are not responded to. If a node n's disruption status $\mathrm{NDS}(n,t) = 1$, it will propagate disruptions to its succeeding nodes $n_j \in \mathrm{NSNL}(n)$ that were not observed, i.e. $\mathrm{NOS}(n_j,t) = 0$. This step also maintains the disruption statuses of the undetected disruptions, making $\mathrm{NDS}(n,t) \leftarrow \mathrm{NDS}(n,t-1)$, while removing detected disruptions due to $\mathrm{NOS}(n,t) = 1$, making $\mathrm{NDS}(n,t) \leftarrow 0$.

Step 3.4 calculates the performance loss $\mathrm{PL}(t)$ of the system at time t. Then, the simulation returns to step 3, incrementing $t \leftarrow t+1$.

Step 4 marks the end of one simulation replication and calculates $\boldsymbol{\mathcal{M}}_1$,$\boldsymbol{\mathcal{M}}_2$, and $\boldsymbol{\mathcal{M}}_3$.

4.3 CDUD Analytics and Protocols

In this subsection, the CDUD analytics and protocols are developed based on the CLOC principle to support the decision-making of $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$.

The analysis of the CDUD model and decision space is as follows. For the purpose of analysis, the event that node n is disrupted at time t is denoted as $D(n,t)$, which is equivalent to $\mathrm{NDS}(n,t) = 1$. Because the disruption status $\mathrm{NDS}(n,t)$ is not known to the decision space $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$: $\mathrm{ASN}(a,t)$, the probability function $P(n,t) = \Pr\big(\mathrm{NDS}(n,t)\big) \in [0,1]$ is defined to assist decision-making ($\Pr(\ \ )$ refers to probability).

Based on the definition of $\boldsymbol{\mathcal{D}}$ and Step 2 of Table 4.4, and calling $\mathrm{DPID} = k$ for short, it is observed that $P(n,0) = k$, and the probability of each $D(n,0)$ independently distributed.
Suppose no response mechanisms are present in the system, or $\mathrm{AL} = \emptyset$, it is observed from Step 3.3 of Table 4.4 that with $n_{j0}, n_{j1}, \ldots \in \mathrm{NPNL}(n), \mathrm{AL} = \emptyset$,

$$D(n,1) = D(n,0) \vee D\big(n_{j0},0\big) \vee D\big(n_{j1},0\big) \vee \ldots \tag{3}$$

$$\Leftrightarrow D(n,1) = \neg\big(\neg D(n,0) \wedge \neg D\big(n_{j0},0\big) \wedge \neg D\big(n_{j1},0\big) \wedge \ldots\big) \tag{4}$$

Because $D(n,0), D\big(n_{j0},0\big), D\big(n_{j1},0\big), \ldots$ are independent events with probability k, the probability of node n being disrupted at time 1 is

$$P(n,1) = 1 - \prod_{n_j}^{\{n\}\cap \mathrm{NPNL}(n)} \Big(1 - P\big(n_j,0\big)\Big)$$

$$= 1 - \prod_{n_j}^{\{n\}\cap \mathrm{NPNL}(n)} (1-k) = 1 - (1-k)^{|\mathrm{NPNL}(n)+1|} \tag{5}$$

Still with the condition $\mathrm{AL} = \emptyset$, the event of node n being disrupted at time t (illustrated in an example in is

$$D(n,t) = D(n,t-1) \vee D\left(n_{j0},t-1\right) \vee D\left(n_{j1},t-1\right) \vee \ldots \tag{6}$$

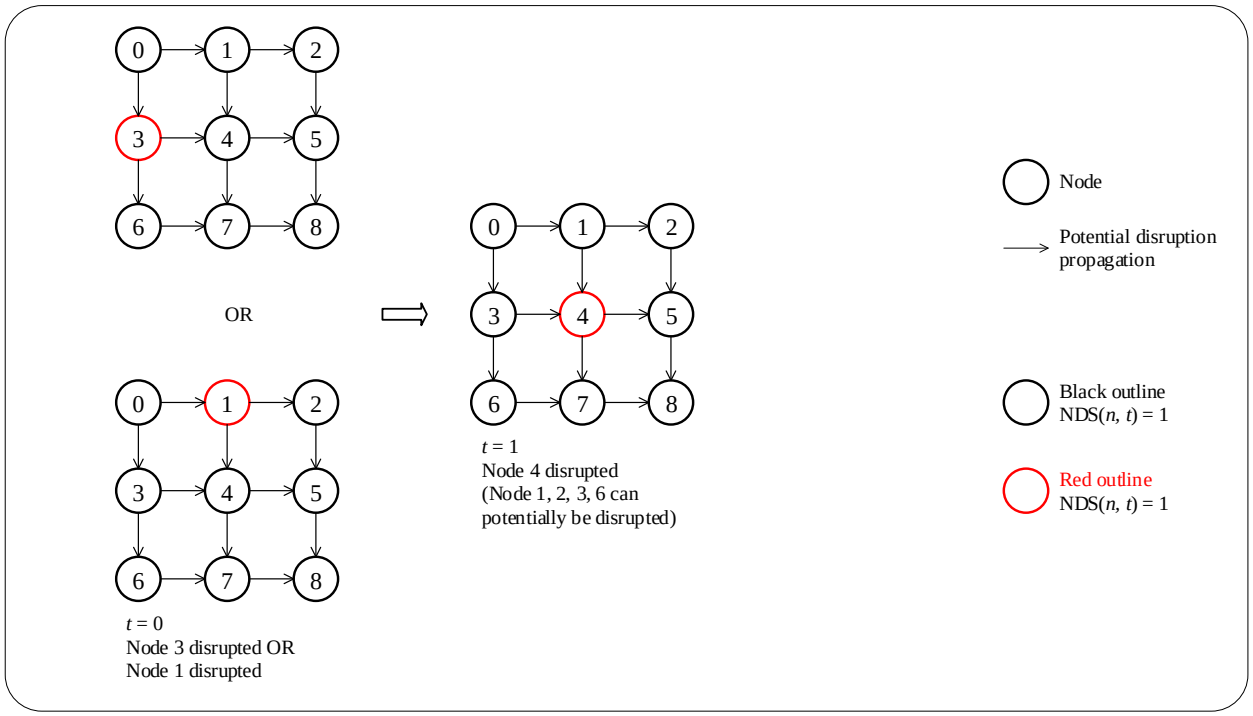

Figure 4.2. Neighboring disruption propagation example with no response

Due to the events $D(n,t-1), D\left(n_{j0},t-1\right), D\left(n_{j1},t-1\right), \ldots$ not guaranteed to be independent, even with the assumption $\mathrm{AL} = \emptyset$, computing $P(n,t)$ with higher values of t would require numerical simulation. With the two above functions, it is observed, with $AL = \emptyset$, that the size of $\mathrm{NPNL}(n)$, or the node in-degree of n, increases the probability of $P(n,t)$ being disrupted.

If $\mathrm{AL} \neq \emptyset$, the event that node n has been observed at time t, or $\mathrm{NOS}(n,t) = 1$, can be defined as $O(n,t)$. Because this type of event is deliberately affected by $\mathrm{ASN}(a,t)$, probability definition for $O(n,t)$ is not possible. However, it is noted from Step 3.2 of Table 4.4 that

$$O(n,t) = O(n,t-1) \vee \exists a \in \mathrm{AL}: \mathrm{ASN}(a,t) = n \tag{7}$$

Then, based on Step 3.3 of Table 4.4, and with $n_{j0}, n_{j1}, \ldots \in \mathrm{NSNL}(n), \mathrm{AL} = \emptyset$, the event $D(n,t)$ can be computed as:

$$D(n,t) = \big(D(n,t-1) \wedge \neg O(n,t)\big) \vee \Big(D\big(n_{j0},t-1\big) \wedge \neg O\big(n_{j0},t\big)\Big) \vee \ldots \quad (8)$$

$$\Leftrightarrow D(n,t) = \neg O(n,t) \wedge \Big(\big(D(n,t-1)\big) \vee \Big(D\big(n_{j0},t-1\big) \wedge \neg O\big(n_{j0},t\big)\Big) \vee \ldots\Big) \quad (9)$$

While $D(n,t)$ is now affected by $O(n,t), O\big(n_{j0},t\big), O\big(n_{j1},t\big)$, ..., in the case $\neg O(n,t)$ is true, the probability of $D(n,t)$ would increase with the node out-degree of n, or the size of $\mathrm{NPNL}(n)$.

4.3.1 CDUD analytics

The aforementioned observation on $D(n,t)$ leads to the first CDUD analytic, which utilizes knowledge of the first CLOC guideline:

$\boldsymbol{\mathcal{A}_1}$: An unobserved node with higher node in-degree has a higher probability of being disrupted. This means

$$\neg O(n,t), |\mathrm{NPNL}(n)| \nearrow \rightarrow \Pr\big(D(n,t)\big) \nearrow \quad (10)$$

Nodes with higher out-degrees also contribute to the probability of its succeeding nodes being disrupted, due to the events $D\big(n_j,t+1\big)$ with $n_j \in \mathrm{NSNL}(n)$ also including an OR clause with $D(n,t)$, thus the second CDUD analytic, which utilizes knowledge of the first CLOC guideline, is defined as:

$\boldsymbol{\mathcal{A}_2}$: An unobserved node with higher node out-degree has a higher probability of propagating disruptions. This means

$$\neg O(n,t), |\mathrm{NSNL}(n)| \nearrow \rightarrow \begin{cases} Pr\left(D\big(n_{j0},t+1\big)\right) \nearrow \\ Pr\left(D\big(n_{j1},t+1\big)\right) \nearrow \\ \ldots \end{cases} , n_{j0}, n_{j1} \ldots \in \mathrm{NSNL}(n): \neg O\big(n_j,t+1\big) \quad (11)$$

Both $\boldsymbol{\mathcal{A}_1}$ and $\boldsymbol{\mathcal{A}_2}$ are important in that an observation $\mathrm{ASN}(n,t)$ can affect the quantities of $\Pr\big(D(n,t)\big)$ and $\sum_{n_j}^{\mathrm{NSNL}(n)} Pr\left(D\big(n_j,t+1\big)\right)$, which affects $\mathrm{PL}(t)$ and $\mathrm{PL}(t+1)$, which in turn affect TPL, MPL, MDP and thus $\boldsymbol{\mathcal{M}}$. Thus, the analytic $\boldsymbol{\mathcal{A}_3}$, which utilizes knowledge of the second CLOC guideline, is defined as

$\boldsymbol{\mathcal{A}_3}$: An observation $O(n,t)$ not only negates the probability of the event $D(n,t)$, but also reduces the probability of the events $D\big(n_j,t+1\big)$ of its unobserved succeeding nodes $n_j \in \mathrm{NSNL}(n)$. This means

$$O(n,t) \rightarrow \Pr\big(D(n,t)\big)=0, \Pr\left(D\big(n_j,t+1\big)\right)\searrow, \forall n_j \in \mathrm{NSNL}(n): \neg O\big(n_j,t+1\big) \qquad (12)$$

So far, the analysis has been limited to the local, neighboring level of a node n. To expand the analysis to the network level, the distance matrix $\mathrm{DIST}: \mathrm{NL} \times \mathrm{NL} \rightarrow \mathbb{Z}_{\geq 0}$ is defined, and can be computed with the information from EL and the Floyd-Warshall algorithm (Floyd, 1962; Dusadeerungsikul & Nof, 2019). Each term of the distance matrix $\mathrm{DIST}\big(n_i,n_j\big) \in \mathbb{Z}_{\geq 0}$ denotes the shortest-path distance between n_i and n_j. Given a shortest path $\mathrm{SP}\big(n_i,n_j\big)=\big(n_i,n_{k0},\dots,n_j\big)$ from node n_i to node n_j with the shortest-path distance of $\mathrm{DIST}\big(n_i,n_j\big)$, if all nodes in the shortest path $\mathrm{SP}\big(n_i,n_j\big)$ are unobserved from t to $t+\mathrm{DIST}\big(n_i,n_j\big)$, the event $D(n_i,t)$ will affect the event $D\left(n_j,t+\mathrm{DIST}\big(n_i,n_j\big)\right)$ as well as the events $D(n_{k0},t+1)$, $D(n_{k1},t+2)$ and so on. Extending the analytic $\boldsymbol{\mathcal{A}_3}$ to the network level results in the analytic $\boldsymbol{\mathcal{A}_4}$, which utilizes knowledge of the second CLOC guideline.

$\boldsymbol{\mathcal{A}_4}$: An observation $O(n,t)$ not only negates the probability of event $D(n,t)$, but also reduces the probability of disruption for nodes connected to it with directed paths (with decreasing with higher distances). This means

$$O(n,t) \rightarrow \begin{cases} \Pr\big(D(n,t)\big) = 0 \\ \Pr\Big(D\big(n_j, t+1\big)\Big) \searrow \searrow \searrow, \forall n_j \in \mathrm{NSNL}(n): \neg O\big(n_j, t+1\big) \\ \Pr\big(D(n_{k0}, t+2)\big) \searrow \searrow, \forall n_{k0}: DIST(n, n_{k0}) = 2, \neg O(n_{k0}, t+2) \\ \Pr\big(D(n_{k1}, t+3)\big) \searrow, \forall n_{k0}: DIST(n, n_{k1}) = 3, \neg O(n_{k1}, t+3) \\ \dots \end{cases} \quad (13)$$

The first part of analytic $\boldsymbol{\mathcal{A}_3}$ dictates that an observation $O(n,t)$ negates the probability of event $D(n,t)$ as well as future events $D(n,t+1)$, completely eliminating the possibility of further disruption propagation originating from node n from the time t onwards. This means, from the perspectives of the disruptions and their propagation, the set of edges EL can remove the edges that has n from time t onwards. If the dynamic set of edges at time t is defined as $\mathrm{DEL}(t) \subset \mathrm{EL}$, and the set of all observations $O(n,t)$ made at time t is defined as OL(t). This means the set of observations $\mathrm{OL}(t)$ consisting of $O(n,t)$ would remove all edges $e = \big(n, n_j\big)$ connected to the nodes n from $\mathrm{DEL}(t-1)$. This means

$$\mathrm{OL}(t) \Rightarrow \mathrm{DEL}(t) = \mathrm{DEL}(t-1) - \big\{e = \big(n, n_j\big) \in \mathrm{DEL}(t-1)\big\}, \quad \forall n_j \in \mathrm{NL}, \forall n \in \mathrm{NL}: O(n,t) \in \mathrm{OL} \quad (14)$$

The fifth analytic $\boldsymbol{\mathcal{A}_5}$, which utilizes knowledge of the third CLOC guideline, can then be defined as $\boldsymbol{\mathcal{A}_5}$: New $\mathrm{OL}(t)$ decisions should consider previous $\mathrm{OL}(t-1)$ in order to maximize both the detection of ongoing disruptions and the prevention of potential future disruption propagation. This analytic can be further expanded to become the analytic $\boldsymbol{\mathcal{A}_6}$, which utilizes knowledge of the third CLOC guideline.

$\boldsymbol{\mathcal{A}_6}$: Within t, new decisions $O(n,t) \in \mathrm{OL}(t)$ should consider both previous decisions $\mathrm{OL}(t-1)$ and themselves, the concurrent decisions $O(n,t)$ made at t. This means within t, each new observation decision $O(n_{i+1}, t)$ should consider all previously made decisions of the same timestep t. Using $\boldsymbol{\mathcal{A}_6}$ to support decision-making has the potential to improve disruption detection and disruption propagation prevention further than $\boldsymbol{\mathcal{A}_5}$.

The six developed CDUD analytics and their corresponding CLOC guidelines are summarized in Table 4.5.

Table 4.5. Summary of the CDUD analytics

Analytic	Description	CLOC guideline
$\boldsymbol{\mathcal{A}_1}$	An unobserved node with a higher in-degree is more likely to be disrupted.	CLOC 1
$\boldsymbol{\mathcal{A}_2}$	An unobserved node with a higher out-degree is more likely to cause more severe disruption propagation.	CLOC 1
$\boldsymbol{\mathcal{A}_3}$	An observation to a node both removes the disruption (if any) and prevents future disruption propagation coming from this node.	CLOC 2
$\boldsymbol{\mathcal{A}_4}$	An observation to a node both removes the disruption (if any) and helps prevent future disruption propagation to nodes connected to it with directed paths.	CLOC 2
$\boldsymbol{\mathcal{A}_5}$	New response decisions should consider past response decisions.	CLOC 3
$\boldsymbol{\mathcal{A}_6}$	New response decisions should consider past response decisions as well as concurrent response decisions.	CLOC 3

4.3.2 CDUD protocols

Based on the six aforementioned analytics, 10 protocols $\boldsymbol{\mathcal{P}}$ are established, categorized by three levels of sophistication: basic, intermediate, and advanced. Basic protocols are not supported by $\boldsymbol{\mathcal{A}_5}$ nor $\boldsymbol{\mathcal{A}_6}$, whereas intermediate protocols are supported by $\boldsymbol{\mathcal{A}_5}$ but not $\boldsymbol{\mathcal{A}_6}$, and advanced protocols are supported by $\boldsymbol{\mathcal{A}_6}$. The higher levels of sophistication are accompanied by higher computational resources requirements.

$\boldsymbol{\mathcal{P}_1}$: random allocation protocol, in which $\mathrm{ASN}(n,t)$ is selected randomly from $\{n \in \mathrm{NL}: \mathrm{NOS}(n,t-1)=0\}$. This protocol is a baseline protocol and is specified for the purpose of comparison. This protocol is not supported by analytics.

$\boldsymbol{\mathcal{P}}_2$: basic degree centrality allocation protocol, in which $\text{ASN}(n,t)$ is selected from $\{n \in \text{NL}: \text{NOS}(n,t-1)=0\}$ sorted by $(|\text{NPNL}(n)|+|\text{NSNL}(n)|)$ in descending order. This protocol is supported by $\boldsymbol{\mathcal{A}}_1$, $\boldsymbol{\mathcal{A}}_2$, and $\boldsymbol{\mathcal{A}}_3$, and it prioritizes nodes with higher node-degree.

$\boldsymbol{\mathcal{P}}_3$: basic harmonic centrality allocation protocol, in which $\text{ASN}(n,t)$ is selected from $\{n \in \text{NL}: \text{NOS}(n,t-1)=0\}$ sorted by the node's harmonic centrality $\text{NHC}(n) \in \mathbb{R}_{>0}$ in descending order. The harmonic centrality measure is defined as the harmonic mean of all distances between all pairs of different nodes of the network (Warshall, 1962). This protocol is supported by $\boldsymbol{\mathcal{A}}_1$, $\boldsymbol{\mathcal{A}}_2$, $\boldsymbol{\mathcal{A}}_3$, and $\boldsymbol{\mathcal{A}}_4$, and it prioritizes nodes with higher harmonic centrality values.

$$\text{NHC}(n) = \sum_{n_j}^{\text{NL}-\{n\}} \frac{1}{\text{DIST}(n,n_j)} \tag{15}$$

$\boldsymbol{\mathcal{P}}_4$: basic expanded centrality allocation protocol, in which $\text{ASN}(n,t)$ is selected from $\{n \in \text{NL}: \text{NOS}(n,t-1)=0\}$ sorted by $(|\text{NPNL}(n)|+|\text{NSNL}(n)|)$ in descending order, tie-breaking by NHC(n). This protocol is supported by $\boldsymbol{\mathcal{A}}_1$, $\boldsymbol{\mathcal{A}}_2$, $\boldsymbol{\mathcal{A}}_3$, and $\boldsymbol{\mathcal{A}}_4$, and it provides a compromise between local-level importance and network-level importance.

$\boldsymbol{\mathcal{P}}_5$: intermediate degree centrality allocation protocol, in which $\text{ASN}(n,t)$ is selected from $\{n \in \text{NL}: \text{NOS}(n,t-1)=0\}$ sorted by $\left|\{n_j \in \text{NDNL}(n) \cap \text{NSNL}(n): \text{NOS}(n_j,t-1)=0\}\right|$ in descending order. This protocol considers the past allocation decisions and is supported by $\boldsymbol{\mathcal{A}}_1$, $\boldsymbol{\mathcal{A}}_2$, $\boldsymbol{\mathcal{A}}_3$, and $\boldsymbol{\mathcal{A}}_5$. This protocol prioritizes nodes with higher numbers of unobserved neighboring (preceding + succeeding) nodes.

$\boldsymbol{\mathcal{P}}_6$: intermediate harmonic centrality allocation protocol, in which $\text{ASN}(n,t)$ is selected from $\{n \in \text{NL}: \text{NOS}(n,t-1)=0\}$ sorted by the node's intermediate harmonic centrality $\text{NIHC}(n) \in \mathbb{R}_{\geq 0}$ in descending order. This protocol builds upon $\boldsymbol{\mathcal{P}}_3$ by considering the past allocation decisions and is supported by $\boldsymbol{\mathcal{A}}_1$, $\boldsymbol{\mathcal{A}}_2$, $\boldsymbol{\mathcal{A}}_3$, $\boldsymbol{\mathcal{A}}_4$, and $\boldsymbol{\mathcal{A}}_5$.

$$\mathrm{NIHC}(n) = \sum_{n_j}^{\{n_j \in \mathrm{NL}-\{n\}:\mathrm{NOS}(n_j,t-1)=0\}} \frac{1}{\mathrm{DIST}(n,n_j)} \tag{16}$$

$\boldsymbol{\mathcal{P}_7}$: intermediate expanded centrality allocation protocol, in which $\mathrm{ASN}(n,t)$ is selected from $\{n \in \mathrm{NL}: \mathrm{NOS}(n,t-1)=0\}$ sorted by $\left|\{n_j \in \mathrm{NDNL}(n) \cap \mathrm{NSNL}(n): \mathrm{NOS}(n_j,t-1)=0\}\right|$, tie-breaking by $\mathrm{NIHC}(n)$ in descending order. This protocol considers the past allocation decisions and is supported by $\boldsymbol{\mathcal{A}_1}$, $\boldsymbol{\mathcal{A}_2}$, $\boldsymbol{\mathcal{A}_3}$, $\boldsymbol{\mathcal{A}_4}$, and $\boldsymbol{\mathcal{A}_5}$.

$\boldsymbol{\mathcal{P}_8}$: advanced degree centrality allocation protocol, in which $\mathrm{ASN}(n,t)$ is selected from $\{n \in \mathrm{NL}: \mathrm{NOS}(n,t-1)=0\}$ sorted by $\left|\{n_j \in \mathrm{NDNL}(n) \cap \mathrm{NSNL}(n): \mathrm{NOS}(n_j,t-1)=0 \vee \mathrm{NOS}(n_j,t)=0\}\right|$ in descending order. This protocol considers the past allocation decisions and concurrent allocation decisions of a node's neighboring (preceding + succeeding) nodes. This protocol is supported by all analytics except $\boldsymbol{\mathcal{A}_4}$.

$\boldsymbol{\mathcal{P}_9}$: advanced harmonic centrality allocation protocol, in which $\mathrm{ASN}(n,t)$ is selected from $\{n \in \mathrm{NL}: \mathrm{NOS}(n,t-1)=0\}$ sorted by the node's advanced harmonic centrality $\mathrm{NIHC}(n) \in \mathbb{R}_{\geq 0}$ in descending order. This protocol builds upon $\boldsymbol{\mathcal{P}_3}$ and $\boldsymbol{\mathcal{P}_6}$ by considering the past allocation decisions and concurrent allocation decisions of all other nodes in its calculation. This protocol is supported by all analytics $\boldsymbol{\mathcal{A}}$.

$$\mathrm{NAHC}(n) = \sum_{n_j}^{\{n_j \in \mathrm{NL}-\{n\}:\mathrm{NOS}(n_j,t-1)=0 \wedge \mathrm{NOS}(n_j,t)=0\}} \frac{1}{\mathrm{DIST}(n,n_j)} \tag{17}$$

$\boldsymbol{\mathcal{P}_{10}}$: advanced expanded centrality allocation protocol, in which $\mathrm{ASN}(n,t)$ is selected from $\{n \in \mathrm{NL}: \mathrm{NOS}(n,t-1)=0\}$ sorted by $\left|\{n_j \in \mathrm{NDNL}(n) \cap \mathrm{NSNL}(n): \mathrm{NOS}(n_j,t-1)=0 \vee \mathrm{NOS}(n_j,t)=0\}\right|$ in descending order, tie-breaking by $\mathrm{NAHC}(n)$. This protocol considers the past allocation decisions and concurrent allocation decisions of all other nodes in its calculation. This protocol is supported by all analytics $\boldsymbol{\mathcal{A}}$.

The ten CDUD protocols are summarized in Table 4.6.

Table 4.6. Summary of the CDUD protocols

Analytic	Description	Collaboration level	Related analytics
$\boldsymbol{\mathcal{P}}_1$	Random allocation protocol: baseline, random.	None	None
$\boldsymbol{\mathcal{P}}_2$	Basic degree centrality allocation protocol: prioritizes higher node degree.	Low	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3$
$\boldsymbol{\mathcal{P}}_3$	Basic harmonic centrality allocation protocol: prioritizes higher harmonic centrality.	Low	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4$
$\boldsymbol{\mathcal{P}}_4$	Basic expanded centrality allocation protocol: prioritizes higher node degree, tie-breaking by harmonic centrality.	Low	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4$
$\boldsymbol{\mathcal{P}}_5$	Intermediate degree centrality allocation protocol: prioritizes higher node degree considering past decisions.	Medium	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_5$
$\boldsymbol{\mathcal{P}}_6$	Intermediate harmonic centrality allocation protocol: prioritizes harmonic centrality considering past decisions.	Medium	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4, \boldsymbol{\mathcal{A}}_5$
$\boldsymbol{\mathcal{P}}_7$	Intermediate expanded centrality allocation protocol: prioritizes higher node degree, tie-breaking by harmonic centrality, considering past decisions.	Medium	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4, \boldsymbol{\mathcal{A}}_5$
$\boldsymbol{\mathcal{P}}_8$	Advanced degree centrality allocation protocol: prioritizes higher node degree, considering past and concurrent decisions.	Very high	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4, \boldsymbol{\mathcal{A}}_6$
$\boldsymbol{\mathcal{P}}_9$	Advanced harmonic centrality allocation protocol: prioritizes higher harmonic centrality, considering past and concurrent decisions.	Very high	All $\boldsymbol{\mathcal{A}}$
$\boldsymbol{\mathcal{P}}_{10}$	Advanced expanded centrality allocation protocol: prioritizes higher node degree, tie-breaking by harmonic centrality, considering past and concurrent decisions.	Very high	All $\boldsymbol{\mathcal{A}}$

4.4 Numerical Experiments and Results

Numerical experiments are conducted to validate the CDUD model, analytics, and protocols. The factors of the experiments include: five network types, six response/disruption scenarios, and ten protocols (from $\boldsymbol{\mathcal{P}}_1$ to $\boldsymbol{\mathcal{P}}_{10}$) with 1000 replications for each factor combination, resulting in 300,000 runs in total. The high number of replications is selected to ensure that the experiments consider a sufficiently high number of different disruption target combinations, while ensuring reasonable total runtime of the experiments. The three system performance metrics $\boldsymbol{\mathcal{M}}_1$, $\boldsymbol{\mathcal{M}}_2$, and $\boldsymbol{\mathcal{M}}_3$ (all minimization objectives) are reported.

The five network types are:

1. GO: 10x10 grid orthogonal both-way propagation;
2. GD: 10x10 grid orthogonal and diagonal both-way propagation;
3. BA: 100-node random Barabasi-Albert with $m_0 = 2, m = 2$ with bidirectional edges;
4. ER: 100-node random Erdos-Renyi with $p = 0.08$ with bidirectional edges;
5. WS: 100-node random Watts-Strogatz with $k = 4, \beta = 0.5$ with bidirectional edges.

The network types GO and GD are selected due to their applicability to the agricultural settings, particularly greenhouses (Marchiori & Latora, 2000; Dusadeerungsikul & Nof, 2019). The BA (Dusadeerungsikul et al., 2018), ER (Barabasi & Albert, 1999), and WS (Erdös & Rényi, 1959) network types are selected because these random network models are common choices for complex networks research and cyber-physical systems research (Watts, 2002; Arora & Ventresca, 2017).

The six response/disruption scenarios are:

1. R20D10: 20 response agents, 10% initial disruption probability.
2. R30D10: 30 response agents, 10% initial disruption probability.
3. R40D10: 40 response agents, 10% initial disruption probability.
4. R20D15: 20 response agents, 15% initial disruption probability.
5. R30D15: 30 response agents, 15% initial disruption probability.

6. R40D15: 40 response agents, 15% initial disruption probability.

Additionally, a separate set of experiments is conducted on an enterprise's internal email network, using the ten aforementioned CDUD protocols and six response/disruption scenarios, with 1000 replications for each factorial combination (subsection 4.4.4).

4.4.1 Comparison by CDUD protocols

The comparison between CDUD protocols is provided in Figure 4.3 and Table 4.7.

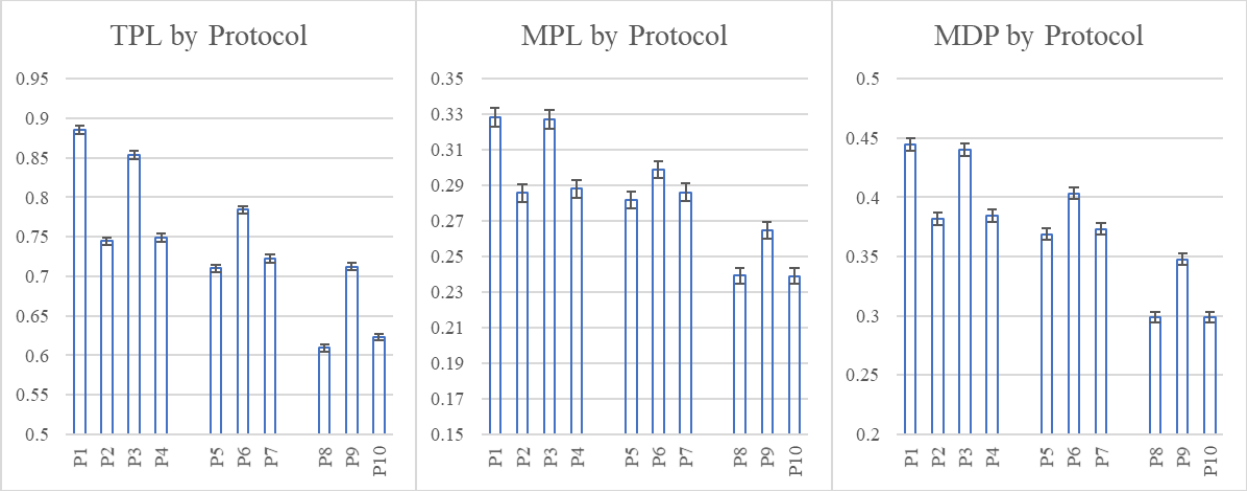

Figure 4.3. CDUD experiment results grouped by CDUD protocols with 95% confidence interval bars

Table 4.7. CDUD experiment results grouped by CDUD protocols

Protocol	TPL	MPL	MDP
$\boldsymbol{\mathcal{P}}_1$	0.8855	0.3284*	0.4448*
$\boldsymbol{\mathcal{P}}_2$	0.7444*	0.2858**	0.3818**
$\boldsymbol{\mathcal{P}}_3$	0.8534	0.3272*	0.4402*
$\boldsymbol{\mathcal{P}}_4$	0.7486*	0.2881**	0.3847**
$\boldsymbol{\mathcal{P}}_5$	0.7102**	0.2817**	0.3688**
$\boldsymbol{\mathcal{P}}_6$	0.7841	0.2989**	0.4032
$\boldsymbol{\mathcal{P}}_7$	0.7225**	0.2861**	0.3734**
$\boldsymbol{\mathcal{P}}_8$	0.6092	0.2392***	0.2988***
$\boldsymbol{\mathcal{P}}_9$	0.7121**	0.2648	0.3476
$\boldsymbol{\mathcal{P}}_{10}$	0.6230	0.2390***	0.2989***
Gap of $\boldsymbol{\mathcal{P}}_8, \boldsymbol{\mathcal{P}}_{10}$ versus other protocols	12.5% - 29.6%	9.7% - 27.2%	14% - 32.8%

*, **, ***, ****: group of confidence interval overlapping
Same group means no significant statistical difference between the different CDUD protocols of the same group.
Different groups mean significant statistical differences between any pair of CDUD protocols belonging to different groups.

The experiment results (Figure 4.3 and Table 4.7) are the system performance metrics TPL, MPL, and MDP averaged across all network types and response/disruption scenarios. With respect to overall performance, the advanced CDUD protocols $\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$ outperform all other protocols with statistical significance, ranging from 9.7% to 32.8%. This superiority applies to all three system performance metrics, with 12.5% to 29.6% for total performance loss TPL, 9.7% to 27.2% for maximum performance loss MPL, and 14% to 32.8% for maximum disruption propagation MDP. The $\boldsymbol{\mathcal{P}}_5$, $\boldsymbol{\mathcal{P}}_7$, and $\boldsymbol{\mathcal{P}}_9$ belongs to the group of second-best overall performances, followed by $\boldsymbol{\mathcal{P}}_2$ and $\boldsymbol{\mathcal{P}}_4$, then by $\boldsymbol{\mathcal{P}}_6$, then by $\boldsymbol{\mathcal{P}}_1$ and $\boldsymbol{\mathcal{P}}_3$. It is notable that the harmonic centrality protocols $\boldsymbol{\mathcal{P}}_3, \boldsymbol{\mathcal{P}}_6, \boldsymbol{\mathcal{P}}_9$ have worse performance compared to the degree centrality protocols. One possible explanation is that the local-level restrain effects are more significant than the global-level restrain effects. The CDUD protocols with higher levels of collaboration ($\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$) also outperform the other levels of collaboration. This emphasizes the role of collaborative control in allocating response decisions. These results (and Table 4.7) indicate that, in general, the advanced

collaborative CDUD protocols ($\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$) outperform the less advanced and less collaborative protocols in detecting and removing disruptions.

4.4.2 Comparison by CDUD protocols and network types

The comparisons between CDUD protocols, grouped by network types, are provided in Figure 4.4.

Figure 4.4. CDUD experiment results grouped by CDUD protocols and network types, with 95% confidence interval bars

It is noted that advanced CDUD protocols $\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$ outperform all other protocols with the network types GO, GD, ER, and WS. These two protocols $\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$ still have good

performances with the network type BA, but they are tied with $\boldsymbol{\mathcal{P}}_2$, $\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_7$. Another observation is the notable differences in performances with different network types, despite all of them having the same number of nodes. This observation is summarized in Table 4.8.

Table 4.8. CDUD experiment results grouped by network types

Network Type	**Number of nodes**	**Number of edges**	**Rank**	**Performance metrics range**	**Performance gaps between $\boldsymbol{\mathcal{P}}_8, \boldsymbol{\mathcal{P}}_{10}$ versus other protocols**
GO	100	180	2	TPL 0.54 - 0.92 MPL 0.21 - 0.34 MDP 0.27 - 0.49	TPL 16.5% - 41% MPL 7.8% - 37.8% MDP 13.3% - 45.1%
GD		342	4	TPL 0.83 - 1.04 MPL 0.32 - 0.41 MDP 0.42 - 0.54	TPL 11.8% - 20.2% MPL 10.4% - 21.2% MDP 13.3% - 45.1%
BA		200	1 (best)	TPL 0.33 - 0.68 MPL 0.13 - 0.24 MDP 0.15 - 0.34	TPL 2.4% - 51% MPL 0.9% - 44.1% MDP 2.1% - 56.5%
ER		800	5 (worst)	TPL 0.80 - 1.09 MPL 0.31 - 0.42 MDP 0.40 - 0.56	TPL 4.4% - 26.7% MPL 5% - 25.4% MDP 8.5% - 28.4%
WS		200	3	TPL 0.54 - 0.86 MPL 0.21 - 0.31 MDP 0.26 - 0.44	TPL 11.4% - 37.3% MPL 15.7% - 33.5% MDP 17.6% - 40.5%

It is noted that the network types GO, BA, and WS have roughly the same number of edges, and the system performance differences are still significantly different. The network types GD and ER rank 4 and 5, respectively, in terms of performance metrics, and one likely reason is the higher number of edges. The advanced collaborative protocols ($\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$) outperform the less advanced and less collaborative protocols by 7.8% to 45.1% for the GO network type, 10.4% to 45.1% for GD, 0.9% to 56.5% for BA, 4.4% to 28.4% for ER, and 11.4% to 40.5% for WS It is also noted that the performance gaps are more significant in the case of BA, and least significant in the case of GD. These results (Table 4.8) indicate that the advanced collaborative CDUD protocols ($\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$) significantly outperform the less advanced and less collaborative protocols in detecting and removing disruptions, with four (GO, GD, ER, and WS) out of the five different network types investigated.

4.4.3 Comparison by CDUD protocols and response/disruption scenarios

The comparisons between CDUD protocols, grouped by response/disruption scenarios, are provided in Figure 4.5 and Table 4.9.

Figure 4.5. CDUD experiment results grouped by CDUD protocols and response/disruption scenarios, with 95% confidence interval bars

Table 4.9. CDUD experiment results grouped by response/disruption scenarios

Scenario	Number of Response Agents	Initial Disruption Probability	Performance metrics range	Performance gaps between $\mathcal{P}_8, \mathcal{P}_{10}$ versus other protocols
R20D10	20	10%	TPL 0.87 - 1.31 MPL 0.28 - 0.41 MDP 0.37 - 0.56	TPL 12.4% - 33.3% MPL 9.5% - 31.5% MDP 13.7% - 33.8%
R30D10	30	10%	TPL 0.41 - 0.66 MPL 0.19 - 0.27 MDP 0.23 - 0.38	TPL 18% - 37.4% MPL 9.8% - 28.9% MDP 17.7% - 40%
R40D10	40	10%	TPL 0.26 - 0.40 MPL 0.14 - 0.21 MDP 0.17 - 0.27	TPL 15.7% - 34.6% MPL 14% - 35% MDP 15.6% - 39.2%
R20D15	20	15%	TPL 1.15 - 1.57 MPL 0.36 - 0.48 MDP 0.47 - 0.64	TPL 9.8% - 26.4% MPL 7.1% - 23.2% MDP 10.8% - 25.9%
R30D15	30	15%	TPL 0.58 - 0.85 MPL 0.27 - 0.35 MDP 0.32 - 0.48	TPL 15.6% - 31.3% MPL 8.7% - 24.4% MDP 15.2% - 33.5%
R40D15	40	15%	TPL 0.38 - 0.55 MPL 0.20 - 0.29 MDP 0.24 - 0.37	TPL 14.5% - 30.7% MPL 12.6% - 31.4% MDP 14.3% - 34.9%

The performance gaps between the advanced CDUD protocols ($\mathcal{P}_8$ and $\mathcal{P}_{10}$) and other protocols follow the same pattern as in the gaps provided in Table 4.7. Across all the different response/disruption scenarios, the performance gaps between the advanced CDUD protocols ($\mathcal{P}_8$ and $\mathcal{P}_{10}$) and the less advanced protocols are 9.8% to 37.4% for TPL, 7.1% to 35% for MPL, and 10.8% to 40% for MDP. These results (Table 4.9) indicate that the advanced collaborative CDUD protocols ($\mathcal{P}_8$ and $\mathcal{P}_{10}$) outperform the less advanced and less collaborative protocols in detecting and removing disruptions, with all the different scenarios of the number of response agents and disruptions involved. With 10 disruptions, the performance gap increases from 9.5%-33.8% with 20 response agents to 14%-39.2% with 40 response agents. With 15 disruptions, the performance gap increases from 7.1%-26.4% with 20 response agents to 12.6%-34.9% with 40 response agents. This means the performance gap between the advanced CDUD protocols ($\mathcal{P}_8$ and $\mathcal{P}_{10}$) and the less advanced protocols increases with higher numbers of response agents used, indicating that the advanced CDUD protocols are more efficient in utilizing response resources.

4.4.4 Set of experiments on an enterprise's internal email network

The previous sets of experiments apply the CDUD model to five different types of numerically generated network models. In this set of experiments, the CDUD model, analytics, and protocols are applied to the problem of detecting ($\boldsymbol{\mathcal{R}}$) hidden propagating computer malware ($\boldsymbol{\mathcal{D}}$) in an actual enterprise's internal email network ($\boldsymbol{\mathcal{C}}$). The purpose of this set of experiments is to test the CDUD model, analytics, and protocols in an actual network model and problem. The internal email network of an enterprise can be vulnerable to propagating computer malware (the disruptions), because of the higher level of trusts and frequency of communication between the participants (the nodes) (Hao Zhong & Nof, 2015). File-sharing is common amongst the participants of the internal email network, and a malware originating from one participant can propagate to other participants through the file-sharing activities. Therefore, the established communication between two participants (nodes n_i, n_j) of an email network ($\boldsymbol{\mathcal{C}}$) constitutes a potential disruption propagation direction (edge $e = (n_i, n_j)$). In the context of the CDUD model, the disruptions ($\boldsymbol{\mathcal{D}}$) in an internal email network are the undetected computer malware, such as computer viruses, trojan horses, and/or computer worms, that can propagate between participants of the email network. The detection agents ($\boldsymbol{\mathcal{R}}$), in this case, can scan the participants' emails to find the malware, but the malware is not known to the email network nor the detection hands until detected.

In this set of experiments, the selected enterprise's internal email network structure is condensed from the email communication of the Enron scandal, specifically from the communication between the Enron high-level employees (Cohen, 2005; Musa et al., 2018). This enterprise's internal email network contains a total of 143 nodes (participants) and 623 undirected edges (an edge is created if any email communication was made). The node count of 143 of this email network allows reasonable comparison with the previous sets of experiments on five general random network models (GO, GD, BA, ER, and WS). To match the specification of the CDUD model, all undirected edges are converted to bi-directional edges. In this set of experiments, the ten aforementioned CDUD protocols and six aforementioned response/disruption scenarios (discussed in 4.4) are the experiment factors, with 1000 replications for each factorial combination, resulting in a total of 60000 runs. The high number of replications is selected to ensure that the experiments consider a sufficiently high number of different disruption target combinations while ensuring

reasonable total runtime of the experiments. The three system performance metrics $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ (all minimization objectives) are reported.

The comparison between CDUD protocols is presented in Figure 4.6 and Table 4.10.

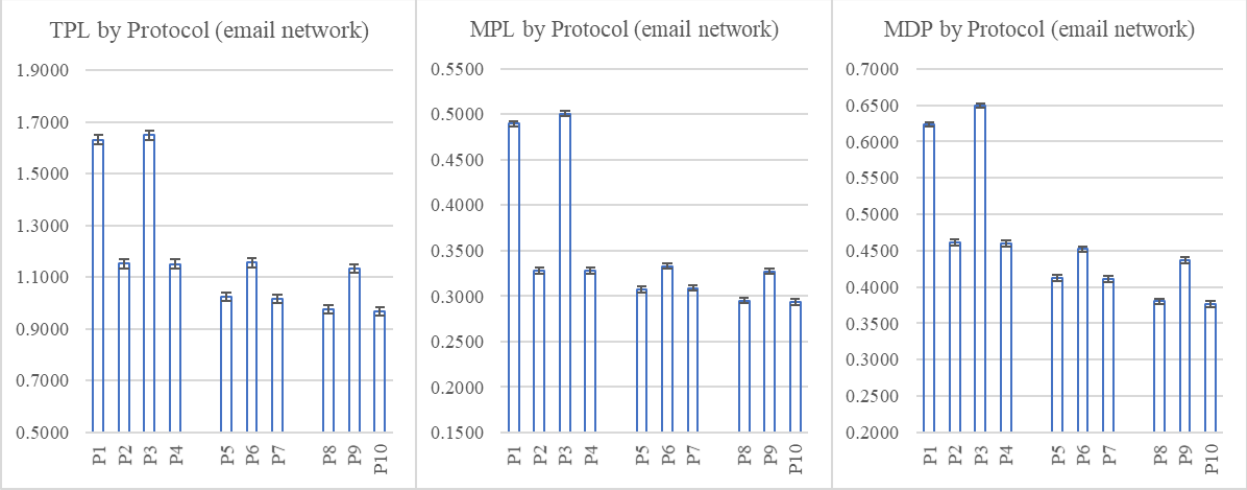

Figure 4.6. CDUD email network experiment results grouped by CDUD protocols with 95% confidence interval bars

Table 4.10. CDUD email network experiment results grouped by CDUD protocols

Protocol	TPL	MPL	MDP
$\boldsymbol{\mathcal{P}}_1$	1.6315*	0.4895*	0.6234*
$\boldsymbol{\mathcal{P}}_2$	1.1521**	0.3283**	0.4610**
$\boldsymbol{\mathcal{P}}_3$	1.6492*	0.5009*	0.6494*
$\boldsymbol{\mathcal{P}}_4$	1.1513**	0.3283**	0.4599**
$\boldsymbol{\mathcal{P}}_5$	1.0235***	0.3073***	0.4119***
$\boldsymbol{\mathcal{P}}_6$	1.1556**	0.3326**	0.4521**
$\boldsymbol{\mathcal{P}}_7$	1.0163***	0.3088***	0.4111***
$\boldsymbol{\mathcal{P}}_8$	0.9766****	0.2950****	0.3802****
$\boldsymbol{\mathcal{P}}_9$	1.1340**	0.3274**	0.4369
$\boldsymbol{\mathcal{P}}_{10}$	0.9669****	0.2936****	0.3764****
Gap of $\boldsymbol{\mathcal{P}}_8, \boldsymbol{\mathcal{P}}_{10}$ versus other protocols	4.9%-40.7%	4.5%-39.7%	8.4%-39.6%

*, **, ***, ****: group of confidence interval overlapping
Same group means no significant statistical difference between the different CDUD protocols of the same group.
Different groups mean significant statistical differences between any pair of CDUD protocols belonging to different groups.

The email network experiment results (Figure 4.6 and Table 4.10) are the system performance metrics TPL, MPL, and MDP averaged across all network types and response/disruption scenarios. Overall, the advanced CDUD protocols $\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$ outperform all other CDUD protocols with statistical significance, ranging from 4.5% to 40.7%. This superiority applies to all three system performance metrics, with 4.9% to 40.7% for total performance loss TPL, 4.5% to 39.7% for maximum performance loss MPL, and 8.4% to 39.6% for maximum disruption propagation MDP. These performance gaps (Table 4.10) are similar to those presented in the experiments on the ER network model type (Table 4.8). The protocols $\boldsymbol{\mathcal{P}}_5$, $\boldsymbol{\mathcal{P}}_7$, and belongs to the group of second-best overall performances, followed by $\boldsymbol{\mathcal{P}}_2$, $\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_6$ and $\boldsymbol{\mathcal{P}}_9$, then by $\boldsymbol{\mathcal{P}}_1$ and $\boldsymbol{\mathcal{P}}_3$. These results (Table 4.10) indicate that, in general, the advanced collaborative CDUD protocols ($\boldsymbol{\mathcal{P}}_8$ and $\boldsymbol{\mathcal{P}}_{10}$) outperform the less advanced and less collaborative protocols in detecting and removing disruptions.

The comparisons between CDUD protocols, grouped by response/disruption scenarios, are provided in Figure 4.7 and Table 4.11.

Figure 4.7. CDUD email network experiment results grouped by CDUD protocols with 95% confidence interval bars

Table 4.11. CDUD email network experiment results grouped by response/disruption scenarios

Scenario	Number of Response Agents	Initial Disruption Probability	Performance metrics range	Performance gaps between $\boldsymbol{\mathcal{P}_8}, \boldsymbol{\mathcal{P}_{10}}$ versus other protocols
R20D10	20	10%	TPL 1.53 - 2.51 MPL 0.38 - 0.62 MDP 0.51 - 0.77	TPL 3.3%-37.2% MPL 1.6%-37.9% MDP 4.8%-31.7%
R30D10	30	10%	TPL 0.61 - 1.35 MPL 0.22 - 0.47 MDP 0.28 - 0.61	TPL 4.2%-54.8% MPL 3.6%-53.8% MDP 8.7%-54.1%
R40D10	40	10%	TPL 0.34 - 0.80 MPL 0.16 - 0.34 MDP 0.18 - 0.48	TPL 7.1%-57.5% MPL 9.4%-52.9% MDP 12.6%-61.2%
R20D15	20	15%	TPL 1.95 - 2.72 MPL 0.47 - 0.66 MDP 0.63 - 0.80	TPL 3.1%-26.7% MPL 1.6%-27.7% MDP 4.1%-19.7%
R30D15	30	15%	TPL 0.85 - 1.55 MPL 0.30 - 0.52 MDP 0.39 - 0.68	TPL 3.8%-44.9% MPL 2.5%-41.5% MDP 7.8%-42.8%
R40D15	40	15%	TPL 0.49 - 0.99 MPL 0.23 - 0.40 MDP 0.26 - 0.57	TPL 6.5%-50.4% MPL 8.4%-42.9% MDP 11.6%-53.3%

Across all the different response/disruption scenarios, the performance gaps (Table 4.11) between the advanced CDUD protocols ($\boldsymbol{\mathcal{P}_8}$ and $\boldsymbol{\mathcal{P}_{10}}$) and the less advanced protocols are 3.1% to 57.5% for TPL, 1.6% to 53.8% for MPL, and 4.1% to 61.2% for MDP. These results (Table 4.11) indicate that the advanced collaborative CDUD protocols ($\boldsymbol{\mathcal{P}_8}$ and $\boldsymbol{\mathcal{P}_{10}}$) outperform the less advanced and less collaborative protocols in detecting and removing disruptions, in all the different scenarios of the number of response agents and disruptions involved. With 10 disruptions, the performance gap increases from 1.6%-37.9% with 20 response agents to 7.1%-61.2% with 40 response agents. With 15 disruptions, the performance gap increases from 1.6%-27.7% with 20 response agents to 6.5%-53.3% with 40 response agents. This means the performance gap between the advanced CDUD protocols ($\boldsymbol{\mathcal{P}_8}$ and $\boldsymbol{\mathcal{P}_{10}}$) and the less advanced protocols increases with higher numbers of response agents used, indicating that the advanced CDUD protocols are more efficient in utilizing response resources. To summarize, the experiment results (Figure 4.6, Table 4.10, Figure 4.7, and Table

4.11) indicate that the CDUD advanced protocols provide superior detection performance against undetected computer malware in the enterprise's internal email network.

4.5 Concluding Remarks

In this chapter, the CDUD model is defined and formulated based on the CRDP framework, and the accompanying CDUD analytics and protocols are developed based on the guidelines of the CLOC principle. The CDUD model explores one important type of disruption: the property of being unknown to the response mechanisms. In the CDUD model, active and dynamic response agents are deployed to detect and remove the disruption, and also protect the client systems from future disruption propagation. Six analytics are developed based on the CLOC principle, and ten protocols are developed with the support of the six analytics. The CDUD model and protocols are validated with numerical experiments. The experiment results indicate that the advanced protocols developed based on the CLOC principle outperform the baseline and less advanced protocols by 9.7% to 32.8%, with statistical significance. The results indicate that the appropriate application of the CRDP formulation and the CLOC design and control principles can lead to significant performance improvement.

Thus, this case in CHAPTER 4 provides a partial answer to both Research Question 2 and Research Question 3 as outlined in CHAPTER 1.

CHAPTER 5. CASE 2 – COLLABORATIVE STRATEGIC PREVENTION OF DISRUPTION PROPAGATION

5.1 CSPD Description

Another important and common property of disruption is the unexpectedness of disruption occurrence. Unexpected disruptions can occur in any location of the client systems, albeit being more likely in some locations more than others. Unexpected disruptions also create difficulties for dynamic response activities. This is because the dynamic response mechanisms must either be on constant standby, incurring high expenditures, or be delayed in response, incurring heavy losses due to disruption propagation. The unexpected disruptions can be somewhat mitigated with the deployment of strategic and static response mechanisms, which is explored in this case study. This problem is relevant to the agriculture plant disease settings and the fire spreading problem in buildings and forests. In the agricultural setting, static mechanisms include immunizations against diseases and statically deployed sensors can detect diseases and mitigate the propagation. Similarly, in the fire spreading case, fire sprinklers and local water resources can help mitigating and containing the fire until the active response mechanisms arrive. The two aforementioned settings inspire the formulation of an RDP model to address the aspect "strategic allocation" of the response mechanisms and the aspect "unexpected" of the disruptions.

Following the CRDP framework, the Collaborative Strategic Prevention of Disruption Propagation (CSPD) model is formulated with the components, interactions, decision space, and system performance metrics. The entities of the client system $\boldsymbol{\mathcal{C}}$ are represented by nodes, each of which can represent a plant, a group of plants, a building, or a building section. The nodes are susceptible to disruptions in $\boldsymbol{\mathcal{D}}$ that can propagate to nearby or connected nodes. The disruption information of a node is not available to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ until the simulation begins. The response mechanisms in $\boldsymbol{\mathcal{R}}$ employed for this case strategic prevention resources that can accurately prevent or mitigate the existences/impacts of the disruptions the nodes. The strategic allocation decisions $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ can only be placed before the simulation begins, and no redeployment is possible. If a disruption affects a protected node, the disruption can be prevented or mitigated, and remaining disruption strength can continue to propagate disruption to nearby/connected nodes. With respect to $\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$, strategic

allocation in $\boldsymbol{\mathcal{R}}$ be deployed to any nodes in $\boldsymbol{\mathcal{C}}$, and an allocation protecting a node also provides limited protection to nearby/connected nodes. The allocation decisions are $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ also aware of the locations of the nodes in $\boldsymbol{\mathcal{C}}$ and potential disruption propagation directions $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$, which is expected in agricultural systems and computer networks. These $\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$ aspects are applicable to the problem contexts of agricultural system and computer networks, where cooperation between $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ is possible and necessary. With respect to $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$, a disruption in $\boldsymbol{\mathcal{D}}$ affecting a node in $\boldsymbol{\mathcal{C}}$ can propagate to the neighboring nodes, thus, network modeling can be applied, per the first CLOC guideline. Three important aspects of $\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$ are noted: (i) the strategic allocations decisions $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ are not aware of the location of the disruptions in $\boldsymbol{\mathcal{D}}$; (ii) the strategic allocation(s) in $\boldsymbol{\mathcal{R}}$ protecting a node can prevent or mitigate the disruptions in $\boldsymbol{\mathcal{D}}$; (iii) strategic allocation(s) in $\boldsymbol{\mathcal{R}}$ protecting a node can prevent or mitigate future disruptions (occurring from propagation.

Within the scope of the CSPD model, one decision type $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ is investigated: the strategic deployment of static response resources in $\boldsymbol{\mathcal{R}}$, called strategic allocation. The challenge of the $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ is that the targets of the disruptions are not known to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ ahead of time and can also propagate. Insufficient protection of certain parts of the client system $\boldsymbol{\mathcal{C}}$ can lead to more severe propagation $\boldsymbol{\mathcal{D}}$, reducing the system viability of $\boldsymbol{\mathcal{C}}$. The objective of the $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ decisions is to contain (or ideally), prevent the disruption propagation for a period of time (until the dynamic response mechanisms arrive, for example). The system performance metric $\boldsymbol{\mathcal{M}}$ investigated is the accumulative performance loss (or disruption damage) after certain periods of time, with the periods of time selected based on the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter. Four intervals for $\boldsymbol{\mathcal{M}}$ are selected: one-fourth of the network diameter $\boldsymbol{\mathcal{M}_1}$, one-half of the network diameter $\boldsymbol{\mathcal{M}_2}$, three-fourth of the network diameter $\boldsymbol{\mathcal{M}_3}$, and the full network diameter $\boldsymbol{\mathcal{M}_4}$.

A summary of the CSPD model formulation is provided in Table 5.1.

Table 5.1. Summary of CSPD description

CRDP Formulation Category	Item	Details	
CSPD Components	$\mathcal{C}$: client system	Nodes representing plants, plant locations, rooms, or buildings (depending on the context).	
	$\mathcal{R}$: response mechanisms	Strategic allocations cannot be changed when the simulation begins.	
	$\mathcal{D}$: disruption propagation	Disruptions are not known to $\mathcal{C}$ and $\mathcal{R}$ ahead of time. Disruptions can propagate if not fully prevented.	
CSPD Interactions	$\mathcal{C}$&$\mathcal{R}$: client-response interaction	Strategic allocation can be deployed in any node in $\mathcal{C}$. Strategic allocation can protect a node and its neighboring nodes.	
	$\mathcal{D}$&$\mathcal{C}$: disruption-client interaction	A disruption affecting a node can propagate to the node's neighboring nodes, cause more disruption(s).	
	$\mathcal{R}$&$\mathcal{D}$: response-disruption interaction	Strategic allocations are aware of potential disruption propagation directions. Strategic allocations do not know the locations of disruptions ahead of time. Strategic allocation can prevent or mitigate disruption propagation.	
CSPD Decision Space	$\mathcal{S}_{\mathcal{R}}$: strategic allocation decision	Each strategic allocation can be given to a node. Strategic allocations cannot be changed once	
CSPD System Performance Metrics	$\mathcal{M}_1$: total performance loss 1/4	Accumulative performance loss (or disruption damage) after certain periods of time	1/4 of the network diameter
	$\mathcal{M}_2$: total performance loss 1/2		1/2 of the network diameter
	$\mathcal{M}_3$: total performance loss 3/4		3/4 of the network diameter
	$\mathcal{M}_4$: total performance loss full		The full network diameter

5.2 CSPD Formulation

Based on the CSPD model description, the CRDP formulation of the CSPD model is as follows. The CSPD model is simulated using the one variation of the TIE/CRDP software presented in APPENDIX A. Each timestep $t \in \mathbb{Z}_{\geq 0}$ is a discrete timestep. The entities and attributes are given in Table 5.2, the system performance metrics are given in Table 5.3.

Table 5.2. Entities and attributes of the CSPD model

Type	Entity/Attribute and Explanation	CRDP domain
Input	$\boldsymbol{\mathcal{C}}$: $\mathrm{NL} = \{n_0, n_1, \dots\}$ Set of nodes, with each node $n \in \mathrm{NL}$ representing a component of the client system.	$\boldsymbol{\mathcal{C}}$
Input	$\boldsymbol{\mathcal{R}}$: $\mathrm{AL} = \{a_0, a_1, \dots\}$ Set of strategic allocations, with each strategic allocation $a \in \mathrm{AL}$ representing an allocation of strategic prevention resources.	$\boldsymbol{\mathcal{R}}$
Input	$\boldsymbol{\mathcal{R}}$: $\mathrm{APP} \in [0,1]$ Strategic allocation primary protection amount, which indicates the proportion of disruption prevented from affecting the primary node protected by a strategic allocation. A strategic allocation cannot be changed once disruptions begin.	$\boldsymbol{\mathcal{R}}$
Input	$\boldsymbol{\mathcal{D}}$: $\mathrm{DL} = \{d_0, d_1, \dots\}$ Set of disruptions, with each disruption $d \in \mathrm{DL}$ having the capability to disruption a node and if not prevented, can propagate further.	$\boldsymbol{\mathcal{D}}$
Input	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$: $\mathrm{EL} = \{e_0, e_1, \dots\}$ The set of bidirected edges, with each bidirected edge $e = \{n_i, n_j\}$ representing a potential disruption propagation direction from node n_i to node n_j and from node n_j to node n_i.	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
Derived	$\mathrm{DCD} \in \mathbb{Z}_{\geq 0}$ The disruption-client diameter, which is the diameter of the network model generated from NL and EL. Alternatively, this is also the greatest distance between any pair of nodes.	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
Input	$\boldsymbol{\mathcal{R}}$: $\mathrm{ASP} \in [0,1], \mathrm{ASP} < \mathrm{APP}$ Strategic allocation secondary protection amount. A strategic allocation protecting a node n also provides a lower amount of protection to the neighboring nodes of node n.	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
The following attributes are defined for each node $n \in$ NL		

Table 5.2. continued

Type	Entity/Attribute and Explanation	CRDP domain
Derived	$\mathrm{NPS}(n) \in [0,1]$ Node n's protection status, with value 0 denoting that node n is not protected, 1 if it is fully protected, and can take values between 0 and 1. Default value of 0.	$\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$
Dynamic	$\mathrm{NDS}(n,t) \in [0,1]$ Node n's disruption status at time t, with value 0 denoting that node n is not disrupted, 1 if it is fully disrupted, and can take values between 0 and 1. Default value of 0.	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
Derived	$\mathrm{NNL}(n) \subset \mathrm{NL}$ Node n's set of neighboring nodes, which includes all nodes with an edge pointing from n to it. $\mathrm{NNL}(n) = \{n_j \in \mathrm{NL}: \exists\{n, n_j\} \in \mathrm{EL}\}$	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
The following attributes are defined for each strategic allocation $\boldsymbol{a} \in$ AL		
Decision	$\mathrm{ASN}(a) \in \mathrm{NL}$ Strategic allocation a's selected node to protect, and cannot be changed once disruptions start. This decision type is made without information of $\mathrm{NDS}(n,t), \forall n \in \mathrm{NL}, \forall t$, and with full information of all other entities and variables. The definition of $\mathrm{ASN}(a)$ means one strategic allocation a can have one node as the primary protection target, and one node can have more than one strategic allocation assigned to it. The relationship between $\mathrm{ASN}(a)$ and $\mathrm{NPS}(n)$ is as follows: $\mathrm{NPS}(n) = \sum_{a}^{\mathrm{AL}} \begin{cases} \mathrm{APP}, & \text{if } \mathrm{ASN}(a) \equiv n \\ \mathrm{ASP}, & \text{if } \mathrm{ASN}(a) \in \mathrm{NNL}(n) \end{cases}$	$\boldsymbol{\mathcal{S}_{\mathcal{R}}}$

The nodes in the client system $\boldsymbol{\mathcal{C}}$ are represented by the set of nodes NL. The strategic allocations in $\boldsymbol{\mathcal{R}}$ are represented by the set of allocations AL. The disruptions in $\boldsymbol{\mathcal{D}}$ are represented by the set of disruptions DL, which causes the disruption status $\mathrm{NDS}(n,t)$ to increase. The allocations of strategic resources to the nodes in NL are represented by $\mathrm{ASN}(a)$. Each allocation increases the protection status $\mathrm{NPS}(n)$ of a node n by the amount APP for the node itself, and by the amount ASP for each of its neighboring nodes, increasing their resilience against disruption propagation. Per the first CLOC guideline, the disruption propagation directions are represented by the set of undirected edges EL, which is known to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{D}}$. This leads to the derivation of the set of neighboring nodes $NNL(n)$ for each node.

Following the specification of Table 5.1, the CSPD system performance metrics are given in Table 5.3.

Table 5.3. System performance metrics of the CSPD model

System Performance Metric	CRDP domain
$\mathrm{PL}(t) = \frac{\sum_{n}^{\mathrm{NL}} \mathrm{NDS}(n,t)}{\lvert \mathrm{NL} \rvert}$ Performance loss, which denotes the total fraction of the client system disrupted at time t. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, $\mathrm{PL}(t)$ is to be minimized.	$\boldsymbol{\mathcal{M}}$
$\mathrm{APL(t)} = \sum_{t_i=0,1,\ldots}^{t} \mathrm{PL}(t_i)$ Accumulative performance loss from time 0 to time t. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, $\mathrm{APL}(t)$ is to be minimized.	$\boldsymbol{\mathcal{M}}$
$\mathrm{TPL}_1 = \mathrm{APL}\left(\left\lfloor \frac{1}{4} * \mathrm{DCD} \right\rfloor\right)$ Total performance loss at one-fourth of the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, TPL_1 is to be minimized.	$\boldsymbol{\mathcal{M}}_1$
$\mathrm{TPL}_2 = \mathrm{APL}\left(\left\lfloor \frac{1}{2} * \mathrm{DCD} \right\rfloor\right)$ Total performance loss at one-half of the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, TPL_2 is to be minimized.	$\boldsymbol{\mathcal{M}}_2$
$\mathrm{TPL}_3 = \mathrm{APL}\left(\left\lfloor \frac{3}{4} * \mathrm{DCD} \right\rfloor\right)$ Total performance loss at three-fourth of the $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, TPL_3 is to be minimized.	$\boldsymbol{\mathcal{M}}_3$
$\mathrm{TPL}_4 = \mathrm{APL(DCD)}$ Total performance loss at maximum $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ network diameter. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, TPL_4 is to be minimized.	$\boldsymbol{\mathcal{M}}_4$

A small example of a CSPD case is provided in Figure 5.1.

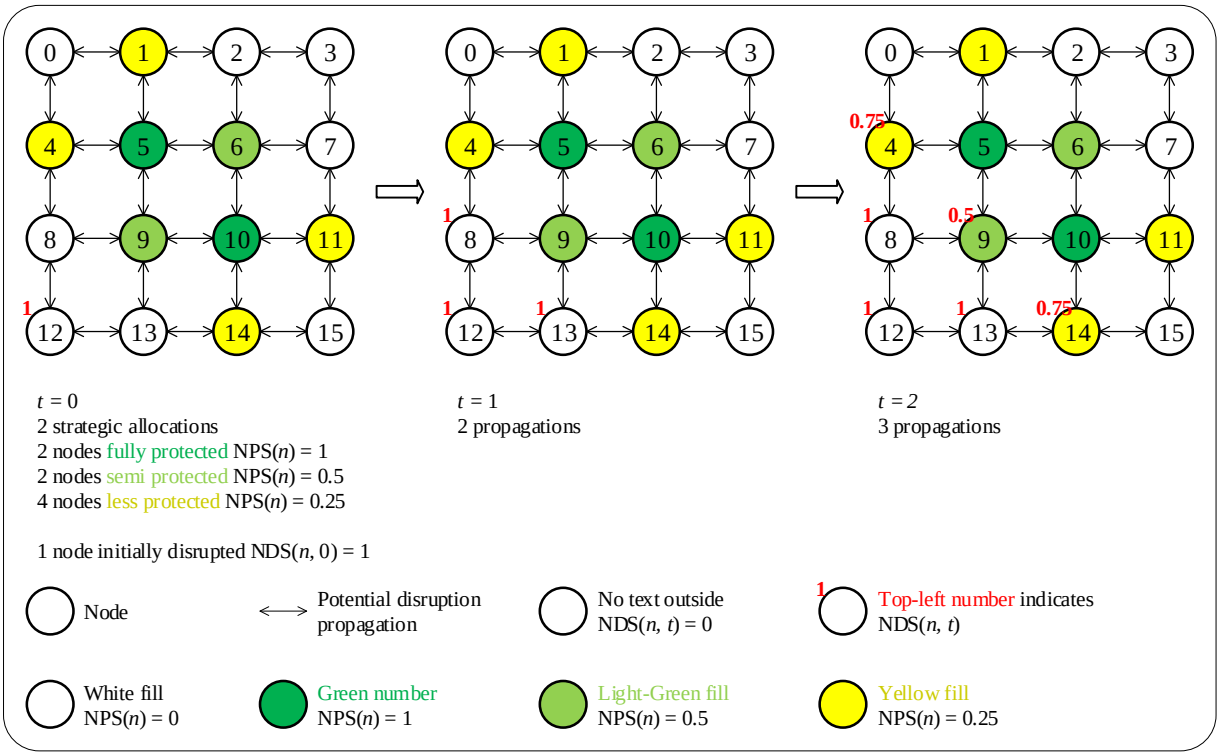

Figure 5.1. CSPD example

In this example, there are 16 nodes numbering 0 to 8 in NL, with 24 bidirected edges representing potential disruption propagation directions.

Two strategic allocations of $\text{APP} = 1, \text{ASP} = 0.25$ are assigned to n_5 and n_{10}, making $\text{NPS}(n_5) = \text{NPS}(n_{10}) = 1$, $\text{NPS}(n_6) = \text{NPS}(n_9) = 0.5, \text{NPS}(n_1) = \text{NPS}(n_4) = \text{NPS}(n_{11}) = \text{NPS}(n_{14}) = 0.25$. All others $\text{NPS}(n) = 0$.

One initial disruption d affects n_{12}, so $\text{NDS}(n_{12}, 0) \leftarrow 1 - \text{NPS}(n_{12}) = 1$. $\text{PL}(0) = 1/16$.

$t = 1$, 2 disruption propagations occur to $\text{NDS}(n_8, 1) = NDS(n_{13}, 1) = 1$. $\text{PL}(1) = 3/16$.

$t = 2$, 3 disruption propagations occur to $\text{NDS}(n_4, 2) = NDS(n_{14}, 2) = 0.75$, and $\text{NDS}(n_9, 2) = 0.5$. $\text{PL}(2) = 5/16$.

$t = 3$, 2 disruption propagations occur to $\text{NDS}(n_0, 3) = \text{NDS}(n_{15}, 3) = 0.75$. $\text{PL}(3) = 6.5/16$.

$t = 4$, 2 disruption propagations occur to $\text{NDS}(n_1, 4) = \text{NDS}(n_{11}, 4) = 0.5$. $\text{PL}(4) = 7.5/16$.

$t = 5$, 2 disruption propagations occur to $\text{NDS}(n_2, 5) = \text{NDS}(n_7, 5) = 0.5$. $\text{PL}(5) = 8.5/16$.

$t = 6$, 1 disruption propagation occurs to $\text{NDS}(n_3, 6) = 0.5$. $\text{PL}(6) = 9/16$.

This leads to $\boldsymbol{\mathcal{M}}_1 = TPL_1 = \mathrm{APL}(1) = 4/16, \boldsymbol{\mathcal{M}}_2 = \mathrm{TPL}_2 = \mathrm{APL}(3) = 15.5/16, \boldsymbol{\mathcal{M}}_3 = \mathrm{TPL}_3 = \mathrm{APL}(4) = 23/16, \boldsymbol{\mathcal{M}}_4 = \mathrm{TPL}_4 = \mathrm{APL}(6) = 40.5/16.$

The complete simulation pseudocode of the CSPD model is provided in Table 5.4.

Table 5.4. Simulation pseudocode of the CSPD model

Step	**Pseudocode**	**CRDP domain**
Step 1	$t \leftarrow 0$, Initialize NL, AL, AP, DL, EL, AS	Simulation
Step 2	Decide $\mathrm{ASN}(a)$ for all $a \in \mathrm{AL}$	$\boldsymbol{\mathcal{S}_{\mathcal{R}}}$
Step 3	Foreach $n \in \mathrm{NL}$ $\mathrm{NPS}(n) \leftarrow \mathrm{NPS}(n) + \sum_{a}^{\mathrm{AL}} \begin{cases} \mathrm{APP}, & \text{if } \mathrm{ASN}(a) \equiv n \\ \mathrm{ASP}, & \text{if } \mathrm{ASN}(a) \in \mathrm{NNL}(n) \end{cases}$ Next n	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
Step 4	Foreach $d \in \mathrm{DL}$, Randomly select $n \in \mathrm{NL}$ without overlap $\mathrm{NDS}(n, 0) \leftarrow 1 - \mathrm{NPS}(n)$ Next d	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
Step 5	For $t \coloneqq 1$ to $t_{\max}$	Simulation
Step 5.1	Foreach $n \in \mathrm{NL}$ $\mathrm{NDS}(n, t) \leftarrow \max\left(NDS(n, t-1), \max_{n_j \in \mathrm{NNL}(n)} \mathrm{NDS}(n_j, t-1) - \mathrm{NPS}(n)\right)$	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
Step 5.2	$\mathrm{PL}(t) \leftarrow \frac{\sum_{n}^{\mathrm{NL}} \mathrm{NDS}(n,t)}{\lvert\mathrm{NL}\rvert}$	$\boldsymbol{\mathcal{M}}$
Step 5.3	Next t	Simulation
Step 6	Compute $\boldsymbol{\mathcal{M}}$	$\boldsymbol{\mathcal{M}}$

In Table 5.4, Step 1 initializes $t \leftarrow 0$, the main inputs of the CSPD model, which includes the set of nodes NL, the set of strategic allocations AL, the primary protection amount AP, the set of disruptions DL, the set of bidirected edges EL, and the secondary protection amount AS.

Step 2 decides $\boldsymbol{S}_{\boldsymbol{R}}$ and assign the strategic resources $a \in \mathrm{AL}$ to the nodes $n \in \mathrm{NL}$ by setting $\mathrm{ASN}(a)$, without knowledge of the targeting of DL. These decisions can be supported by analytics and protocols, which are discussed below in the following section.

Step 3 actuates the strategic protection status of each node $n \in \mathrm{NL}$. Each node n receives an increase to protection status $\mathrm{NPS}(n)$ of APP for each strategic allocation a assigned to it through $\mathrm{ASN}(a) \equiv n$ and an increase of ASP for each strategic allocation a assigned to any of its neighboring nodes.

Step 4 initializes the disruptions. Each disruption $d \in \mathrm{DL}$ chooses a different node $n \in \mathrm{NL}$ to target, and if the node n is not protected (which means $\mathrm{NPS}(n) = 0$), the disruption status is set to $\mathrm{NDS}(n, 0) \leftarrow 1$. If the node is protected with $\mathrm{NPS}(n) > 0$, the disruption status is set to $\mathrm{NDS}(n, 0) \leftarrow 1 - \mathrm{NPS}(n)$. In this case, the selected distribution is random uniform.

Step 5 begins the dynamic simulation of the system. The simulation ends when the maximum time $t_{\max}$ is reached or the system state no longer changes.

Step 5.1 propagates the disruptions affecting each node $n \in \mathrm{NL}$. A node will receive retain its current disruption status $\mathrm{NDS}(n, t) \leftarrow \mathrm{NDS}(n, t-1)$ or receive a higher disruption status from one of its neighboring nodes n_j, setting $\mathrm{NDS}(n, t) \leftarrow \mathrm{NDS}\big(n_j, t-1\big) - \mathrm{NPS}(n)$.

Step 5.2 calculates the current performance loss of the client system $\mathrm{PL}(t)$. Then, the simulation returns to step 5, incrementing $t \leftarrow t + 1$.

Step 6 marks the end of one simulation replication and calculates $\boldsymbol{\mathcal{M}}_1$,$\boldsymbol{\mathcal{M}}_2$, $\boldsymbol{\mathcal{M}}_3$ and $\boldsymbol{\mathcal{M}}_4$.

5.3 CSPD Analytics and Protocols

In this subsection, the CSPD analytics and protocols are developed based on the CLOC principle to support the decision-making of $\boldsymbol{S}_{\boldsymbol{R}}$.

The analysis of the CSPD model and decision space is as follows. Based on Step 4 and Step 5.1 of Table 5.4, which entails the disruption propagation behavior, any node $n \in \mathrm{NL}$ affected by a positive initial disruption (from Step 4) will stay affected by an amount of disruption status greater or equal than the original disruption status. Namely,

$$\mathrm{NDS}(n,0) > 0 \Rightarrow \mathrm{NDS}(n,t) \geq \mathrm{NDS}(n,0) > 0, \forall t \tag{18}$$

This is because the strategic allocation of node n only has one opportunity to prevent disruptions and their propagation affecting node n, according to Step 4. This situation occurs if $\mathrm{NPS}(n) < 1$, and if node n is a disruption target, $\mathrm{NDS}(n,0) = 1 - \mathrm{NPS}(n) > 0$, or

$$\mathrm{NPS}(n) < 1 \Rightarrow \mathrm{NDS}(n,0) = 1 - \mathrm{NPS}(n) > 0 \Rightarrow \mathrm{NDS}(n,t) \geq 1 - \mathrm{NPS}(n) > 0, \forall t \tag{19}$$

Analyzing $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$ at the local level, Step 4, and Step 5.1, it is observed that a disruption $d \in \mathrm{DL}$ initially affecting node n can also potentially propagate its effects to the neighboring nodes $n_j \in \mathrm{NNL}(n)$. Thus,

$$\begin{aligned} \mathrm{NDS}(n,0) > 0 \Rightarrow \forall n_j \in \mathrm{NNL}(n), \mathrm{NDS}(n,0) > \mathrm{NPS}(n_j) \\ \Rightarrow \mathrm{NDS}(n_j,t) \geq \mathrm{NDS}(n_j,0) > 0, \forall t \geq 1 \end{aligned} \tag{20}$$

This relationship also applies to n itself. If there exists a stronger disruption coming from one of its neighboring nodes n_j, the value of $\mathrm{NDS}(n,t)$ would increase.

Using the previous equation, the analysis can be extended to the network level. The distance matrix is defined as $\mathrm{DIST}: \mathrm{NL} \times \mathrm{NL} \rightarrow \mathbb{Z}_{\geq 0}$, which can be computed with the information from EL and the Floyd-Warshall algorithm (Floyd, 1962; Rossi & Ahmed, 2015). Each term of the distance matrix $\mathrm{DIST}(n_i, n_j) \in \mathbb{Z}_{\geq 0}$ denotes the shortest-path distance between n_i and n_j. A path $\mathrm{PATH}(n_i, n_j) = (n_i, n_{k0}, \dots, n_j)$ from node n_i to node n_j is defined as any path between n_i and n_j without revisiting nodes (or $n_i \neq n_{ko} \neq \cdots \neq n_j$), and the length of the path is $|\mathrm{PATH}(n_i, n_j)| \geq \mathrm{DIST}(n_i, n_j) + 1$. The case $|\mathrm{PATH}(n_i, n_j)| = \mathrm{DIST}(n_i, n_j) + 1$ is when the path is a shortest path

between n_i and n_j. Define $\mathrm{PAL}(n_i, n_j)$ as the set of all such paths, the following observation is made

$$\mathrm{NDS}(n_i, 0) > 0 \Rightarrow \exists \mathrm{PATH}(n_i, n_j) \in \mathrm{PAL}(n_i, n_j): 1 > \sum_{n}^{\mathrm{PATH}(n_i, n_j)} \mathrm{NPS}(n)$$
$$\Rightarrow \mathrm{NDS}(n_j, t) > 0, \forall t \geq \mathrm{DIST}(n_i, n_j) \quad (21)$$

This observation is explained as follows. A disruption affecting node n_i can potentially propagate to node n_j if it can find a path from node n_i to node n_j, $\mathrm{PATH}(n_i, n_j)$, that satisfies the condition that the sum of all the strategic resources $\mathrm{NPS(n)}$ of the nodes $(n_i, n_{k0}, n_{k1}, \dots, n_j)$ along the path is less than 1. This observation also implies that shorter paths are more demanding in terms of $\mathrm{NPS}(n)$. With this observation, there is sufficient mathematical understanding to proceed to the development of CSPD analytics.

5.3.1 CSPD analytics

The local level analysis of $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$ leads to the first CSPD analytic:

$\boldsymbol{\mathcal{A}}_1$: Strategic allocation to a node with a higher degree can better protect the client system from disruptions and their propagation. This means

$$\mathrm{ASN}(a) = n, |\mathrm{NNL}(n)| \nearrow \Rightarrow (\mathrm{AP} + \mathrm{AS} \times |\mathrm{NNL}(n)|) \nearrow \Rightarrow \sum_{n_j}^{\mathrm{NL}} \mathrm{NPS}(n_j) \nearrow \quad (22)$$

The above equation shows that the higher $|\mathrm{NNL}(n)|$ is, the higher the potential increase to the total protection status of the client system $\sum_{n_j}^{\mathrm{NL}} \mathrm{NPS}(n_j)$. The implication also extends to $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$ at the local level.

$$|\mathrm{NNL}(n)| \nearrow \Rightarrow \max_{n_j \in \mathrm{NNL}(n)} \{\mathrm{NDS}(n_j, 0)\} \nearrow \Rightarrow \mathrm{NDS}(n, 1) \nearrow \quad (23)$$

To extend the analytic $\boldsymbol{\mathcal{A}_1}$ to the network level, a network-level centrality measure needs to be defined. The harmonic centrality value of a node n is defined as $\mathrm{NHC}(n) \in \mathbb{R}_{\geq 0}$, and is the harmonic mean of all the distances between all pairs of different nodes of the network (Warshall, 1962).

$$\mathrm{NHC}(n) = \sum_{n_j}^{\mathrm{NL}-\{n\}} \frac{1}{\mathrm{DIST}(n, n_j)} \tag{24}$$

From the above equation, the second CSPD analytic is developed. Other network-level centrality measures, or a combination thereof, can also be employed.

$\boldsymbol{\mathcal{A}_2}$: Strategic allocation to a node with higher centrality can better protect the client system from disruptions and their propagation. This is because a more central node is involved in a higher number of shorter paths between any two pair of nodes in the network.

$$\mathrm{NHC}(n) \nearrow \Rightarrow \left(\sum_{n_i, n_j}^{\mathrm{NL}} \sum_{\mathrm{PATH}(n_i, n_j) : n \in \mathrm{PATH}(n_i, n_j)}^{\mathrm{PL}(n_i, n_j)} \frac{1}{\left|\mathrm{PATH}(n_i, n_j)\right|} \right) \nearrow \tag{25}$$

Thus, placing a strategic allocation to such nodes would increase the total protection status of the paths compared to the size of the path

$$\mathrm{ASN}(a) = n, \mathrm{NHC}(n) \nearrow \Rightarrow \left(\sum_{n_i, n_j}^{\mathrm{NL}} \sum_{\mathrm{PATH}(n_i, n_j) : n \in \mathrm{PATH}(n_i, n_j)}^{\mathrm{PL}(n_i, n_j)} \sum_{n_k}^{\mathrm{PATH}(n_i, n_j)} \frac{\mathrm{NPS}(n_k)}{\left|\mathrm{PATH}(n_i, n_j)\right|} \right) \nearrow \tag{26}$$

Placing $\mathrm{ASN}(a) = n$ also means an increase of AS to node n's neighboring nodes. Due to the restriction of $\mathrm{NPS}(n) \in [0,1]$, allocation of strategic resources should consider existing allocations. This leads to the third CSPD analytic

$\boldsymbol{\mathcal{A}_3}$: A new strategic allocation to a node should avoid resource redundancy. This analytic defines the redundant prevention resources allocated to a node n as $\mathrm{RPA}(n) \in \mathbb{R}_{\geq 0}$

$$\mathrm{RPA}(n) = |(\mathrm{AP} \times |\{a \in \mathrm{AL}: \mathrm{ASN}(a) \equiv n\}|) + \mathrm{AP} \times |\{a \in \mathrm{AL}: \mathrm{ASN}(a) \in \mathrm{NNL}(n)\}| - 1| \quad (27)$$

Basically, $\mathrm{RPA}(n) > 0$ if there exists redundant protection status, because $\mathrm{NPS}(n)$ is reduced to 1. This leads to the next analytic

$\boldsymbol{\mathcal{A}_4}$: A new strategic allocation to a node should concurrently consider the protection status of other nodes. This means a strategic allocation decision $\mathrm{ASN}(a)$ should consider $\mathrm{NPS}(n)$ of all $n \in$ NL to reduce redundancy and maximize the protection of the nodes.

The four developed CSPD analytics and their corresponding CLOC guidelines are summarized in Table 5.5.

Table 5.5. Summary of the CSPD analytics

Analytic	**Description**	**CLOC guideline**
$\boldsymbol{\mathcal{A}_1}$	Strategic allocation to a node with a higher degree can help protecting the client system better.	CLOC 1, CLOC 2
$\boldsymbol{\mathcal{A}_2}$	Strategic allocation to a node with higher centrality can help protecting the client system better.	CLOC 1, CLOC 2
$\boldsymbol{\mathcal{A}_3}$	Strategic allocations should avoid redundancy.	CLOC 3
$\boldsymbol{\mathcal{A}_4}$	A new strategic allocation to a node should concurrently consider the protection status of other nodes.	CLOC 3

5.3.2 CSPD protocols

Based on the 4 aforementioned CSPD analytics, 6 CSPD protocols $\boldsymbol{\mathcal{P}}$ to support decision-making $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ are developed.

$\boldsymbol{\mathcal{P}}_1$: Random allocation protocol, in which $\mathrm{ASN}(a)$ is selected randomly from $\{n \in \mathrm{NL}: \forall a' \in \mathrm{AL}: \mathrm{ASN}(a') \neq n\}$, with $\mathrm{ASN}(a')$ denoting the existing allocations. This protocol is a baseline protocol and is specified for the purpose of comparison. This protocol is not supported by analytics.

$\boldsymbol{\mathcal{P}}_2$: Degree centrality allocation protocol, in which $\mathrm{ASN}(a)$ is selected from $\{n \in \mathrm{NL}: \forall a' \in \mathrm{AL}: \mathrm{ASN}(a') \neq n\}$ sorted by $|\mathrm{NNL}(n)|$ in descending order, with $\mathrm{ASN}(a')$ denoting the existing allocations. The selection restricts the number of strategic allocations per node to one in order to avoid the case of a node being selected excessively. This protocol is supported by $\boldsymbol{\mathcal{A}}_1$.

$\boldsymbol{\mathcal{P}}_3$: Harmonic centrality allocation protocol, in which $\mathrm{ASN}(n)$ is selected from $\{n \in \mathrm{NL}: \forall a' \in \mathrm{AL}: \mathrm{ASN}(a') \neq n\}$ sorted by $\mathrm{NHC}(n)$ in descending order, with $\mathrm{ASN}(a')$ denoting the existing allocations. The selection restricts the number of strategic allocations per node to one in order to avoid the case of a node being selected excessively. This protocol is supported by $\boldsymbol{\mathcal{A}}_2$ at a limited level. It is noted that both $\boldsymbol{\mathcal{P}}_2$ and $\boldsymbol{\mathcal{P}}_3$ focus entirely on the topology of NL and EL, and does not consider the protection status of other nodes. The next 3 CSPD protocols are created to address this limitation.

$\boldsymbol{\mathcal{P}}_4$: CLOC – local coverage allocation protocol, in which $\mathrm{ASN}(a)$ is sequentially selected from $\{n \in \mathrm{NL}\}$ sorted by node n's local coverage index $\mathrm{NLCI}(n) \in \mathbb{R}$ in descending order, with $\mathrm{ASN}(a')$ denoting the existing allocations. It is noted that the selection pool becomes $\{n \in \mathrm{NL}\}$ to account for the case of multiple strategic allocations to a node. The index $\mathrm{NLCI}(n)$ is updated every single allocation $\mathrm{ASN}(a')$ in a sequential manner, and is defined as:

$$\mathrm{NLCI}(n) = \max\left\{0; 1 - \mathrm{AP} \times \left|\{a': ASN(a') \equiv n\}\right| - \mathrm{AS} \times \left|\{a': \mathrm{ASN}(a') \in \mathrm{NNL}(n)\}\right|\right\} + \sum_{n_j}^{\mathrm{NNL}(n)} \max\left\{0; \left(1 - AP \times \left|\{a': ASN(a') \equiv n_j\}\right| - \mathrm{AS} \times \left|\{a': \mathrm{ASN}(a') \in \mathrm{NNL}(n_j)\}\right|\right)\right\} \quad (28)$$

The first max{ } evaluates the unprotected portion of node n, and the summation sign $\sum$ evaluates the total unprotected portion of its neighboring nodes. The higher $\mathrm{NLCI}(n)$, the more

vulnerable the node and its neighbors become. This protocol utilizes all of the CSPD analytics $\boldsymbol{\mathcal{A}}$, although $\boldsymbol{\mathcal{A}}_2$ is used at a limited level and $\boldsymbol{\mathcal{A}}_4$ is used at the local level.

$\boldsymbol{\mathcal{P}}_5$: CLOC – harmonic coverage allocation protocol, in which $\mathrm{ASN}(a)$ is sequentially selected from $\{n \in \mathrm{NL}: \forall a' \in \mathrm{AL}: \mathrm{ASN}(a') \neq n\}$ sorted by node n's harmonic-based coverage index $\mathrm{NHCI}(n) \in \mathbb{R}$ in descending order, with $\mathrm{ASN}(a')$ denoting the existing allocations. The index $\mathrm{NHCI}(n)$ extends from $\mathrm{NLCI}(n)$, and is defined as:

$$\mathrm{NHCI}(n) = \mathrm{NLCI}(n) + \sum_{n_j}^{\mathrm{NL}-\{n\}} \frac{\mathrm{NLCI}(n)}{\mathrm{DIST}(n, n_j)} \tag{29}$$

Evidently, utilizing $\boldsymbol{\mathcal{P}}_5$ requires $\mathrm{NLCI}(n)$ to be fully updated and computed after every allocation $\mathrm{ASN}(a')$. The main benefit of this protocol is the incorporation of harmonic centrality into the index. This protocol utilizes all of the CSPD analytics $\boldsymbol{\mathcal{A}}$, although $\boldsymbol{\mathcal{A}}_2$ is used at a limited level.
The main limitation of $\boldsymbol{\mathcal{P}}_4$ and $\boldsymbol{\mathcal{P}}_5$ is their incapability to consider the network path-based disruption propagation. $\boldsymbol{\mathcal{P}}_4$ is limited to the local and neighboring level, whereas $\boldsymbol{\mathcal{P}}_5$ only considers the shortest path distance. To address this limitation, the next protocol $\boldsymbol{\mathcal{P}}_6$ is developed.

$\boldsymbol{\mathcal{P}}_6$: CLOC – global coverage allocation protocol, in which $\mathrm{ASN}(a)$ is sequentially selected from $\{n \in \mathrm{NL}: \forall a' \in \mathrm{AL}: \mathrm{ASN}(a') \neq n\}$. To efficiently consider all the possible disruption propagation paths, a simulation-based approach is employed. The following pseudocode, presented in Table 5.6, entails the protocol.

Table 5.6. Protocol pseudocode of the CSPD model

Step	Pseudocode
Step 1	Foreach $a \in \mathrm{AL}$
Step 2	Foreach $n \in \mathrm{NL}$, Define $\mathrm{TPM}(n) \leftarrow 0$ #Note: $\mathrm{TPM} =$ temporary performance metric
Step 3	Foreach $n_i \in \mathrm{NL}$
Step 4	Try $\mathrm{ASN}(a) = n_i$, update $\mathrm{NPS}(n), \forall n \in \mathrm{NL}$ accordingly
Step 5	Foreach $n_j \in \mathrm{NL}$
Step 6	Try one single disruption affecting n_j, simulate until a certain t
Step 7	$\mathrm{TPM}(a) \leftarrow \mathrm{TPM}(a) + \sum_{n}^{\mathrm{NL}} \mathrm{NDS}(n,t) \,/\lvert\mathrm{NL}\rvert$
Step 8	Next n_j
Step 9	$\mathrm{ASN(a)} \leftarrow \arg\min_{n \in \mathrm{NL}} \mathrm{TPM}(n)$
Step 10	Next n_i
Step 11	Next a

This approach subsequently tries to allocate strategic resources to each node, and evaluates each allocation based on its performance against the different cases of single disruption attacking different nodes. Each allocation iteration picks the best node to allocate to, and future iterations consider previous iterations when simulating the disruptions.

The six CSPD protocols are summarized in Table 5.7.

Table 5.7. Summary of the CSPD protocols

Analytic	Description	Collaboration level	Related analytics
$\boldsymbol{\mathcal{P}}_1$	Random allocation protocol: baseline, random.	None	None
$\boldsymbol{\mathcal{P}}_2$	Degree centrality allocation protocol: prioritizes higher node degree.	Low	$\boldsymbol{\mathcal{A}}_1$
$\boldsymbol{\mathcal{P}}_3$	Harmonic centrality allocation protocol: prioritizes higher harmonic centrality.	Low	$\boldsymbol{\mathcal{A}}_2$
$\boldsymbol{\mathcal{P}}_4$	CLOC - Local coverage allocation protocol: considering the existing protection statuses of neighboring nodes.	High	$\boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4$
$\boldsymbol{\mathcal{P}}_5$	CLOC - Harmonic coverage allocation protocol: considering the existing protection statuses of all nodes using harmonic centrality.	High	$\boldsymbol{\mathcal{A}}_2, \boldsymbol{\mathcal{A}}_3, \boldsymbol{\mathcal{A}}_4$
$\boldsymbol{\mathcal{P}}_6$	CLOC - Global coverage allocation protocol: considering the existing protection statuses of all nodes through path-based simulation.	High	All $\boldsymbol{\mathcal{A}}$

5.4 Numerical Experiments and Results

Numerical experiments are conducted to validate the CSPD model, analytics, and protocols. The factors of the experiments include: four network types, ten response/disruption scenarios, and six protocols (from $\boldsymbol{\mathcal{P}}_1$ to $\boldsymbol{\mathcal{P}}_6$) with 100 replications for each factor combination, resulting in 30,000 runs in total. The number of replications is selected as a balance between ensuring sufficient coverage of different disruption target combinations and ensuring reasonable total runtime of the experiments (due to the computationally expensive nature of the CSPD protocol $\boldsymbol{\mathcal{P}}_6$). The four system performance metrics $\boldsymbol{\mathcal{M}}_1$, $\boldsymbol{\mathcal{M}}_2$, $\boldsymbol{\mathcal{M}}_3$, and $\boldsymbol{\mathcal{M}}_4$ (all minimization objectives) are reported.

The four network types are:

1. GO: 10x10 grid orthogonal both-way propagation;
2. BA: 100-node random Barabasi-Albert with $m_0 = 2, m = 2$ with bidirectional edges;
3. ER: 100-node random Erdos-Renyi with $p = 0.08$ with bidirectional edges;
4. WS: 100-node random Watts-Strogatz with $k = 4, \beta = 0.5$ with bidirectional edges.

The network type GO is selected due to its applicability to the agricultural settings, particularly greenhouses (Marchiori & Latora, 2000; Dusadeerungsikul & Nof, 2019). The BA, ER, and WS network types are selected because these random network models are common choices for complex networks research and cyber-physical systems research (Arora & Ventresca, 2017; Dusadeerungsikul et al., 2018).

The selected amount for APP is 1, and ASP is 0.3. The ten response/disruption scenarios are:

1. R10D10: 10 strategic allocations, 10 initial disruptions.
2. R20D10: 20 strategic allocations, 10 initial disruptions.
3. R30D10: 30 strategic allocations, 10 initial disruptions.
4. R40D10: 40 strategic allocations, 10 initial disruptions.
5. R50D10: 50 strategic allocations, 10 initial disruptions.
6. R10D20: 10 strategic allocations, 20 initial disruptions.
7. R20D20: 20 strategic allocations, 20 initial disruptions.
8. R30D20: 30 strategic allocations, 20 initial disruptions.
9. R40D20: 40 strategic allocations, 20 initial disruptions.
10. R50D20: 50 strategic allocations, 20 initial disruptions.

Additionally, a separate set of experiments is conducted on an enterprise's internal email network, using the six aforementioned CSPD protocols and ten response/disruption scenarios, with 100 replications for each factorial combination (subsection 5.4.4).

5.4.1 Comparison by CSPD protocols

The comparison between CSPD protocols is provided in Figure 5.2 and Table 5.8.

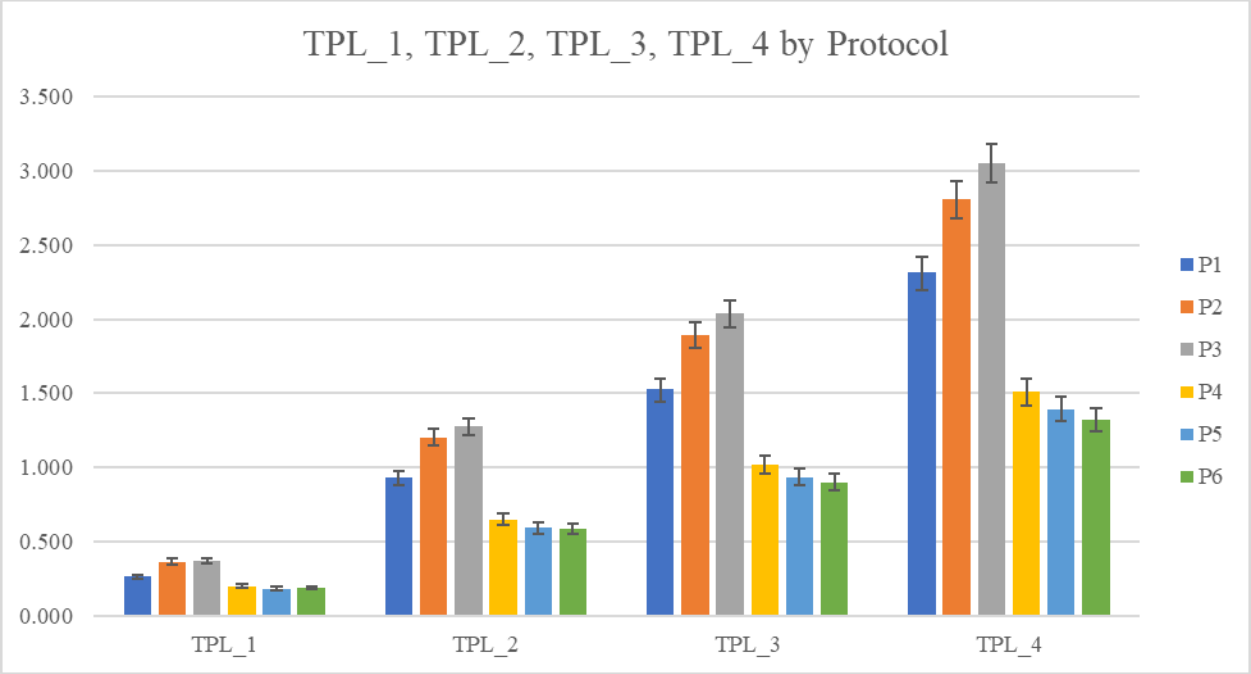

Figure 5.2. CSPD experiment results grouped by CSPD protocols with 95% confidence interval bars

Table 5.8. CSPD experiment results grouped by CSPD protocols

Protocol	$\mathcal{M}_1$: TPL_1	$\mathcal{M}_2$: TPL_2	$\mathcal{M}_3$: TPL_3	$\mathcal{M}_4$: TPL_4
$\mathcal{P}_1$	0.263*	0.932*	1.530*	2.319*
$\mathcal{P}_2$	0.363**	1.201**	1.890**	2.805**
$\mathcal{P}_3$	0.367**	1.274**	2.035**	3.048**
$\mathcal{P}_4$	0.199***	0.648***	1.017***	1.509***
$\mathcal{P}_5$	0.181***	0.591***	0.934***	1.394***
$\mathcal{P}_6$	0.187***	0.581***	0.900***	1.322***
Gap of CLOC protocols versus other protocols	31.1% - 50.7%	37.6% - 54.4%	41.2% - 55.8%	43.0% - 56.6%

*, **, ***: group of confidence interval overlapping
Same group means no significant statistical difference between the different CDUD protocols of the same group.
Different groups mean significant statistical differences between any pair of CDUD protocols belonging to different groups.

The experiment results (Figure 5.2 and Table 5.8) are the system performance metrics TPL_1, TPL_2, TPL_3, and TPL_4 averaged across all network types and response/disruption scenarios. For the purpose of comparison, the group of CSPD protocols $\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$ are referred to as the non-CLOC protocols, and the CSPD protocols $\boldsymbol{\mathcal{P}}_4, \boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$ are referred to as the CLOC protocols. With respect to overall performance, the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) outperform the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) with statistical significance, ranging from 31.1% to 56.6%. These performance gaps apply to all four system performance metrics TPL_1, TPL_2, TPL_3, and TPL_4. The gaps (Table 5.8) increase as TPL becomes more long-term, with 31.1% - 50.7% for TPL_1, 37.6% - 54.4% for TPL_2, 41.2% - 55.8% for TPL_3, and 43.0% - 56.6% for TPL_4. Notably, the performances of the CLOC protocols $\boldsymbol{\mathcal{P}}_4, \boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$ are not different with statistical significance, which could be caused by the errors from different factors (network types and disruption/response scenarios). These results (Table 5.8) indicate that, in general, the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) outperform the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) with statistical significance.

5.4.2 Comparison by CSPD protocols and network types

The comparisons between CSPD protocols, grouped by network types, are provided in Figure 5.3, Figure 5.4, and Table 5.9. Similar to 5.4.1, the group of CSPD protocols $\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$ are referred to as the non-CLOC protocols, and the CSPD protocols $\boldsymbol{\mathcal{P}}_4, \boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$ are referred to as the CLOC protocols.

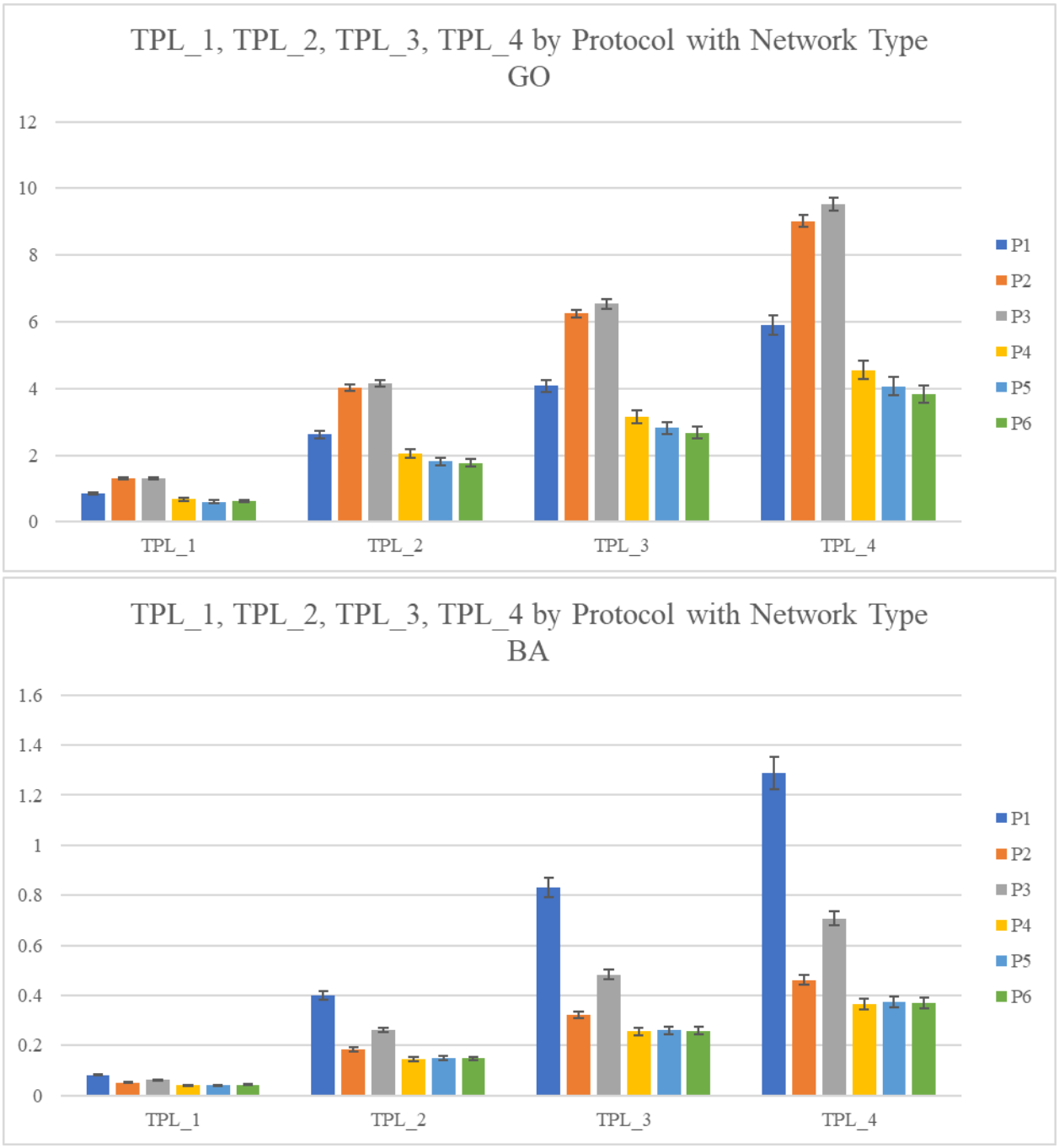

Figure 5.3. CSPD experiment results grouped by CSPD protocols and network types GO and BA, with 95% confidence interval bars

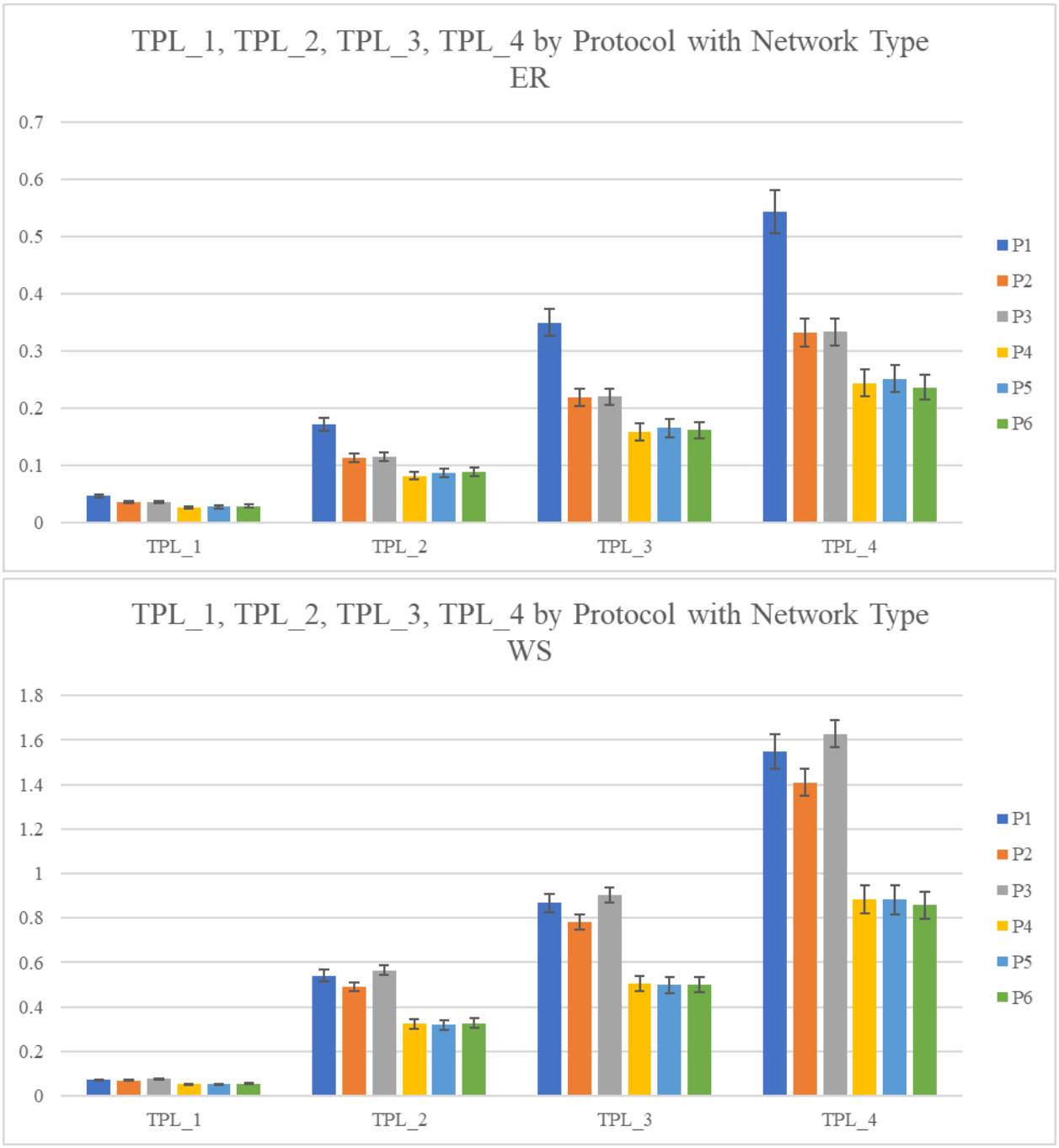

Figure 5.4. CSPD experiment results grouped by CSPD protocols and network types ER and WS, with 95% confidence interval bars

Table 5.9. CSPD experiment results grouped by network types

Network Type	Number of nodes	Number of edges	Rank	Performance metrics range	Performance gaps between CLOC protocols versus other protocols
GO	100	180	4 (worst)	TPL_1: 0.60 - 1.30 TPL_2: 1.76 - 4.16 TPL_3: 2.68 - 6.54 TPL_4: 3.83 - 9.53	TPL_1: 29.0% - 53.4% TPL_2: 32.7% - 57.6% TPL_3: 34.2% - 59.0% TPL_4: 35.1% - 59.8%
BA		200	2	TPL_1: 0.04 - 0.08 TPL_2: 0.15 - 0.40 TPL_3: 0.26 - 0.83 TPL_4: 0.37 - 1.29	TPL_1: 20.0% - 49.8% TPL_2: 20.4% - 63.5% TPL_3: 20.7% - 69.2% TPL_4: 20.8% - 71.6%
ER		800	1 (best)	TPL_1: 0.03 - 0.05 TPL_2: 0.08 - 0.17 TPL_3: 0.16 - 0.35 TPL_4: 0.24 - 0.54	TPL_1: 27.4% - 44.0% TPL_2: 27.9% - 52.2% TPL_3: 27.4% - 54.7% TPL_4: 28.8% - 56.5%
WS		200	3	TPL_1: 0.05 - 0.08 TPL_2: 0.32 - 0.57 TPL_3: 0.50 - 0.90 TPL_4: 0.86 - 1.63	TPL_1: 25.7% - 33.8% TPL_2: 34.7% - 43.5% TPL_3: 36.2% - 44.8% TPL_4: 39.2% - 47.3%

The superior performance of the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) is still present with statistical significance with all the different network types. It is noted that while the network types GO, BA, and WS have the same number of nodes (200) and roughly the same number of edges (180 to 200), and the performance metrics ranges are still significantly different. With respect to performance metrics, the gaps between the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) and the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1$, $\boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) are 20% - 53.4% for TPL_1, 20.4% - 63.5% for TPL_2, 20.7% - 69.2% for TPL_3, and 20.8% - 71.6% for TPL_4. With respect to network types, the gaps between the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) and the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1$, $\boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) are 29% - 59.8% for GO, 20.0% - 71.6% for BA, 27.4% - 56.5% for ER, and 25.7% - 47.3% for WS. These results (Table 5.9) indicate that the CLOC protocols significantly outperform the non-CLOC protocols in all the investigated network types.

5.4.3 Comparison by CSPD protocol groups and response/disruption scenarios

The comparisons between two protocol groups (non-CLOC $\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2, \boldsymbol{\mathcal{P}}_3$ and CLOC $\boldsymbol{\mathcal{P}}_4, \boldsymbol{\mathcal{P}}_5, \boldsymbol{\mathcal{P}}_6$), grouped by response/disruption scenarios, are provided in Figure 5.5, Figure 5.6, and Table 5.10. Note in these two figures, N standards for non-CLOC, and C standards for CLOC.

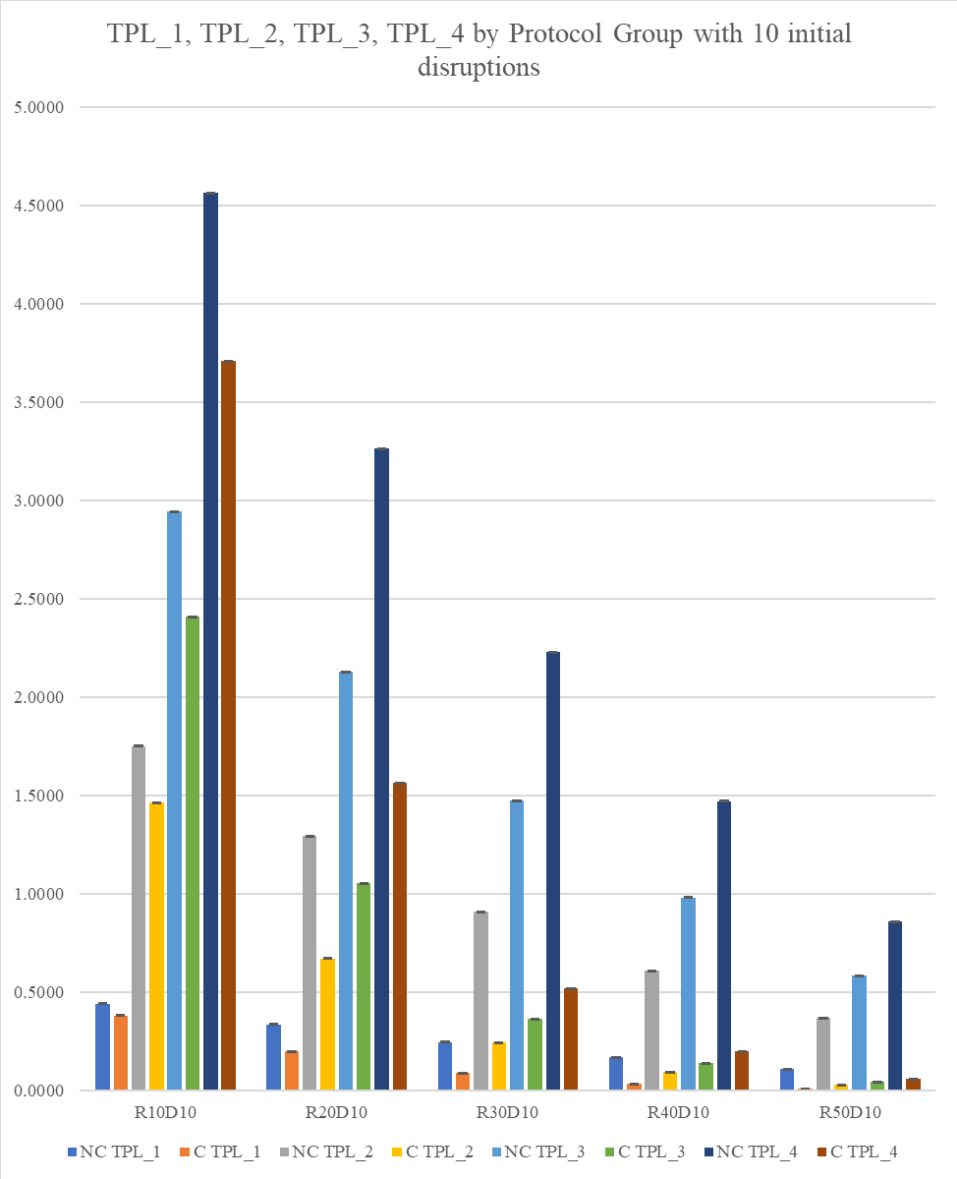

Figure 5.5. CSPD experiment results grouped by CSPD protocol groups and response/disruption scenarios with 10 initial disruptions, with 95% confidence interval bars

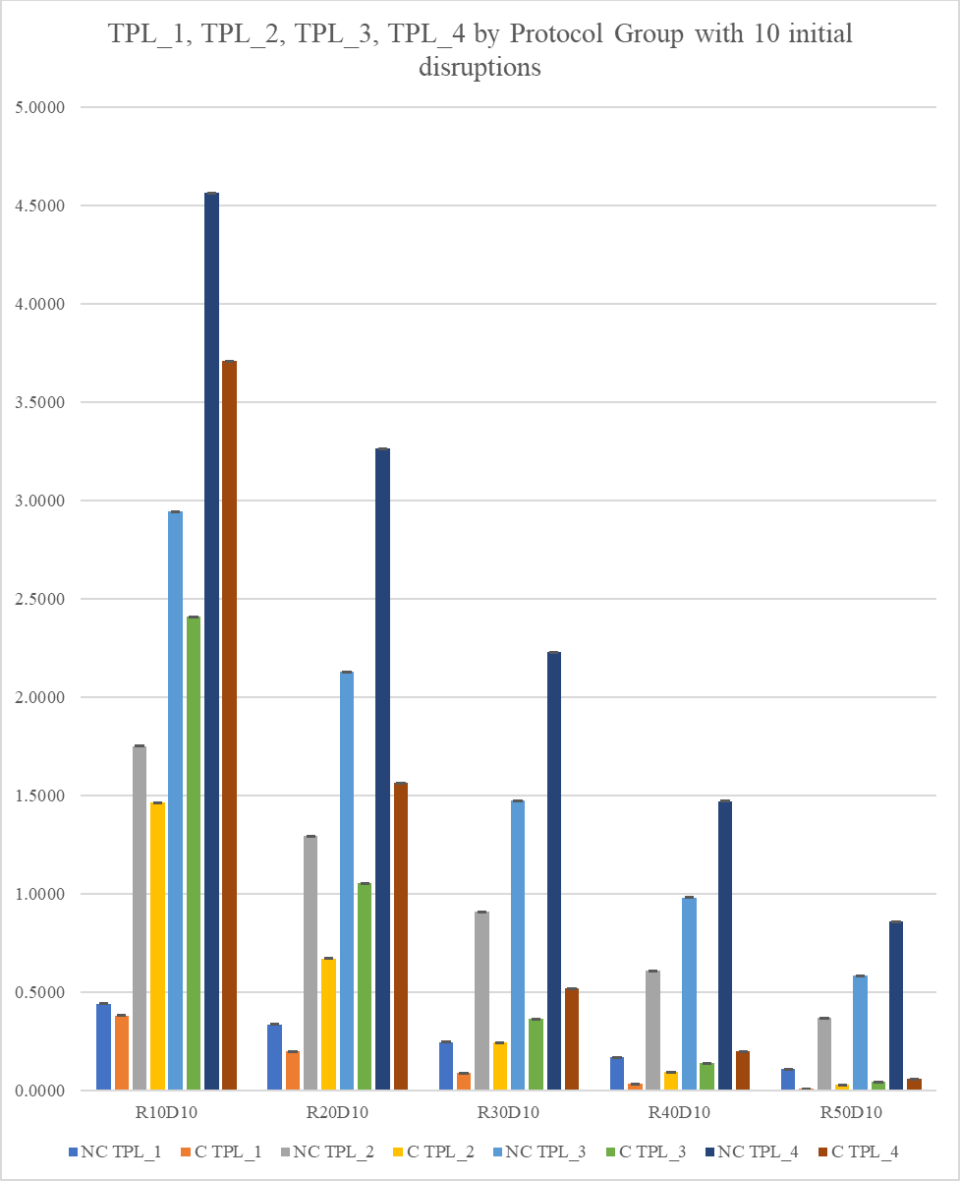

Figure 5.6. CSPD experiment results grouped by CSPD protocol groups and response/disruption scenarios with 20 initial disruptions, with 95% confidence interval bars

Table 5.10. CSPD experiment results grouped by response/disruption scenarios and CSPD protocol groups

Scenario	$\mathbf{TPL_1}$		$\mathbf{TPL_2}$		$\mathbf{TPL_3}$		$\mathbf{TPL_4}$	
	non-CLOC	**CLOC**	**non-CLOC**	**CLOC**	**non-CLOC**	**CLOC**	**non-CLOC**	**CLOC**
R10D10	0.4426	0.3830	1.7532	1.4638	2.9456	2.4105	4.5631	3.7105
R20D10	0.3380	0.2017	1.2957	0.6722	2.1290	1.0561	3.2637	1.5635
R30D10	0.2479	0.0909	0.9097	0.2454	1.4759	0.3657	2.2283	0.5208
R40D10	0.1714	0.0375	0.6098	0.0965	0.9828	0.1428	1.4723	0.2018
R50D10	0.1100	0.0116	0.3697	0.0295	0.5850	0.0438	0.8605	0.0616
R10D20	0.6423	0.5715	2.1579	1.8703	3.4477	2.9361	5.1432	4.3589
R20D20	0.5091	0.3365	1.6492	1.0136	2.5850	1.5461	3.8305	2.2452
R30D20	0.3837	0.1652	1.2043	0.4375	1.8663	0.6486	2.7360	0.9214
R40D20	0.2756	0.0715	0.8473	0.1815	1.3052	0.2677	1.8993	0.3777
R50D20	0.1875	0.0231	0.5623	0.0585	0.8619	0.0870	1.2437	0.1224
All non-CLOC vs CLOC comparisons at the same scenario and performance metric are different with $\alpha = 0.05$.								

The performance gaps (Table 5.10) between the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) and the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1$, $\boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) follow the same pattern as in the gaps provided in Table 5.8. Across all the different response/disruption scenarios, the performance gaps between the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) and the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1$, $\boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) are 11.02% to 89.43% for TPL_1, from 13.33% to 92.03% for TPL_2, from 14.84% to 92.52% TPL_3, and from 15.25% to 92.84% TPL_4. It is noted that the performance gaps (Table 5.10) increase as the number of allocations increases and the number of initial disruptions increases, which means the CLOC protocols can utilize the strategic resources more effectively.

5.4.4 Set of experiments on an enterprise's internal email network

The previous sets of experiments apply the CSPD model to four different types of numerically generated network models: GO, BA, ER, and WS. In this set of experiments, the CSPD model, analytics, and protocols are applied to a problem: The problem of preventing ($\boldsymbol{\mathcal{R}}$) unexpected and propagating computer malware ($\boldsymbol{\mathcal{D}}$) in an actual enterprise's internal email network ($\boldsymbol{\mathcal{C}}$). This set

of experiments is conducted to test the CSPD model, analytics, and protocols in an actual network model and problem. An enterprise's internal email network can be vulnerable to unexpected and propagating computer malware (the disruptions), due to the higher level of trusts between the participants (the nodes), and the higher frequency of communication (the edges) between them (Hao Zhong & Nof, 2015). Collaborative activities and file-sharing are common amongst the participants of the internal email network, and malware originating from one participant can propagate to other participants through the communication. Therefore, the established collaboration between two participants (nodes n_i, n_j) of an email network ($\boldsymbol{\mathcal{C}}$) constitutes a bilateral potential disruption propagation direction (edge $e = \{n_i, n_j\}$). In the context of the CSPD model, the disruptions ($\boldsymbol{\mathcal{D}}$) in an internal email network are the unexpected computer malware that can potentially attack any participants of the email network. The strategic resources ($\boldsymbol{\mathcal{R}}$) in this case include participant-level firewalls, secure communication protocols between participants, and/or cybersecurity awareness training for participants.

In this set of experiments, the selected enterprise's internal email network structure is condensed from the email communication of the Enron scandal, specifically from the communication between the Enron high-level employees (Cohen, 2005; Musa et al., 2018), the same network that was used in subsection 4.4.4. This network contains a total of 143 nodes (participants) and 623 undirected edges (an edge is created if any email communication was made). The node count of 143 of this email network allows reasonable comparison with the previous sets of experiments on four general random network models (GO, BA, ER, and WS). In this set of experiments, the six aforementioned CSPD protocols and ten aforementioned response/disruption scenarios (discussed in 5.4) are the experiment factors, with 100 replications for each factorial combination, resulting in a total of 6000 runs. The number of replications is selected as a balance between ensuring sufficient coverage of different disruption target combinations and ensuring reasonable total runtime of the experiments (due to the computationally expensive nature of the CSPD protocol $\boldsymbol{\mathcal{P}}_6$). The four system performance metrics $\boldsymbol{\mathcal{M}}_1$, $\boldsymbol{\mathcal{M}}_2$, $\boldsymbol{\mathcal{M}}_3$, and $\boldsymbol{\mathcal{M}}_4$ (all minimization objectives) are reported.

The comparison between CSPD protocols is presented in Figure 5.7 and Table 5.11.

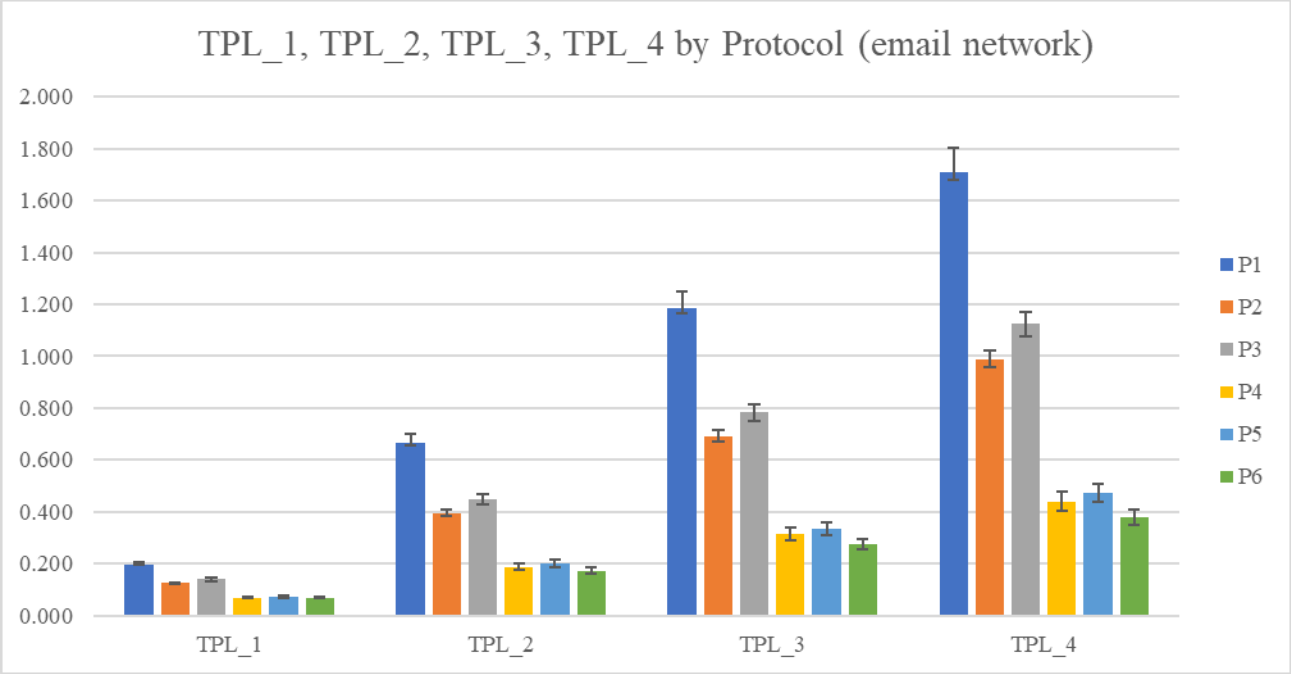

Figure 5.7. CSPD email network experiment results grouped by CSPD protocols with 95% confidence interval bars

Table 5.11. CSPD email network experiment results grouped by CSPD protocols

Protocol	$\mathcal{M}_1$: TPL_1	$\mathcal{M}_2$: TPL_2	$\mathcal{M}_3$: TPL_3	$\mathcal{M}_4$: TPL_4
$\mathcal{P}_1$	0.198*	0.667*	1.185*	1.709*
$\mathcal{P}_2$	0.124**	0.396**	0.691**	0.987**
$\mathcal{P}_3$	0.139***	0.447***	0.784***	1.124***
$\mathcal{P}_4$	0.070****	0.188****	0.314****	0.440****
$\mathcal{P}_5$	0.073****	0.200****	0.336****	0.472****
$\mathcal{P}_6$	0.069****	0.171****	0.275****	0.380****
Gap of CLOC protocols versus other protocols	41.0% - 63.0%	56.8% - 74.3%	60.2% - 76.8%	61.5% - 77.8%

*, **, ***, ****: group of confidence interval overlapping
Same group means no significant statistical difference between the different CDUD protocols of the same group.
Different groups mean significant statistical differences between any pair of CDUD protocols belonging to different groups.

The experiment results (Figure 5.7 and Table 5.11) are the system performance metrics TPL_1, TPL_2, TPL_3, and TPL_4 averaged across all network types and response/disruption scenarios. According to Table 5.11, the CSPD protocols $\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$ provide the best performance, followed by $\boldsymbol{\mathcal{P}}_2$, then by $\boldsymbol{\mathcal{P}}_3$For the purpose of protocol comparison, the group of CSPD protocols $\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$ are referred to as the non-CLOC protocols, and the CSPD protocols $\boldsymbol{\mathcal{P}}_4, \boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$ are referred to as the CLOC protocols. With respect to overall performance, the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) outperform the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) with statistical significance, ranging from 41.0% to 77.8% better performance. This gap applies to all four system performance metrics TPL_1, TPL_2, TPL_3, and TPL_4. The gaps (Table 5.11) increase as TPL becomes more long-term, with 41.0% - 63.0% for TPL_1, 56.8% - 74.3% for TPL_2, 60.2% - 76.8% for TPL_3, and 61.5% - 77.8% for TPL_4. The experiment results (Table 5.11) indicate that, in general, the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) outperform the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1, \boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) with statistical significance.

The comparisons between CSPD protocols, grouped by response/disruption scenarios, are provided in Figure 5.8, Figure 5.9, and Table 5.12.

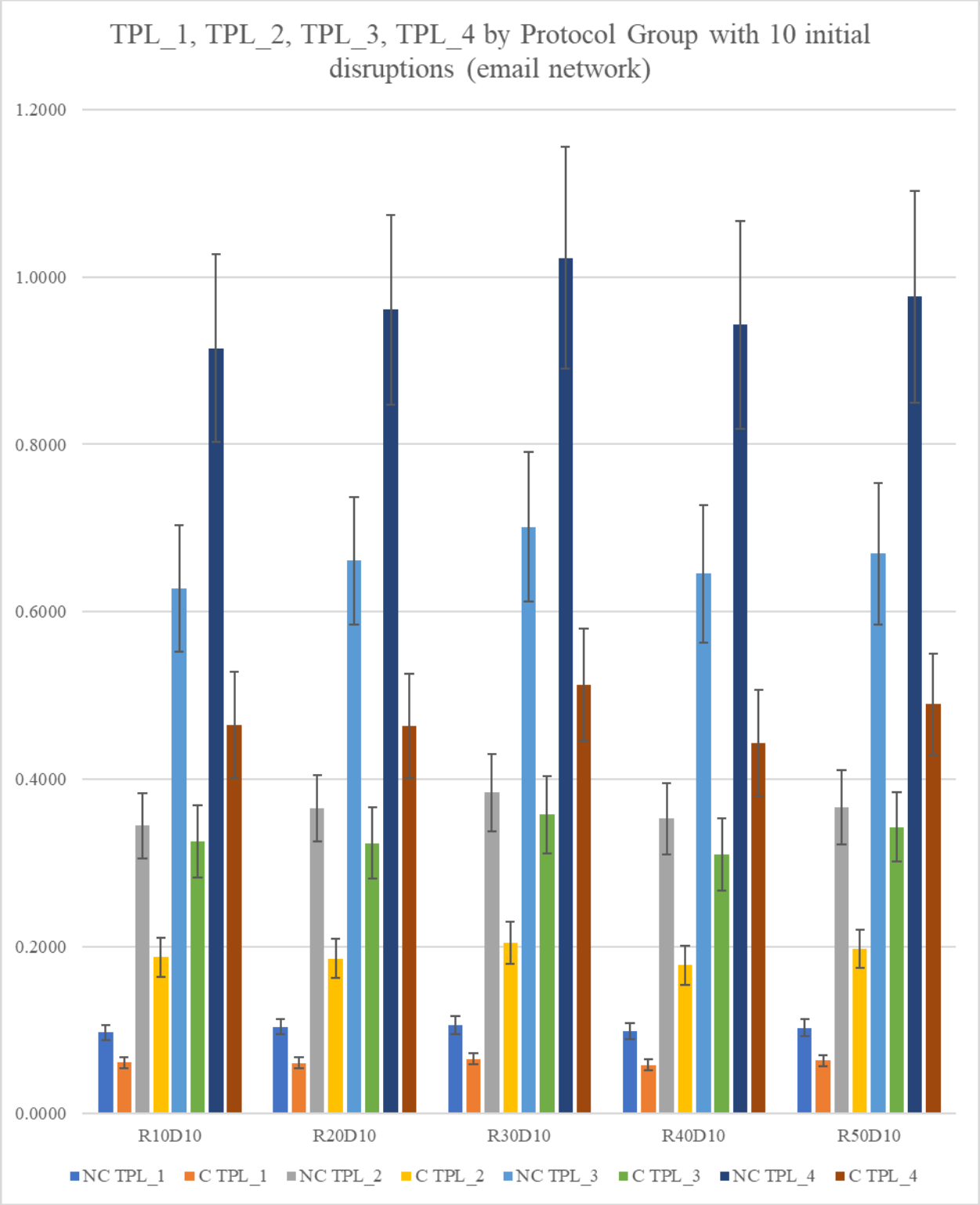

Figure 5.8. CSPD email network experiment results grouped by CSPD protocol groups and response/disruption scenarios with 10 initial disruptions, with 95% confidence interval bars

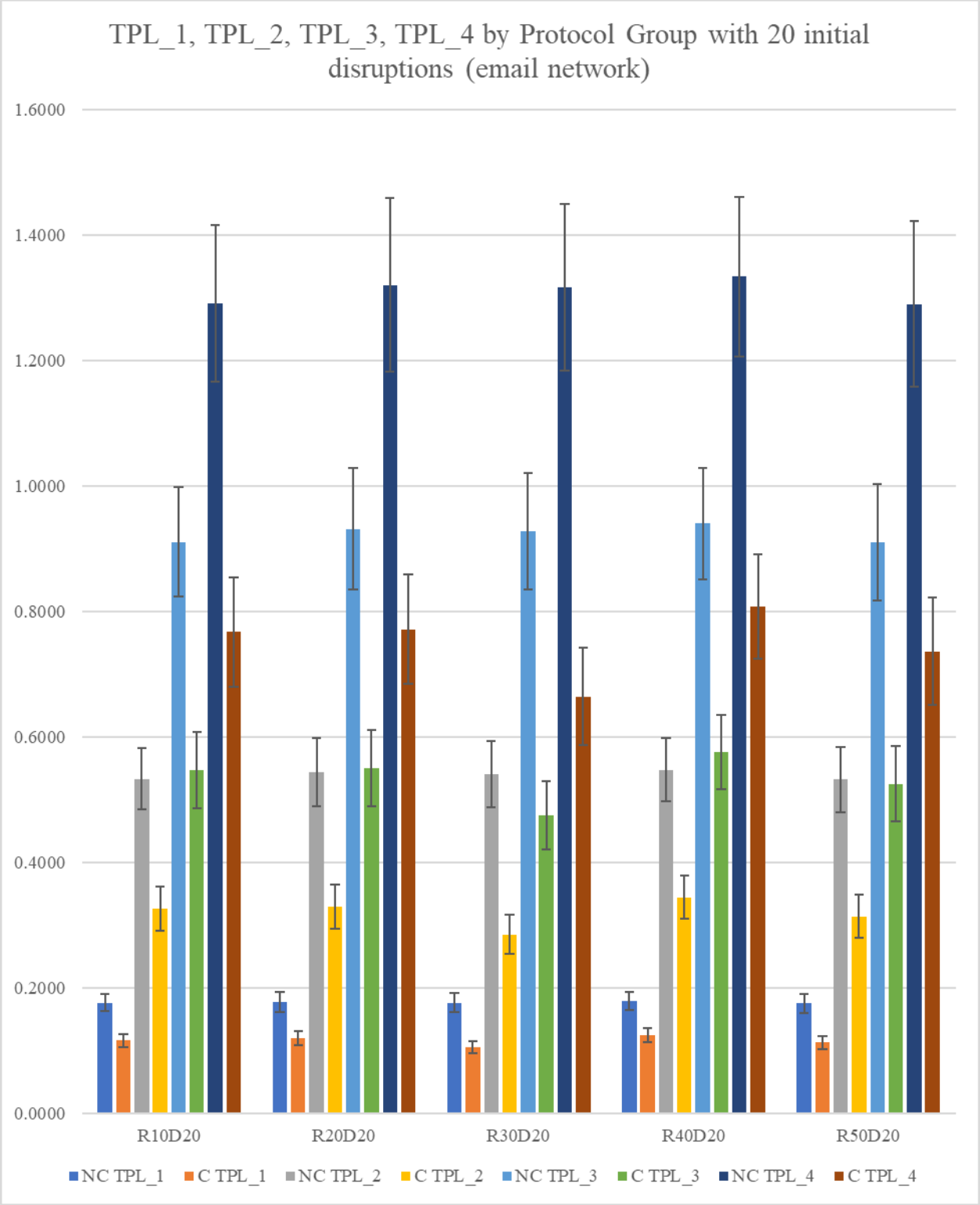

Figure 5.9. CSPD email network experiment results grouped by CSPD protocol groups and response/disruption scenarios with 10 initial disruptions, with 95% confidence interval bars

Table 5.12. CSPD email network experiment results grouped by response/disruption scenarios and CSPD protocol groups

Scenario	$\mathbf{TPL_1}$		$\mathbf{TPL_2}$		$\mathbf{TPL_3}$		$\mathbf{TPL_4}$	
	non	**CLOC**	**non**	**CLOC**	**non**	**CLOC**	**non**	**CLOC**
R10D10	0.0971	0.0611	0.3444	0.1870	0.6277	0.3253	0.9146	0.4647
R20D10	0.1040	0.0607	0.3648	0.1853	0.6609	0.3233	0.9612	0.4629
R30D10	0.1057	0.0656	0.3837	0.2042	0.7014	0.3574	1.0229	0.5122
R40D10	0.0981	0.0585	0.3527	0.1778	0.6452	0.3099	0.9430	0.4431
R50D10	0.1027	0.0633	0.3664	0.1967	0.6692	0.3426	0.9763	0.4894
R10D20	0.1770	0.1165	0.5333	0.3261	0.9113	0.5469	1.2905	0.7680
R20D20	0.1779	0.1200	0.5448	0.3302	0.9321	0.5507	1.3208	0.7718
R30D20	0.1770	0.1061	0.5411	0.2859	0.9285	0.4749	1.3174	0.6646
R40D20	0.1799	0.1256	0.5482	0.3451	0.9406	0.5762	1.3342	0.8080
R50D20	0.1756	0.1134	0.5324	0.3146	0.9108	0.5256	1.2904	0.7369
All non-CLOC vs CLOC comparisons at the same scenario and performance metric are different with $\alpha = 0.05$.								

Across all the different response/disruption scenarios, the performance gaps (Table 5.12) between the CLOC protocols ($\boldsymbol{\mathcal{P}}_4$, $\boldsymbol{\mathcal{P}}_5$, and $\boldsymbol{\mathcal{P}}_6$) and the non-CLOC protocols ($\boldsymbol{\mathcal{P}}_1$, $\boldsymbol{\mathcal{P}}_2$, and $\boldsymbol{\mathcal{P}}_3$) are 30.16% to 41.60% for TPL_1, from 37.06% to 49.60% for TPL_2, from 38.74% to 51.97% TPL_3, and from 39.44% to 53.01% TPL_4. It is noted that the performance gaps (Table 5.12) increase as the number of allocations increases and the number of initial disruptions increases, which means the CLOC protocols can utilize the strategic resources more effectively. From Figure 5.7, Table 5.11, Figure 5.8, Figure 5.9, and Table 5.12, it can be concluded that the CSPD advanced protocols provide superior prevention performance against unexpected computer malware in the enterprise's internal email network.

5.5 Concluding Remarks

In this chapter, the CSPD model is defined and formulated based on the CRDP framework, and the accompanying CSPD analytics and protocols are developed based on the guidelines of the CLOC principle. The CSPD model explores one important type of disruption: the property of being

unexpected to the response mechanisms, and one important type of response mechanisms: the static resources that cannot be changed during disruptions. In the CSPD model, the strategic and static resources can prevent and mitigate disruption propagation, but must be pre-allocated before disruptions occur. Four analytics are developed based on the CLOC principle, and six protocols are developed with the support of the four analytics. The CSPD model and protocols are validated with numerical experiments. The experiment results indicate that the advanced protocols developed based on the CLOC principle outperform the baseline and less advanced protocols by 31.1% to 56.6%, with statistical significance. The results indicate that the appropriate application of the CRDP formulation and the CLOC design and control principles can lead to significant performance improvement.

Thus, this case in CHAPTER 5 provides a partial answer to both Research Question 2 and Research Question 3 as outlined in CHAPTER 1.

CHAPTER 6. CASE 3 – COLLABORATIVE TEAMING AND COORDINATION OF DYNAMIC REPAIR AGENTS

6.1 CTCD Description

In general, preparation and configuration of the response team must be done disruptions occur (Velasquez et al., 2010; Rossi & Ahmed, 2015). In such cases, the selected response teams, with their operations protocols established, are selected to be on standby, resulting in two types of decisions to be considered: the teaming decisions and the team coordination decisions. An important disruption characteristic to consider is the recurring nature of the disruptions (Yoon et al., 2008), which is enabled by their evolutionary capability. These three problem aspects inspire the formulation of the RDP model discussed in this case study.

Following the CRDP framework, the Collaborative Teaming and Coordination of Dynamic Repair Agents (CTCD) model is formulated with the components, interactions, decision space, and system performance metrics. This model is a continuation of the work of Hao Zhong and Nof (2015); Hao Zhong (2016). The entities of the client system $\boldsymbol{\mathcal{C}}$ are represented by the nodes, each of which can represent a component or subsystem of the CPS of interest. The nodes are susceptible to disruptions in $\boldsymbol{\mathcal{D}}$ that can propagate to connected nodes that are not being responded to. The disruption information of a node is not available to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ until the simulation begins. The response mechanisms in $\boldsymbol{\mathcal{R}}$ employed for this case are dynamic repair agents that can remove disruptions and repair the disrupted nodes. Each dynamic agent can respond to any disrupted node (as opposed to static response mechanisms that have limited response capabilities and options), but can only respond to one node at a time. Two response decisions are involved, the off-line/strategic teaming decision $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ which selects the dynamic repair agents for the response team, and the on-line/tactical coordination decision $\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$ which assigns the agents to response tasks. The selected response team cannot be changed on-line. The dynamic repair agents are aware of the disruption status of all nodes, but disruptions can re-disrupt nodes that are not supervised, highlighting the recurring nature of the disruption propagation. This means effective coordination of response activities is required to ensure the resilience of the client system. With respect to $\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$, repair agents in $\boldsymbol{\mathcal{R}}$ can be deployed to any nodes in $\boldsymbol{\mathcal{C}}$, but with different repair/response times, defined by a matrix. The

repair agents are $\boldsymbol{S_{\mathcal{R}}}$ also aware of the locations of the nodes in $\boldsymbol{\mathcal{C}}$ and potential disruption propagation directions $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$. With respect to $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$, a disruption in $\boldsymbol{\mathcal{D}}$ affecting a node in $\boldsymbol{\mathcal{C}}$ can propagate to connected nodes, but with different disruption propagation time. This enables the network modeling, per the first CLOC guideline, and the resulting network is a directed and weighted network. Three important aspects of $\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$ are noted: (i) the teaming decision $\boldsymbol{S_{\mathcal{R}}}$ is not aware of the initial locations of the disruptions in $\boldsymbol{\mathcal{D}}$; (ii) the dynamic repair agents in $\boldsymbol{\mathcal{R}}$ can remove the disruption and prevent ongoing disruption propagation. The challenge of $\boldsymbol{S_{\mathcal{R}}}$ is that the targets of the initial disruptions are not known to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ ahead of time and can also propagate. The challenge of $\boldsymbol{S_{\mathcal{R}}^{t}}$ is the recurring nature of the disruption propagation, with insufficient and/or inefficient response decisions leading to more severe disruption propagation, eventually overwhelming the response capabilities. The system performance metrics $\boldsymbol{\mathcal{M}}$ investigated are: recovery fraction $\boldsymbol{\mathcal{M}_1}$, recovery time $\boldsymbol{\mathcal{M}_2}$, total performance loss $\boldsymbol{\mathcal{M}_3}$, and maximum disruption propagation fraction $\boldsymbol{\mathcal{M}_4}$. The metric recovery fraction $\boldsymbol{\mathcal{M}_1}$ indicates the probability of full recovery from disruption propagation. The metric recovery time $\boldsymbol{\mathcal{M}_2}$ indicates the total time taken to fully recover from the disruptions. Both $\boldsymbol{\mathcal{M}_1}$ and $\boldsymbol{\mathcal{M}_2}$ are important resilience measures for the client system $\boldsymbol{\mathcal{C}}$, because the critical infrastructure and CPS should be fully recovered as reliably as possible and as soon as possible. The metric total performance loss $\boldsymbol{\mathcal{M}_3}$ measures the total over-time performance loss of $\boldsymbol{\mathcal{C}}$ due to disruptions. $\boldsymbol{\mathcal{M}_3}$ is relevant when the client system $\boldsymbol{\mathcal{C}}$ is still expected to be operational under disruption, such as in the case of computer networks. The metric maximum performance loss $\boldsymbol{\mathcal{M}_4}$ measures the highest level of performance loss to ever occur, and is important to consider because certain disruption types incur long-term or permanent damages that cannot be recovered from, such as loss of sensitive information in computer networks.

A summary of the CTCD model formulation is provided in Table 6.1.

Table 6.1. Summary of CTCD description

CRDP Formulation Category	Item	Details
CTCD Components	$\boldsymbol{\mathcal{C}}$: client system	Nodes representing components and subsystems of the cyber-physical systems.
	$\boldsymbol{\mathcal{R}}$: response mechanisms	A response team consisting of dynamic repair agents. The team can be selected off-line.
	$\boldsymbol{\mathcal{D}}$: disruption propagation	Initial disruptions are not known to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$ ahead of time. Disruptions can propagate if not removed.
CTCD Interactions	$\boldsymbol{\mathcal{C}}$&$\boldsymbol{\mathcal{R}}$: client-response interaction	Repair agents can respond to any node in $\boldsymbol{\mathcal{C}}$, with different response times defined by a matrix.
	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$: disruption-client interaction	A disruption affecting a node, if not responded to, can propagate to the node's succeeding nodes, with varying times (the resulting $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ can be modeled as a network with directed and weighted edges).
	$\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$: response-disruption interaction	The teaming decisions and the coordination decisions are aware of potential disruption propagation direction and time. The teaming decisions and the coordination decisions are not aware of initial disruptions ahead of time. The coordination decisions are aware of ongoing disruptions.
CTCD Decision Space	$\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}$: teaming decision	Teaming decision to select the dynamic repair agents for the response team off-line.
	$\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$: coordination decision	Coordination decisions to assign agents to repair tasks during real-time.
CTCD System Performance Metrics	$\boldsymbol{\mathcal{M}}_1$: recovery fraction	The probability of fully recovering the client system from disruptions within a given time.
	$\boldsymbol{\mathcal{M}}_2$: recovery time	Total time taken to fully remove the disruptions.
	$\boldsymbol{\mathcal{M}}_3$: total performance loss	Total over-time disruptions affecting the client system.
	$\boldsymbol{\mathcal{M}}_4$: maximum disruption propagation fraction	Maximum number of disruptions affecting the client system at any point in time.

6.2 CTCD Formulation

Based on the CTCD model description, the CRDP formulation of the CTCD model is as follows. The CTCD model is simulated using the one variation of the TIE/CRDP software presented in APPENDIX A. The CTCD simulation is a discrete-event simulation. The entities and attributes are given in Table 6.2, the system performance metrics are given in Table 6.3, the discrete events are given in Table 6.4.

Table 6.2. Entities and attributes of the CTCD model

Type	Entity/Attribute and Explanation	CRDP domain
Input	$\boldsymbol{\mathcal{C}}$: $\mathrm{NL} = \{n_0, n_1, \dots\}$ Set of nodes, with each node $n \in \mathrm{NL}$ representing a component or subsystem of the client system.	$\boldsymbol{\mathcal{C}}$
Input	$\boldsymbol{\mathcal{R}}$: $\mathrm{TL} = \{\mathrm{AL}_0, \mathrm{AL}_1, \dots\}$ Set of response teams, with each team consisting of different dynamic agents that can respond to disruptions affecting $\boldsymbol{\mathcal{C}}$.	$\boldsymbol{\mathcal{R}}$
Input	$\boldsymbol{\mathcal{R}}$: $\mathrm{AL} = \{a_0, a_1, \dots\}$ A team AL is a set of dynamic agents, with each agent $a \in \mathrm{AL}$ having the capability to dynamically respond to disruptions affecting $\boldsymbol{\mathcal{C}}$.	$\boldsymbol{\mathcal{R}}$
Input	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$: $\mathrm{RRM}(a, n) \in \mathbb{R}_{\geq 0}$ The response requirement matrix denoting the total time taken for a certain agent a to respond to a certain node n. This matrix has 2 dimensions of $\sum_{\mathrm{AL}}^{\mathrm{TL}}\lvert\mathrm{AL}\rvert$ rows and $\lvert\mathrm{NL}\rvert$ columns.	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
Input	$\boldsymbol{\mathcal{D}}$: $\mathrm{DL} = \{d_0, d_1, \dots\}$ Set of disruptions, with each disruption $d \in \mathrm{DL}$ having the capability to disrupt and re-disrupt a nod. If left unresponded to, a disruption can propagate to other nodes.	$\boldsymbol{\mathcal{D}}$
Input	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$: $\mathrm{EL} = \{e_0, e_1, \dots\}$ The set of directed edges, with each directed edge $e = (n_i, n_j)$ representing a potential disruption propagation direction from node n_i to node n_j.	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
The following attributes are defined for each node $n \in$ NL		
Dynamic	$\mathrm{NDS}(n, t) \in \{0,1\}$ Node n's disruption status at time t, with value 0 denoting that node n is not disrupted. If $\mathrm{NDS}(n, t) = 1$, the node is disrupted and can propagate disruptions to its succeeding nodes. Default value of 0.	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$

Table 6.2. continued

Type	Entity/Attribute and Explanation	CRDP domain
Dynamic	$\mathrm{NLDT}(n,t) \in \mathbb{R}_{\geq 0}$ Node n's latest disrupted time, which is used to keep track of the node's previous disrupted time. Default value is 0.	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
Derived	$\mathrm{NPEL}(n) \subset \mathrm{EL}$ Node n's set of incoming/preceding edges, which consists of all edges pointing towards n. $\mathrm{NPEL}(n) = \{e = (n_i, n_j) \in \mathrm{EL}: n_j \equiv n\}$	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
Derived	$\mathrm{NSEL}(n) \subset \mathrm{EL}$ Node n's set of outgoing/succeeding edges, which consists of all edges pointing from n. $\mathrm{NSEL}(n) = \{e = (n_i, n_j) \in \mathrm{EL}: n_i \equiv n\}$	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
The following attributes are defined for each edge $\boldsymbol{e = (n_i, n_j) \in}$ EL		
Input	$\mathrm{EDPT}(e) \in \mathbb{R}_{\geq 0}$ Edge e's disruption propagation time, and is also the weight of the edge. Suppose node n_i is disrupted at time t, then at time $t + \mathrm{EDPT}(e)$, node n_j will become disrupted if both node n_i and node n_j have not been responded to by an agent.	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
Dynamic	$\mathrm{EDPS}(e,t) \in \{0,1\}$ Edge e 's disruption propagation status, mainly used for simulation. $\mathrm{EDPS}(e,t) = 1$ means the disruption propagation along edge e will occur as planned. $\mathrm{EDPS}(e,t) = 0$ means the disruption propagation is halted, due to the intervention of a dynamic agent.	$\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$
The following attributes are defined for each agent $\boldsymbol{a \in}$ AL		
Decision	$\mathrm{SRT} \in \mathrm{TL}$ Selected response team to be on standby. This decision cannot be changed during real-time. Only agents $a \in \mathrm{AL} \equiv \mathrm{SRT}$ can respond to disruptions.	$\boldsymbol{\mathcal{S}_{\mathcal{R}}}$
Decision	$\mathrm{NAA}(n,t) \in \mathrm{AL}$ Node n's assigned agent at time t. Default value is $\emptyset$.	$\boldsymbol{\mathcal{S}_{\mathcal{R}}^{t}}$
Dynamic	$\mathrm{ABS}(a,t) \in \{0,1\}$ Agent a's busy status at time t. $\mathrm{ABS}(a,t) = 0$ means the agent is idle, and $\mathrm{ABS}(a,t) = 1$ means the agent is busy (currently responding to a disruption).	$\boldsymbol{\mathcal{R}}$

The nodes in the client system $\boldsymbol{\mathcal{C}}$ are represented by the set of nodes NL. The selected repair agents in $\boldsymbol{\mathcal{R}}$ to be on standby are represented by the set of agents SRT $\equiv$ AL, which must be selected from the set of teams TL through $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$. The disruptions in $\boldsymbol{\mathcal{D}}$ are represented by the attributes $\mathrm{NDS}(n,t)$, and $\mathrm{NDS}(n,0) = 1$ is caused the targeting of $d \in$ DL. The allocations of response activities to the nodes in NL are represented by $\mathrm{NAA}(n,t)$. Different agents can have different response times for different nodes, defined by the response requirement matrix RRM. A busy agent is denoted with $\mathrm{ABS}(a,t) = 1$, and cannot be preempted for other tasks. Per the first CLOC guideline, the disruption propagation directions are represented by the set of directed and weighted edges EL, which is known to $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{D}}$. The weights of the edges are represented by $\mathrm{EDPT}(e)$. The set of preceding edges $\mathrm{NPEL}(n)$ and the set of succeeding edges $\mathrm{NSEL}(n)$ are also defined for each node.

Following the specification of Table 6.1, the CDUD system performance metrics are given in Table 6.3.

Table 6.3. System performance metrics of the CTCD model

System Performance Metric	CRDP domain
$$\mathrm{PL}(t) = \sum_{n}^{\mathrm{NL}} \frac{\mathrm{NDS}(n,t)}{\lvert \mathrm{NL} \rvert}$$ Performance loss at time t, which denotes the fraction of the client system disrupted at time t. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, $\mathrm{PL}(t)$ is to be minimized.	$\boldsymbol{\mathcal{M}}$
$$\mathrm{RF} \in [0,1]$$ Recovery fraction, which is the fraction of the simulation replications with successful complete recovery of the client system from disruption propagation. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, RF is to be maximized.	$\boldsymbol{\mathcal{M}}_1$
$$\mathrm{RT}$$ Recovery time, which is the time that the client system is fully recovered from disruptions, or infinity if the simulation ends and there are disruptions remaining. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, RT is to be minimized.	$\boldsymbol{\mathcal{M}}_2$
$$\mathrm{TPL} = \int_{t=0}^{t=t_{\max}} \mathrm{PL}(t)\,dt$$ Total performance loss, which is the total accumulative damage caused by the disruption propagation on the client system. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, TPL is to be minimized.	$\boldsymbol{\mathcal{M}}_3$
$$\mathrm{MDPF} = \max_{\mathrm{t}} \mathrm{PL}(t)$$ Maximum disruption propagation fraction, which is the highest performance loss suffered by the client system at any point in time. From the perspective of $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}}$, MDPF is to be minimized.	$\boldsymbol{\mathcal{M}}_4$

Table 6.4. Discrete events in CTCD model

Event	Pseudocode
$\mathrm{NDP}\left(t, e = \left(n_i, n_j\right)\right)$ Node disruption propagates Corresponds to $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$, $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$	if $\left(\mathrm{EDPS}(e,t) = 1 \text{ and } \mathrm{NDS}\left(n_j, t\right) = 0\right)$ $\mathrm{NDS}\left(n_j, t\right) \leftarrow 1$ $\mathrm{NDLT}\left(n_j, t\right) \leftarrow t$ foreach $\left(e_{nj} = \left(n_{ji}, n_{jj}\right) \in \mathrm{NSEL}\left(n_j\right) \mid \mathrm{EDPS}\left(e_{nj}, t\right) = 0 \text{ and } \mathrm{NAA}(n_{jj}, t = \emptyset\right)$ $\mathrm{EDPS}\left(e_{nj}, t\right) \leftarrow 1$ Schedule event $\mathrm{NDP}\left(t + \mathrm{EDPT}\left(e_{nj}\right), e_{nj}\right)$
$\mathrm{AERN}(t, a, n)$ Agent ends responding node Corresponds to $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$, $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$, and $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$	$\mathrm{NDS}(n, t) \leftarrow 0$ $\mathrm{NAA}(n, t) \leftarrow \emptyset$ $\mathrm{ABS}(a, t) \leftarrow 0$ For each $\left(e = \left(n_i, n_j\right) \in \mathrm{NPEL}(n) \mid \mathrm{NDS}(n_i, t) = 1 \text{ and } \mathrm{NAA}(n_i, t) = \emptyset\right)$ $\mathrm{EDPS}(e, t) \leftarrow 1$ Schedule event $\mathrm{NDP}(t + \mathrm{EDPT}(e), e)$

The event $\mathrm{NDP}(t, e)$ propagates an ongoing disruption along an edge $e = \left(n_i, n_j\right)$, from its start node n_i to its end node n_j, at time t, causing the node n_j to be disrupted, or $\mathrm{NDS}\left(n_j, t\right) \leftarrow 1$. An event $\mathrm{NDP}(t, e)$ was caused at time $t - \mathrm{EDPT}(e)$ when node n_i was disrupted, or $\mathrm{NDS}(n_i, t) = 1$, which set $\mathrm{EDPS}(e, t) \leftarrow 1$ and scheduled the event $\mathrm{NDP}(n_i, t)$. If $\exists t_o \in [t - \mathrm{EDPT}(e), t] : \mathrm{EDPS}(e, t_o) = 0$, which is caused by an agent responding to either n_i or n_j, this event $\mathrm{NDP}(t, e)$ is canceled because $\mathrm{EDPS}(e, t) = 0$. An example is shown in Figure 6.1.

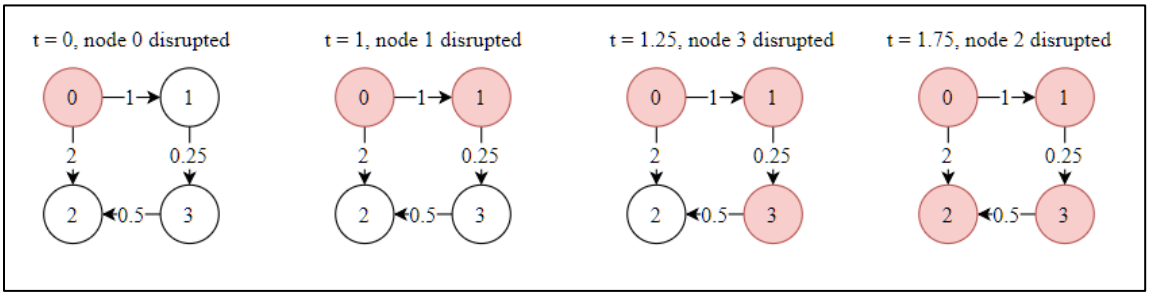

Figure 6.1. CTCD disruption propagation example

The event $\mathrm{AERN}(t, a, n)$ finishes the response activity of the agent a to node n at time t, removing the disruption affecting node n by setting $\mathrm{NDS}(n, t) \leftarrow 0$. This event also releases the agent from the task, setting $\mathrm{NAA}(n) \leftarrow \emptyset$ and $\mathrm{ABS}(n) \leftarrow 0$, allowing the agent to be responded to. If one of node n's preceding is currently disrupted and not being responded to, the disruption propagation process is restarted. An example is shown in Figure 6.2.

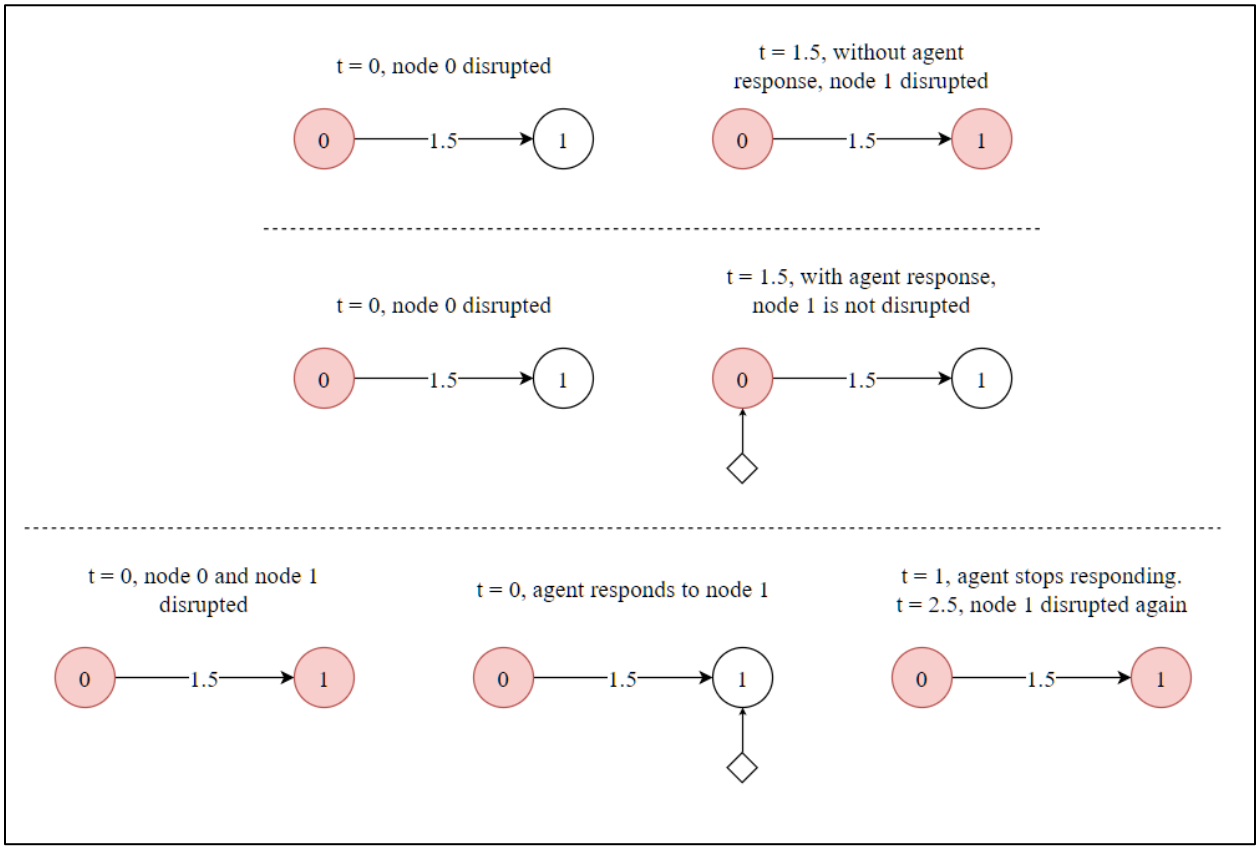

Figure 6.2. CTCD response mechanisms example

Table 6.5. Simulation pseudocode of the CTCD model

Step	Pseudocode	CRDP domain
Step 1	$t \leftarrow 0$, Initialize NL, TL, RRM, DL, EL	Simulation
Step 2	Decide SRT AL ← SRT	$\boldsymbol{\mathcal{S}_{\mathcal{R}}}$
Step 3	Foreach $d \in$ DL, Randomly select $n_d \in$ NL without overlap $\mathrm{NDS}(n_d, 0) \leftarrow 1$ Foreach $e_{nj} \in \mathrm{NSEL}(n)$ $\mathrm{EDPS}(e_{nj}, 0) \leftarrow 1$ Schedule event $\mathrm{NDP}(t + \mathrm{EDPT}(e_{nj}), e_{nj})$ Next d	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$
Step 4	while ($t < t_{\max}$ and $\sum_{n \in NL} \mathrm{NDS}(n,t) > 0$)	$\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$
Step 4.1	Run all events NDP at time t Run all events AERN at time t	Simulation
Step 4.2	Decide for all $a \in$ AL | $\mathrm{ABS}(a,t) = 0$ Select $n \in$ NL | $\mathrm{NDS}(n,t) = 1$ and $\mathrm{NAA}(n,t) = \emptyset$	$\boldsymbol{\mathcal{S}}_{\mathcal{R}}^{t}$
Step 4.2.1	$\mathrm{NAA}(n,t) \leftarrow a$; $\mathrm{ABS}(a,t) \leftarrow 1$	$\boldsymbol{\mathcal{S}}_{\mathcal{R}}^{t}$
Step 4.2.2	Foreach $(e \in \mathrm{NPEL}(n))$ $\mathrm{EDPS}(e,t) \leftarrow 0$ Foreach $(e \in \mathrm{NSEL}(n))$ $\mathrm{EDPS}(e,t) \leftarrow 0$ Schedule event $\mathrm{AERN}(t + \mathrm{RRM}(a,n), a, n)$	$\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}, \boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$
Step 4.3	$\mathrm{PL}(t) \leftarrow \sum_{n}^{\mathrm{NL}} \frac{\mathrm{NDS}(n,t)}{\lvert\mathrm{NL}\rvert}$ $\mathrm{TPL} \leftarrow \mathrm{TPL} + \mathrm{PL}(t) * (t - t_{\mathrm{last}})/\lvert\mathrm{NL}\rvert$ $\mathrm{MDPF} \leftarrow \max(\mathrm{MDPF}, \mathrm{PL})$;	$\boldsymbol{\mathcal{M}}$
Step 4.4	Reorder event calendar based on time and event order $t_{\mathrm{last}} \leftarrow t$; $t \leftarrow$ next minimum t on event calendar;	Simulation
Step 5	Compute $\boldsymbol{\mathcal{M}}$ if $\mathrm{PL}(t) = 0, \mathrm{RT} \leftarrow t$, else, $\mathrm{RT} \leftarrow +\infty$ if $\mathrm{PL}(t) = 0, \mathrm{RF} \leftarrow \mathrm{RF} + 1$, else, $\mathrm{RF} \leftarrow \mathrm{RF} + 0$	$\boldsymbol{\mathcal{M}}$

In Table 6.5, Step 1 initializes the input of the simulation, which includes the set of nodes NL, the set of teams TL, the response requirement matrix RRM, the set of disruption DL, and the set of directed and weighted edges EL.

Step 2 selects the response team SRT to be on standby, setting AL ← SRT. These decisions can be supported by analytics and protocols, which are discussed below in the following section.

Step 3 selects the nodes n_d to be disrupted initially and schedules the future disruption propagation events for those nodes. In this case, the selected distribution is random uniform.

Step 4 starts the discrete-event simulation. An event calendar is maintained, and the simulation jumps to the next time t on the event calendar. Step 4.1 runs all the scheduled events at time t.

Step 4.2 starts the coordination processes of the dynamic repair agents. All agents a that are not busy, or $\mathrm{ABS}(a,t) = 0$ are considered. Step 4.2.1 decides and actuates the coordination. An agent is assigned to respond to a disrupted node that has not been responded to. Step 4.2.2 prevents all ongoing disruption propagation(s) coming to and from the selected node, and the event $\mathrm{AERN}(t + \mathrm{RRM}(a,n), a, n)$ is scheduled.

Step 4.3 computes the performance loss $\mathrm{PL}(t)$ and update the metrics TPL and MDPF.

Step 4.4. reorders the event calendar, which could have been updated due to new events, then sets the time t to the earliest event on the calendar. Then, the simulation returns to Step 4.1 and repeats until the simulation length is reached or all disruptions have been responded to. The maximum time limit is necessary because it is possible for $\mathrm{PL}(t)$ to never reach 0 due to insufficient and/or inefficient response activities.

Step 5 finalizes the system performance metrics recovery time RT and recovery fraction RF.

6.3 CTCD Analytics and Protocols

In this subsection, the CSPD analytics and protocols are developed based on the CLOC principle to support the decision-making of $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$. The analysis of the CTCD model and decision space is as follows.

6.3.1 CTCD teaming decisions

Node n's set of incoming/preceding neighboring nodes $\mathrm{NPNL}(n) \subset NL$ is formally defined as

$$\mathrm{NPNL}(n) = \{n_i \neq n \in \mathrm{NL}: \exists e = (n_i, n) \in \mathrm{EL}\} \tag{30}$$

Node n's set of outgoing/preceding neighboring nodes $\mathrm{NSNL}(n) \subset \mathrm{NL}$ is formally defined as

$$\mathrm{NSNL}(n) = \{n_j \neq n \in \mathrm{NL}: \exists e = (n, n_j) \in \mathrm{EL}\} \tag{31}$$

Suppose a $(\mathrm{NL}, \mathrm{EL})$ is given with all determined values for all $\mathrm{EDPT}(e)$, and a node n_d is selected as the only node disrupted initially, meaning at $t = 0, \mathrm{NDS}(n_d, 0) = 1$ and $\mathrm{NDS}(n, 0) = 0, \forall n \in \mathrm{NL} - \{n_d\}$, with no response agents available, meaning $\mathrm{AL} = \emptyset$. With $e = (n_i, n_j)$, it is observed that

$$\mathrm{NDS}\left(n_j, \mathrm{EDPT}(e)\right) = 1, \forall e \in \mathrm{NSEL}(n_d) \tag{32}$$

This is because the disrupted node n_d is the only cause of disruption. It is noted that, however, certain nodes $n_j \in \mathrm{NONL}(n_d)$ may be disrupted earlier than the expected value of $\mathrm{EDPT}(e)$ if a shorter disruption propagation path exists from n_d to that node n_j. From the aforementioned observation on $\mathrm{NSNL}(n)$, the first CTCD analytic $\boldsymbol{\mathcal{A}}_1$: neighboring disruption analytic $\mathrm{NNDA}(n) \in \mathbb{R}_{\geq 0}$ is defined as

$$\mathrm{NNDA}(n) = \sum_{n_j}^{\mathrm{NSNL}(n)} 1 / \min_{e \equiv (n, n_j)} \{\mathrm{EDPT}(e)\} \tag{33}$$

The $\mathrm{NNDA}(n)$ analytic provides information regarding the local-level impact of a disruption affecting node n. The value of $\mathrm{NNDA}(n)$ increases with a higher number of outgoing edges, or $|\mathrm{NOEL}|$, and with lower weight for each edge $e \in \mathrm{NSEL}(e)$. The formula also addresses the case where multiple edges exist from n to n_j, and $\mathrm{NNDA}(n)$ only considers the shortest edge.

Compared to network-level analytics, $\mathrm{NNDA}(n)$ is more limited in terms of information provided, but requires less computational power to calculate, which is complexity $O(|\mathrm{EL}|)$.

An analytic more advanced than $\mathrm{NNDA}(n)$ would consider the network-level aspect of disruption propagation. Based on the observation that disruptions propagate from one node n_i (if n_i is the only disrupted node initially) to another node n_j through the shortest path from n_i to n_j, the shortest-path matrix $\mathrm{DIST} = (d_{i,j}) \in \mathbb{R}_{\geq 0}^{|\mathrm{NL}|\times|\mathrm{NL}|}$ can be computed to assist with the calculation of network-level analytics. The matrix DIST can be computed efficiently using the Floyd-Warshall algorithm (Hao Zhong & Nof, 2015) with complexity $O(|\mathrm{NL}|^3)$. Each entry is defined as $\mathrm{DIST}(n_i, n_j) \in \mathbb{R}_{\geq 0}$ representing the shortest-path distance from node n_i to node n_j, with the edge directions applied and edge weights represented by $\mathrm{EDPT}(e)$. If no such path exists, $\mathrm{DIST}(n_i, n_j) = \mathrm{null}$, and $1/\mathrm{DIST}(n_i, n_j) = 0$. Using the shortest-path distance matrix, the second CTCD analytic $\boldsymbol{\mathcal{A}}_2$, the harmonic centrality analytic $\mathrm{NHCA}(n) \in \mathbb{R}_{\geq 0}$ is defined as

$$\mathrm{NHCA}(n) = \sum_{n_j \neq n}^{\mathrm{NL}} 1/\mathrm{DIST}(n, n_j) \tag{34}$$

The $\mathrm{NHCA}(n)$ analytic provides information regarding the network-level impact of a disruption affecting node n. The value of $\mathrm{NHCA}(n)$ increases if node n is closer to more nodes. The formula of $\mathrm{NHCA}(n)$ also addresses the case where multiple edges exist between one pair of nodes in that only the shortest path is considered in the calculation. Compared to the local-level analytic $\mathrm{NNDA}(n)$, $\mathrm{NHCA}(n)$ provides more information regarding disruption propagation risk, but requires more computational power to calculate.

The main limitation of both $\mathrm{NNDA}(n)$ and $\mathrm{NHCA}(n)$ is that their disruption propagation analyses do not consider the performance metrics used to evaluate a problem instance. While $\mathrm{NHCA}(n)$ can provide a relative ranking between nodes, the proportional differences in values of $\mathrm{NHCA}(n)$ between nodes may not reflect the actual differences with respect to the performance metrics TPL and MDPF.

To address the total performance loss metric TPL, the CTCD third analytic $\boldsymbol{\mathcal{A}_3}$, the rate of disruption propagation analytic $\mathrm{NRDP}(n) \in \mathbb{R}_{\geq 0}$ is defined as

$$\mathrm{NRDP}(n) = \frac{\int_{t=0}^{t=\max\limits_{n_i,n_j \in NL} \mathrm{SPD}(n_i,n_j)} \left|\{n_j \in \mathrm{NL}: \mathrm{DIST}(n,n_j) \leq t\}\right| dt}{\max\limits_{n_i,n_j \in \mathrm{NL}} \mathrm{DIST}(n_i,n_j)} \tag{35}$$

The analytic $\mathrm{NRDP}(n)$ aggregates the rate of increasing total performance loss of the CPS if node n is the sole initially disrupted node with no response agents present. To address the maximum disruption propagation fraction metric MDPF, the fourth CTCD analytic $\boldsymbol{\mathcal{A}_4}$, the maximum disruption propagation analytic $\mathrm{NMDP}(n) \in \mathbb{R}_{\geq 0}$ is defined as

$$\mathrm{NMDP}(n) = \frac{\left|\{n_j \in \mathrm{NL}: \mathrm{DIST}(n,n_j) \neq \mathrm{null}\}\right|}{\max\limits_{n_i,n_j \in \mathrm{NL}} \mathrm{DIST}(n_i,n_j)} \tag{36}$$

The analytic $\mathrm{NMDP}(n)$ considers the maximum damage a disruption affecting node n can cause. Both $\mathrm{NRDP}(n)$ and $\mathrm{NMDP}(n)$ overcome the limitation of $\mathrm{NHCA}(n)$ that tends to give a higher weight to nearby nodes n_j with extremely close proximity to n due to the $1/\mathrm{DIST}(n,n_j)$ formula. A simple 3-node example is provided in Figure 6.3.

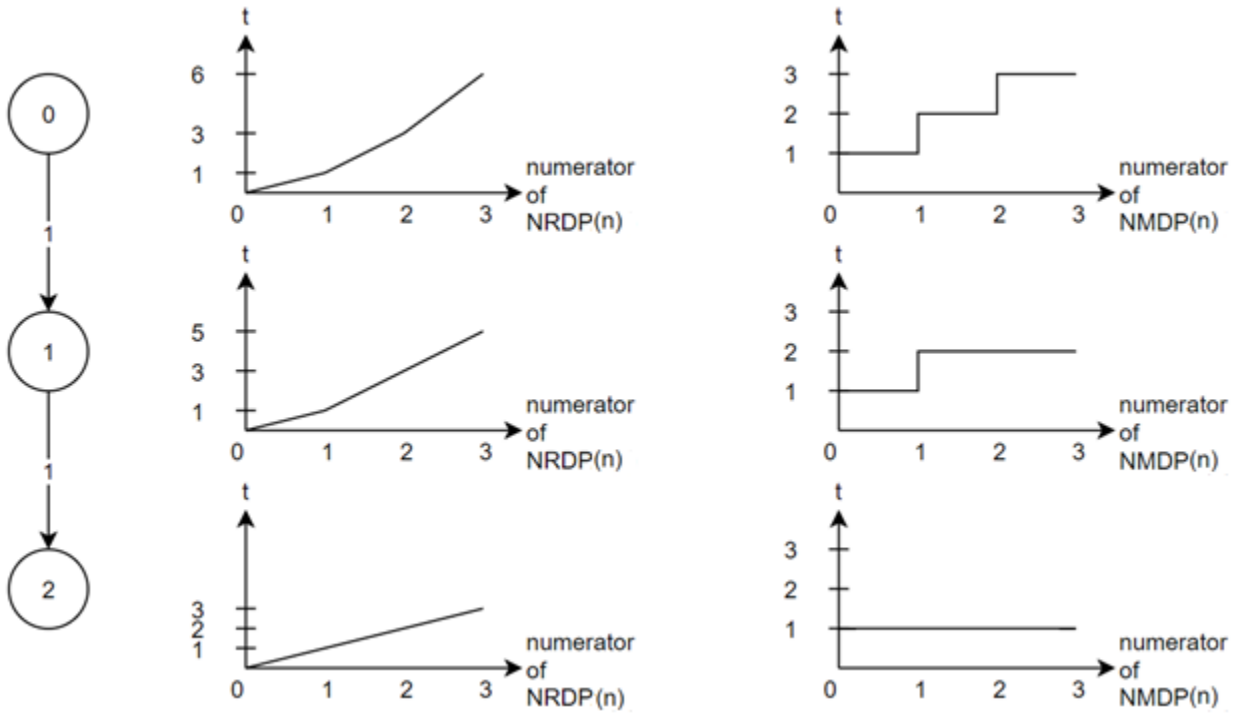

Figure 6.3. Example of NRDP(n) and NMDP(n)

Node n's strategic value $\mathrm{NSV}(n) \in \mathbb{R}$ is selected from $\mathrm{NNDA}(n)$, $\mathrm{NHCA}(n)$, $\mathrm{NRDP}(n)$, $\mathrm{NMDP}(n)$ or a function combining these four indices. This decision is left open to individual cases and scenarios, depending on the available information and computational resources.

The next step is to evaluate the strategic compatibility of each agent team. A team of agents is defined as $\mathrm{AL} = \{a_0, a_1, \ldots\}$ with each agent a_i capable of responding to a disruption affecting node n_j after a period of time $\mathrm{RRM}(i,j)$, which takes in two integer arguments. An alternative notation for $\mathrm{RRM}(i,j)$ is $\mathrm{RRM}(a_i, n_j)$, which takes in the first argument as an agent, and the second argument as a node.

Agent a's estimated effectiveness index $\mathrm{AEI}(a)$ towards the client system is defined as

$$\mathrm{AEI}(a) = \sum_{n}^{\mathrm{NL}} \frac{\mathrm{NSV}(n)}{\mathrm{RRM}(a,n) * \sum_{n_o}^{\mathrm{NL}} \mathrm{NSV}(n_o)} \tag{37}$$

Aggregating all agents of a team, the fifth CTCD analytic, $\boldsymbol{\mathcal{A}_5}$, the team AL's strategic compatibility index $\mathrm{TSCI(AL)} \in \mathbb{R}_{\geq 0}$ is defined as

$$\mathrm{TSCI(AL)} = \sum_{a}^{\mathrm{AL}} \frac{\mathrm{AEI}(a)}{|\mathrm{AL}|} = \sum_{a}^{\mathrm{AL}} \sum_{n}^{\mathrm{NL}} \frac{\mathrm{NSV}(n)}{|\mathrm{AL}| * \mathrm{RRM}(a,n) * \sum_{n_o}^{\mathrm{NL}} \mathrm{NSV}(n_o)} \tag{38}$$

The index $\mathrm{TSCI(AL)}$ estimates the total effectiveness of a team of response agents given a certain selected method of deciding $\mathrm{NSV}(n)$, the team's RRM, and the set of nodes NL.

The protocol to support the $\boldsymbol{\mathcal{S}_{\mathcal{R}}}$ teaming decision can opt to select the appropriate response team AL or a limited set of AL, based on the evaluation of $\mathrm{TSCI(AL)}$. Higher values of $\mathrm{TSCI(AL)}$ would indicate higher strategic compatibilities. For the purpose of comparison, four protocols $\boldsymbol{\mathcal{P}_1}$ to $\boldsymbol{\mathcal{P}_4}$ are defined.

$\boldsymbol{\mathcal{P}_1}$: Random team selection protocol, which selects a response team randomly. This is the baseline protocol.

$\boldsymbol{\mathcal{P}_2}$: Low-compatibility team selection protocol, which selects a response team with a low value of strategic compatibility $\mathrm{TSCI(AL)}$, around the minimum value.

$\boldsymbol{\mathcal{P}_3}$: Medium-compatibility team selection protocol, which selects a response team with a medium value of strategic compatibility $\mathrm{TSCI(AL)}$, around the average or median values.

$\boldsymbol{\mathcal{P}_4}$: High-compatibility team selection protocol, which selects a response team with a high value of strategic compatibility $\mathrm{TSCI(AL)}$, around the maximum value.

The CTCD analytics and protocols for teaming decisions are summarized in Table 6.6.

Table 6.6. Summary of the CTCD teaming decisions analytics and protocols

Analytic/ Protocol	Name	CLOC guideline
$\boldsymbol{\mathcal{A}}_1$	Neighboring disruption analytic $\mathrm{NNDA}(n)$	CLOC 1
$\boldsymbol{\mathcal{A}}_2$	Harmonic centrality analytic $\mathrm{NHCA}(n)$	CLOC 1
$\boldsymbol{\mathcal{A}}_3$	Rate of disruption propagation analytic $\mathrm{NRDPA}(n)$	CLOC 1
$\boldsymbol{\mathcal{A}}_4$	Maximum disruption propagation analytic $\mathrm{NMDPA}(n)$	CLOC 1
$\boldsymbol{\mathcal{A}}_5$	Team strategic compatibility index $\mathrm{TSCI}(\mathrm{AL})$	CLOC 1, 2
$\boldsymbol{\mathcal{P}}_1$	Random team selection protocol	None
$\boldsymbol{\mathcal{P}}_2$	Low-compatibility team selection protocol	CLOC 1, 2
$\boldsymbol{\mathcal{P}}_3$	Medium-compatibility team selection protocol	CLOC 1, 2
$\boldsymbol{\mathcal{P}}_4$	High-compatibility team selection protocol	CLOC 1, 2

6.3.2 CTCD dynamic coordination decisions

With respect to the dynamic coordination decisions $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$, three CTCD analytics are developed to monitor the state of $\boldsymbol{\mathcal{C}}$, $\boldsymbol{\mathcal{R}}$, and $\boldsymbol{\mathcal{D}}$. The analytics can be used to evaluate the performance of the system, and to support the decision-making process of the response team AL. The analytics developed based on the modeling and simulation logic and reflect the state of the system.

The sixth CTCD analytic $\boldsymbol{\mathcal{A}}_6$, total disruption strength analytic, is defined as $\mathrm{TDS}(t) = \sum_n^{\mathrm{NL}} \mathrm{NDS}(n,t)$, which reflects the total number of disruptions present in the client system $\boldsymbol{\mathcal{C}}$ at time t.

The seventh CTCD analytic $\boldsymbol{\mathcal{A}}_7$, the node response task analytic $\mathrm{NRTA}(n) \in \mathbb{R}_{>0}$, is defined as the average of all the agents' response times for this node, if it is disrupted. The response task analytic is calculated as $\mathrm{NRTA}(n) = \sum_a^{\mathrm{AL}} \frac{\mathrm{RRM}(a,n)}{|\mathrm{AL}|}$.

Based on $\mathrm{NRTA}(n)$, the eight CTCD analytic $\boldsymbol{\mathcal{A}}_8$, the total response workload, is defined as $\mathrm{TRW}(t) \in \mathbb{R}_{\geq 0}$. This analytic reflects the expected workload needed to respond to all disrupted nodes $\{n \in \mathrm{NL}: \mathrm{NDS}(n,t) = 1\}$ in the set of nodes NL.

$$\mathrm{TRW}(t) = \sum_{n}^{\{n \in \mathrm{NL}:\mathrm{NDS}(n,t)=1\}} \mathrm{NRTA}(n) \tag{39}$$

Four CTCD coordination protocols, $\boldsymbol{\mathcal{P}}_5$ to $\boldsymbol{\mathcal{P}}_8$, are developed for $\boldsymbol{\mathcal{S}}_{\boldsymbol{\mathcal{R}}}^{t}$.

$\boldsymbol{\mathcal{P}}_5$: first-come-first-serve protocol (FCFS), a baseline protocol, prioritizes disrupted nodes that were disrupted earlier. The tie-breaker for this rule is the lower node ID. The corresponding FCFS selection index of each node n is defined as $\mathrm{NLDT}(n) \in \mathbb{R}_{\geq 0}$, which is recorded by the simulator.

$\boldsymbol{\mathcal{P}}_6$: shortest processing (response) time (SPT) protocol, also a baseline protocol, prioritizes the nodes with the shortest response time, for the agent being considered. The tie-breaker for this rule is the lower $\mathrm{NLDT}(n)$ and then the lower node ID. The corresponding SPT selection index of each node n, for a given agent a, is $\mathrm{RRM}(a, n)$.

$\boldsymbol{\mathcal{P}}_7$: minimizing neighboring disruption propagation (MNDP) protocol, which prioritizes the nodes with lower average un-disrupted edge propagation time. This protocol is developed based on $\boldsymbol{\mathcal{A}}_6$, $\mathrm{TDS}(t)$, and seeks to minimize the growth of the total disruption strength, $\frac{d}{dt}\mathrm{TDS}(t)$. The MNDP protocol utilizes the important interaction between the response mechanisms and disruption propagation: an agent's response to a node halts all incoming and outgoing disruption propagation from that node. A more basic version of this index $\mathrm{NMND}(n)$ is illustrated in Figure 6.4, where node B is prioritized because it has more succeeding nodes that have not been disrupted.

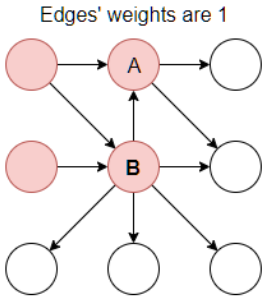

Figure 6.4. MNDP protocol illustration

The undirected and unweighted version of MNDP is discussed in (Zhong, 2016) as the activity-based priority scheduling protocol. The MNDP improves upon the activity-based priority scheduling protocol by (1) adjusting to the directed network by considering only succeeding undisrupted nodes; (2) prioritizing the nodes with lower (which means faster) disruption propagation time. The MNDP selection index of each node n is defined as $\mathrm{NMND}(n) \in \mathbb{R}_{>0}$, which is calculated as followed:

$$\mathrm{NMND}(n) = \begin{cases} \sum_{e=(n_i,n_j)}^{\mathrm{NSEL}_{\mathrm{MNDP}}(n)} \frac{\mathrm{EDPT}(e)}{|\mathrm{NSEL}_{\mathrm{MNDP}}(n)|}, & \text{if } |\mathrm{NSEL}_{\mathrm{MNDP}}(n)| > 0 \\ \infty, & \text{if } |\mathrm{NSEL}_{\mathrm{MNDP}}(n)| = 0 \end{cases} \tag{40}$$

with $\mathrm{NSEL}_{\mathrm{MNDP}}(n) = \{e = (n_i, n_j) \in \mathrm{NSEL}(n) \mid \mathrm{NDS}(n_j, t) = 0 \text{ and } \mathrm{NAA}(n_j, t) = \emptyset\}$

Nodes with no un-disrupted succeeding node receive a very large value to $\mathrm{NMND}(n)$, and are with the lowest response priority, because disruption cannot propagation from them. The tie-breaker for this rule is the lower processing time for the agent a being considered $\mathrm{RRM}(a, n)$, then the lower $\mathrm{NLDT}(n)$ and then the lower node ID.

$\boldsymbol{\mathcal{P}}_8$: minimizing additional task workload (MATW) protocol, which improves upon MNDP. This coordination protocol is developed based on $\boldsymbol{\mathcal{A}}_6$, $\boldsymbol{\mathcal{A}}_7$, and $\boldsymbol{\mathcal{A}}_8$, and seeks to minimize the growth of the total response workload, $\frac{d}{dt}\mathrm{TRW}(t)$. Similar to MNDP, the MATW protocol utilizes the important interaction between the response mechanisms and disruption propagation: an agent's response to a node halts all incoming and outgoing disruption propagation from that node. With MATW, the agent will prioritize the node that, if disrupted, will lead to the most additional workload on the agent network. Compared to the other three coordination protocols $\boldsymbol{\mathcal{P}}_5$, $\boldsymbol{\mathcal{P}}_6$, and $\boldsymbol{\mathcal{P}}_7$, the MATW protocol provides a balance of minimizing disruption propagation as well as the agent network's processing times.

Then, the corresponding MATW selection index of each node n is defined as $\mathrm{NMAT}(n) \in \mathbb{R}_{>0}$, which is calculated as followed:

$$\mathrm{NMAT}(n) = \begin{cases} \sum_{e=(n_i,n_j)}^{\mathrm{NSEL}_{\mathrm{MNDP}}(n)} \dfrac{\mathrm{EDPT}(e)}{\mathrm{NRTA}(n_j)}, & \text{if } |\mathrm{NSEL}_{\mathrm{MNDP}}| > 0 \\ \infty, & \text{if } |\mathrm{NSEL}_{\mathrm{MNDP}}| = 0 \end{cases} \tag{41}$$

with $\mathrm{NSEL}_{\mathrm{MNDP}}(n) = \{e = (n_i, n_j) \in \mathrm{NSEL}(n) \mid \mathrm{NDS}(n_j, t) = 0 \text{ and } \mathrm{NAA}(n_j, t) = \emptyset\}$.

Nodes with no un-disrupted succeeding node receive a very large value to $\mathrm{NMAT}(n)$. The tie-breaker for this rule is the lower processing time for the agent a being considered $\mathrm{RRM}(a, n)$, then the lower $\mathrm{NLDT}(n)$ and then the lower node ID.

Table 6.7. Summary of the CTCD coordination analytics and protocols

Analytic/ Protocol	**Description**	**CLOC guideline**
$\boldsymbol{\mathcal{A}}_6$	Total disruption strength $\mathrm{TDS}(t)$	CLOC 1
$\boldsymbol{\mathcal{A}}_7$	Node response task analytic $\mathrm{NRTA}(n)$	CLOC 3
$\boldsymbol{\mathcal{A}}_8$	Total response workload $\mathrm{TRW}(t)$	CLOC 1, 3
$\boldsymbol{\mathcal{P}}_5$	First-come-first-serve FCFS	None
$\boldsymbol{\mathcal{P}}_6$	Shortest-processing-time SPT	None
$\boldsymbol{\mathcal{P}}_7$	Minimizing neighboring disruption propagation MNDP	CLOC 1-3
$\boldsymbol{\mathcal{P}}_8$	Minimizing additional task workload MATW	CLOC 1-3

6.4 Numerical Experiments and Results

Two sets of numerical experiments are conducted to validate the CTCD model, analytics, and protocols.

1. The first set of numerical experiments are conducted to validate only the CTCD coordination protocols.
2. The second set of numerical experiments are conducted to validate both the CTCD teaming protocols and CTCD coordination protocols.

6.4.1 The first set of CTCD numerical experiments

The first set of numerical experiments involves a random directed and weighted network adapted from the BA random network model from Floyd (1962). The undirected and unweighted network is generated following the BA random network model. Then, the procedure is adapted to add directions and weights to the edges. Each edge receives a probability of 1/3 to be bidirectional, 1/3 to be directional towards the node with the lower node ID, and 1/3 to be directional towards the node with the higher node ID. Then, each edge independently and randomly receives a weight with distribution Uniform (0.5, 1.5). The response requirement matrix is also randomly generated with each agent receiving an inherent value from Uniform (0.5, 1.5). Then for each agent, the response times to the nodes receive values from Uniform (0.5, 1.5) multiplied by its inherent value as well. The size of the client system is 400 nodes. A total of 9 disruption scenarios, presented in Table 6.8, are run, with 4 different online scheduling protocols, and 400 replications each. The graphical results with 95% confidence intervals are presented in Figure 6.5.

Table 6.8. CTCD first set of experiments – disruption scenarios

Scenario	Number of disruptions	Number of agents
1	25% of node count = 100	10% of disruption count = 10
2	25% of node count = 100	20% of disruption count = 20
3	25% of node count = 100	30% of disruption count = 30
4	50% of node count = 200	10% of disruption count = 20
5	50% of node count = 200	20% of disruption count = 40
6	50% of node count = 200	30% of disruption count = 60
7	75% of node count = 300	10% of disruption count = 30
8	75% of node count = 300	20% of disruption count = 60
9	75% of node count = 300	30% of disruption count = 90

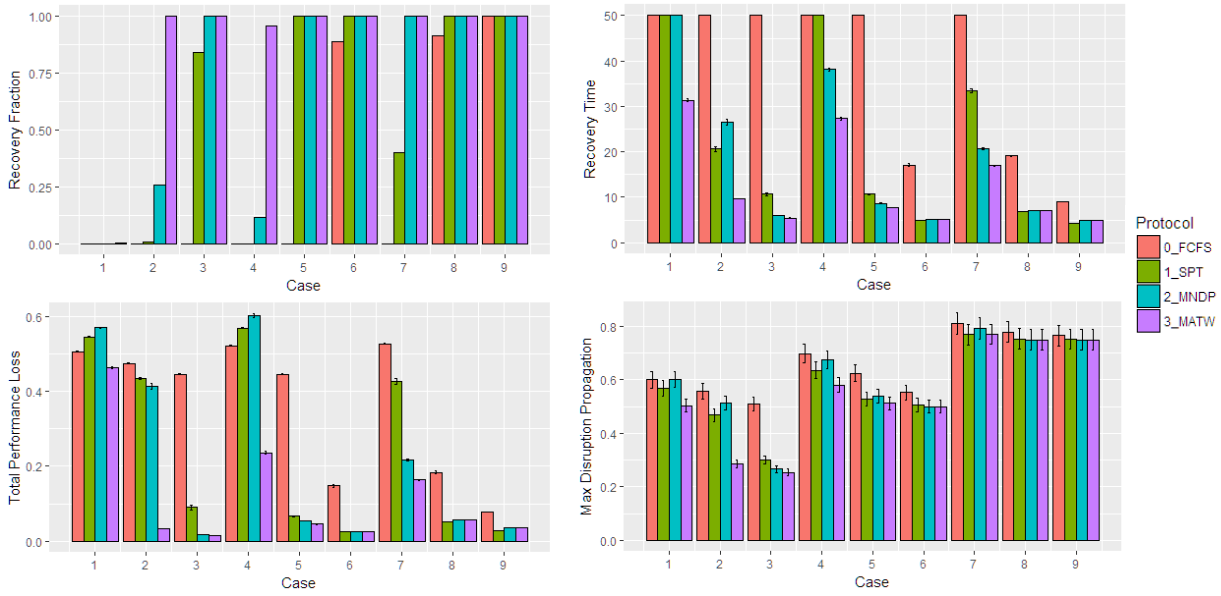

Figure 6.5 CCTD first set of experiments grouped by CCTD coordination protocols and response/disruption scenarios, with 95% confidence interval bars

With respect to recovery fraction $\mathcal{M}_1$: RF, the CTCD coordination protocols MNDP and MATW perform significantly better (providing higher values) than the baseline protocols FCFS and SPT in cases 1-7, and the same for cases 8 and 9. This implies that MNDP and MATW are more efficient in terms of number of agents used. With respect to recovery time $\mathcal{M}_2$: RT (minimization objective), some mixed results are seen, but FCFS performs worst in all cases. Regarding total performance loss $\mathcal{M}_3$: TPL, MATW generally performs better (providing lower values) compared to MNDP, which in turn performs significantly better than FCFS and SPT. In the cases with more agents (cases 3, 5, 6, 8, 9), however, the results are significantly different between the three CTCD coordination protocols SPT, MNDP and MATW. With respect to maximum disruption propagation fraction $\mathcal{M}_4$: MDPF (minimization objective), except with cases 2 and 3, the different CTCD coordination protocols do not perform significantly different from each other. Regarding total response fraction, both MNDP and MATW generally perform better than SPT and FCFS. It can be concluded that the online scheduling protocols MNDP and MATW perform better in the majority of the performance metrics compared to the baseline protocols FCFS and SPT. It is also observed that SPT, MNDP, and MATW performs better than FCFS for most cases and performance metrics.

6.4.2 The second set of CTCD numerical experiments

In this section, experiments are conducted to illustrate the CTCD teaming analytics and protocols with respect to three types of random network models: BA (Barabasi & Albert, 1999), ER (R. Albert & Barabasi, 2002), and WS (Erdös & Rényi, 1959). The details of the experiments are as follows.

Table 6.8. CTCD first set of experiments – disruption scenarios

Factor	# variations	Details
Network type	3	BA random network vs ER random network vs WS random network, all 100-node and 200-edge.
CTCD teaming protocols	4	$\boldsymbol{\mathcal{P}}_1$: random, $\boldsymbol{\mathcal{P}}_2$: low-compatibility, $\boldsymbol{\mathcal{P}}_3$: medium-compatibility, $\boldsymbol{\mathcal{P}}_4$: high-compatibility
CTCD Coordination protocols	4	$\boldsymbol{\mathcal{P}}_5$: FCFS, $\boldsymbol{\mathcal{P}}_6$: SPT, $\boldsymbol{\mathcal{P}}_7$: MNDP, $\boldsymbol{\mathcal{P}}_8$: MATW

The three random network models mentioned above are used for the network, with 100 replications each. The networks are created with 100 nodes and 200 edges for all three types. The BA networks are created with 2 initial nodes, and the growth rate of 2 edges per new node, until 200 edges are reached. The ER networks are created with the aforementioned number of nodes and number of edges, and only fully connected networks are selected. The WS networks are created with mean degree 4, and rewiring probability of 0.5. Because the three random network models mentioned are undirected and unweighted networks, adjustments are required for it to work with the CTCD model. For each undirected and unweighted edge, there is a 2/3 probability for the edge to be unidirectional and 1/3 probability for the edge to be converted to two directed edges of opposite directions. Each directed edge e receives a weight $\mathrm{EDPT}(e)$ ranging from 0.5 to 1.5, uniformly distributed. With respect to disruption propagation, 25 initial disruptions, selected randomly based on uniform distribution, are selected. The parameters of 10 agents and 25 initial disruptions for the 100-node networks are selected based on the previous work of Watts (2002). The four aforementioned CTCD coordination protocols for the repair agents are used.

With respect to the response teams, a pool of 1000 teams of 10 response agents each is created. Each team receives an across-agent-variation index $\mathrm{AAVI}(\mathrm{AL})$ with random distribution

$\mathrm{UNIF}(0,1)$, which determines the degree of variation between agents. Additionally, each agent receives an across-node-variation index $\mathrm{ANVI}(a)$ with random distribution $\mathrm{UNIF}(0,1) \times \mathrm{AAVI}(\mathrm{AL})$, which determines the degree of variation in terms of response time for that agent to the different nodes. Then, for each agent, the unnormalized response time $\mathrm{URT}(a,n)$ for each node (out of 100) is generated with random distribution $\mathrm{UNIF}\big(1,1+\mathrm{ANVI}(a)\big)$. Then, the unnormalized response time is normalized so that the average response time across all nodes for each agent is equal to 1. This procedure results in the creation of diverse teams and uniform teams. The diverse teams have higher values of $\mathrm{AAVI}(\mathrm{AL})$ and have more diverse agents, whereas the uniform team have lower values of $\mathrm{AAVI}(\mathrm{AL})$ and have more uniform agents. The more diverse agents have higher values of $\mathrm{ANVI}(a)$ and tend to have a wider range of response times across all nodes, whereas the uniform agents have lower values of $\mathrm{ANVI}(a)$ and tend to have similar values of response times. All agents, however, have an average response time of 1 across all nodes, thus, all teams are economically balanced. The important goal of a response team is to effectively handle disruptions and their propagation. Simulating the full CRDP model with 1000 provided teams would be computationally expensive. Therefore, the CTCD teaming analytics and protocols are applied to guide the team selection decision, which result in the $\mathrm{TSCI}(\mathrm{AL})$ for the 1000 provided teams. For this set of experiments, 4 groups of $\mathrm{TSCI}(\mathrm{AL})$ are selected: the high-compatibility group which consists of the top 10 teams based on $\mathrm{TSCI}(\mathrm{AL})$ ranking; the medium-compatibility group which consists of the middle 10 teams; the low-compatibility group which consists of the lowest 10 teams; and a random group of 10 teams.

Comparisons between strategic compatibility levels are provided in Figure 6.6 and Table 6.9.

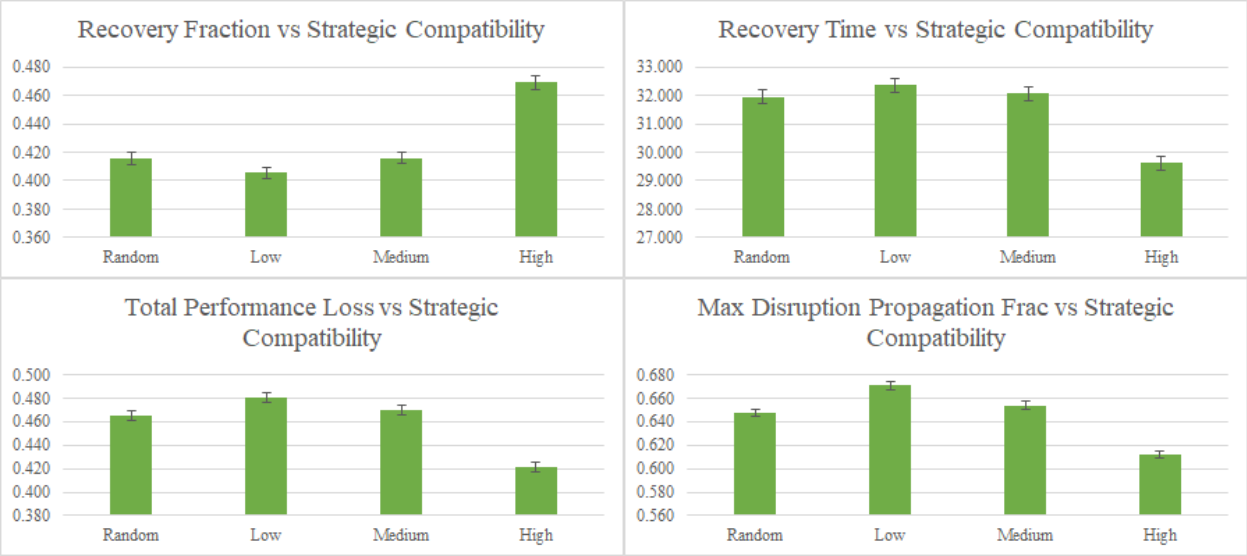

Figure 6.6. CCTD experiment results grouped by CCTD teaming protocols, with 95% confidence interval bars

Table 6.9. Comparison table of strategic compatibility levels

Strategic Compatibility	RF	RT	TPL	MDPF
Random	0.416	31.941	0.465*	0.647*
Low	0.405*	32.368	0.480	0.671*
Medium	0.416	32.055	0.469	0.654*
High	**0.469***	**29.628***	**0.421***	**0.612***
*: indicates statistical significance at $\alpha = 0.05$ **Bolded values** are the best values of a metric				

The high strategic compatibility teams significantly outperform the other three team types by 12.7%-15.6% in RF, by 7.3%-8.5% in RT, by 9.6%-12.4% in TPL, and by 5.5%-8.8% in MDPF. Overall, the high strategic compatibility teams is statistically proven to provide the best performance, followed by either the randomly selected teams or the medium strategic compatibility teams, then by the low strategic compatibility teams. The next comparisons are between strategic compatibility levels and online response protocols, which are provided in Figure 6.7 and Table 6.10.

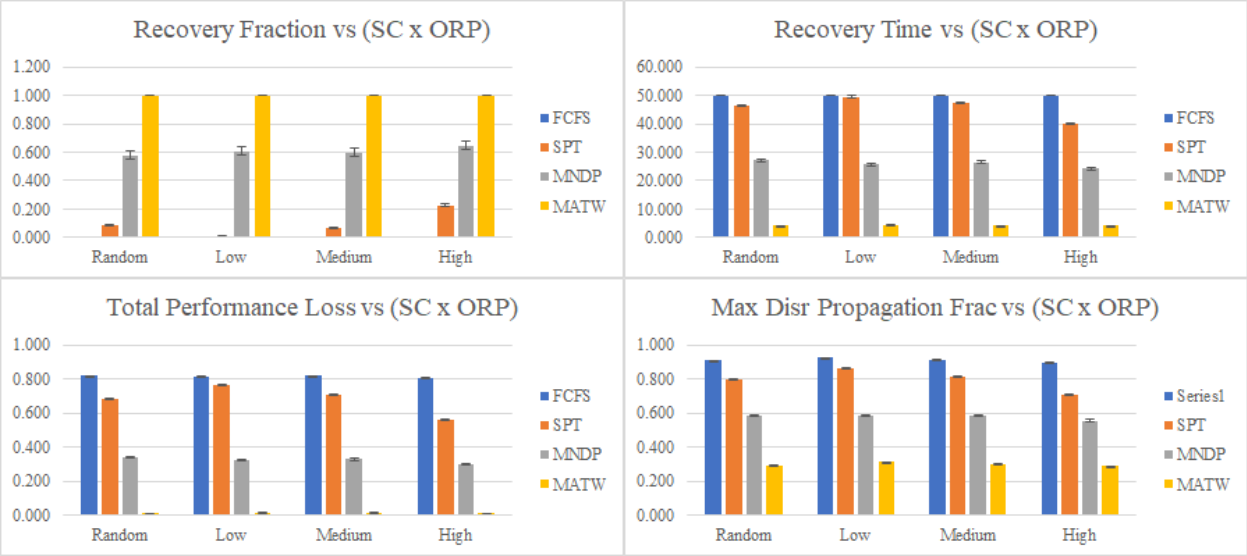

Figure 6.7. Comparison chart of CTCD teaming protocols and CTCD coordination protocols (with 95% confidence intervals)

Table 6.10. Comparison table of CTCD teaming protocols and CTCD coordination protocols

SC	**RP**	RF	RT	TPL	MDPF		**RP**	RF	RT	TPL	MDPF
Rand	FCFS	0.000	50.0	0.820	0.908*		MNDP	0.58*	27.1*	0.34	0.59
Low	FCFS	0.000	50.0	0.81*	0.924*		MNDP	0.681	25.6*	0.32	0.58
Med	FCFS	0.000	50.0	0.81	0.913*		MNDP	0.60	26.6*	0.33	0.58
High	FCFS	0.000	49.9	**0.80***	**0.895***		MNDP	**0.65***	**24.3***	**0.30***	**0.56***
Rand	SPT	0.087*	46.3*	0.684*	0.799		MATW	1.00	4.30	0.015	0.29*
Low	SPT	0.014*	49.4*	0.765*	0.861*		MATW	1.00	4.40*	0.016	0.31*
Med	SPT	0.068*	47.3*	0.711*	0.815		MATW	1.00	4.33	0.016	0.30*
High	SPT	**0.227***	**40.1***	**0.559***	**0.706***		MATW	1.00	**4.15***	**0.014***	**0.29***

*: indicates difference to all other values with statistical significance at $\alpha = 0.05$.
Non-*: indicates confidence interval overlap with at least one non-* value at $\alpha = 0.05$.
SC = strategic compatibility, RP = response protocol, Rand = random, Med = medium
Best values of a metric of a category when comparing strategic compatibility are **bolded**.

When the coordination protocol FCFS is employed, the high strategic compatibility teams provide only limited improvement of 0.6%-1.3% in the reduction of total performance loss and 1.3%-3.7% in the reduction of maximum disruption propagation. The result is explained by the very low effectiveness of the FCFS protocol in preventing the propagation of disruptions. With the

coordination protocol SPT, the high strategic compatibility teams significantly improve the resilience of the client system: 161%-1540% increase in recovery fraction, 13.4%-18.9% reduction in recovery time, 18.2%-26.9% reduction in performance loss, and 11.6%-18% reduction in maximum disruption propagation. It is can be concluded that the SPT protocol highly depends on the appropriate strategic preparation of agent teams. With the coordination protocol MNDP, the high strategic compatibility teams provide a high improvement in resilience: 6.8%-11.2% increase in recovery chance, 5.3%-10.5% reduction in recovery time, 7.6%-12.2% reduction in performance loss, and 4.6%-5.2% reduction in maximum disruption propagation. With the coordination protocol MATW, the high strategic compatibility teams provide lower improvement in resilience (compared to SPT and MNDP): 3.6%-5.8% reduction in recovery time, 6.5%-11.6% reduction in total performance loss, and 2.1%-7.7% reduction in maximum disruption propagation. The lower improvement can be partially explained by the high effectiveness of the coordination protocol MATW.

From the results of the experiments, it can be concluded that the high strategic compatibility teams provide superior performance compared to other team types, demonstrating the effectiveness of employing the CLOC principle in the selection of response teams. The higher performance is most notable with the usage of the coordination protocol SPT, followed by MNDP, then by MATW. It is also noted that the medium strategic compatibility teams provide around the same level of performance as the randomly selected teams.

6.5 Concluding Remarks

In this chapter, the CTCD model is defined and formulated based on the CRDP framework, and the accompanying CTCD analytics and protocols are developed based on the guidelines of the CLOC principle. The CTCD model explores two types of response decisions: the teaming decisions made off-line and the dynamic coordination decisions made during real-time. The CTCD model also the recurring nature of disruption propagation. In the CTCD model, one team of response agents must be selected to be on standby, and only agents in this team can respond to disruptions affecting the client system. Five CTCD teaming analytics and four CTCD teaming protocols are developed based on the CLOC principle, together with three CTCD coordination analytics and four CTCD coordination protocols. The CTCD model and protocols are validated

with numerical experiments. The experiment results indicate that the advanced coordination protocols developed based on the CLOC principle outperform the baseline and less advanced protocols by at least 50%, with statistical significance. The experiment results also indicate that the advanced teaming protocols outperform the baseline protocols by 2.1% to 12.1%, with statistical significance. The results indicate that the appropriate application of the CRDP formulation and the CLOC design and control principles can lead to significant performance improvement.

Thus, this case in CHAPTER 6 provides a partial answer to both Research Question 2 and Research Question 3 as outlined in CHAPTER 1.

CHAPTER 7. CONCLUSIONS

7.1 Summary of Design Recommendations

As discussed in CHAPTER 2, the response to disruption propagation (RDP) problem exists in different domains: fire spreading, agricultural plant disease, propagating computer malware, and supply network disruptions. The consequences of disruption propagation can be catastrophic, with significant economic damages, personal injuries, and even deaths. Engineers and managers of complex systems subjected to RDP problems are recommended to apply the CRDP framework and the CLOC principle to better prepare and coordinate the response activities. The design recommendations are as follows.

1. It is recommended that the components, interactions, decisions, and system performance metrics of the RDP problem concerned be systematically specified and characterized in accordance with the CRDP framework. Employing the CRDP framework enables the practitioners to better understand the RDP problem concerned. Furthermore, the CRDP framework enables analogical reasoning across different RDP problem domains. Observing and analyzing a strategy employed in a different problem domain can potentially lead to the development of novel decision-making strategies and methods for the RDP problem concerned. This design recommendation is based on the CRDP formulations of three case studies presented in this dissertation (Section 4.2, 5.2, and 6.2).
2. It is recommended that the CLOC principle be applied in the development of analytics and protocols to guide the preparation, planning, and coordination decisions of response resources. Specifically,
    a. It is recommended that the disruption propagation behaviors are modeled as a complex network in accordance with the first CLOC guideline. The entities of the client system can be represented as nodes, and the disruption propagation directions can be represented as edges. The network modeling allows better situation awareness and a better understanding of disruption propagation behavior through the use of complex network analysis such as centrality analysis and path analysis.

b. It is recommended that the propagation-restraining effects are identified and utilized, per the second CLOC guideline. Preventing potential severe propagation can reduce the total catastrophic damage incurred to the client system affected and further reduce the response workload, providing better response performance.
c. It is recommended that the practitioners identify and utilize the collaborative and synergistic mechanisms of having multiple response resource groups/agent teams available, per the third CLOC guideline. Particularly, the decision-making of new response decisions should consider past response decisions and concurrent response decisions. Utilizing collaboration and synergy can ensure the coverage of the propagation-restraining effect, further improving response performance
d. The aforementioned design recommendations are based on the CLOC principle (Section 3.2) and the analytics and protocols developed for the three case studies presented in this dissertation (Section 4.3, 5.3, and 6.3).

3. In RDP problems involving detecting hidden disruptions, it is recommended that the design of detection analytics and protocols prioritizes the locations/nodes with the highest likelihood of being disrupted. Furthermore, new detection decisions should consider past detection decisions and results, as well as concurrent detection decisions. This design recommendation is based on the analytics and protocols developed for the CDUD model (Section 4.3).
4. In RDP problems involving allocating static resources to prevent unexpected disruptions, it is recommended that the practitioners ensure the coverage of static resource deployment. The coverage of resources can be evaluated through the use of network modeling. This design recommendation is based on the analytics and protocols developed for the CSPD model (Section 5.3).
5. In RDP problems involving selecting/forming teams of agents to standby against disruption propagation, it is recommended that the team forming and preparation process evaluates the strategic compatibility of the teams, and selects the team with the highest strategic compatibility. This allows the team to respond to a wide range of disruption scenarios. This design recommendation is based on the teaming analytics and protocols developed for the CTCD model (Section 6.3).

6. In RDP problems involving repair and/or removal of recurring disruptions, it is recommended that the propagation-restraining effect is analyzed and utilized effectively. This is because recurring disruptions can re-disrupt the nodes that were previously repaired, forming a competition between disruption propagation and response. Ineffective and/or insufficient response can lead to the disruptions achieving a critical mass, after which the propagation can not be stopped by the response resources available. This design recommendation is based on the dynamic coordination analytics and protocols developed for the CTCD model (Section 6.3).

It is also recommended that the CRDP framework and the CLOC principle be appropriately adapted for the specific RDP problems. Even though three case studies are provided in this dissertation (with four related case studies published in the literature), the real-life problem contexts could be significantly different and more complex compared to the case studies discussed. Therefore, it is necessary to appropriately adapt the CRDP framework and the CLOC principle, which are designed to be general and applicable to multiple different cases, to the corresponding problem contexts. The components and subsystems of the client system $\boldsymbol{\mathcal{C}}$ can be represented as nodes, and various node attributes can be defined and specified to represent the characteristics of the nodes. Different node types, such as client/server in computer networks, can be specified. For problems with physical traveling, node locations can be specified. For client systems with heterogeneous node importance, different node weights can be specified. The response mechanisms in $\boldsymbol{\mathcal{R}}$ can be defined together with different attributes to reflect their characteristics. Response types can be specified, depending on their activities. For problems with physical traveling, response location is also an important attribute. For problems with heterogeneous response capabilities, possible response attributes include traveling speed, response speed, response quality, detection accuracy, etc. The disruptions and disruption propagation in $\boldsymbol{\mathcal{D}}$ can also be defined with different attributes to reflect their characteristics. Disruptions can have different types, different severity (high/low) different targets (certain disruptions can only affect certain node types, for example), and/or different attack frequency.

Characterizing the interactions $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$, $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$, and $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$ is more sophisticated due to the specific nature of the RDP problem of concerned. The CLOC principle has provided two common

interactions applicable to the investigated RDP problems: the disruption propagation behavior in $\boldsymbol{\mathcal{D}}$&$\boldsymbol{\mathcal{C}}$ and the propagation-restraining effect of response in $\boldsymbol{\mathcal{R}}$&$\boldsymbol{\mathcal{D}}$. Other interactions are specific to the RDP problems of concern, and several examples are provided in Subsection 3.1.2. Similarly, the design and control decisions $\boldsymbol{\mathcal{S}}_{\#}$ and $\boldsymbol{\mathcal{S}}_{\#}^{t}$ are specific to the RDP problems of concern, and several examples are provided in Subsection 3.1.3. The system performance metrics $\boldsymbol{\mathcal{M}}$, however, are more likely to be applicable to multiple different cases. The $\boldsymbol{\mathcal{M}}$ examples are provided in Subsection 3.1.4.

The possible applications and adaptation considerations of the CRDP framework are summarized in Table 7.1.

Table 7.1. Summary of CRDP applications and modeling adaptations

Aspect	Examples and modeling adaptation considerations	
$\mathcal{C}$: client system	Examples: building complex, forests; plant plots; computers, sensor nodes; firms, machines	Modeled as nodes. Attributes: node type, node status, node location, node importance
$\mathcal{R}$: response mechanisms	Examples: human responder, autonomous agents, robots, local resources	Attributes: response type, response status, response location, response capability (speed, quality, accuracy)
$\mathcal{D}$: disruption	Examples: fire; plant diseases; propagating computer malware; supply network disruption	Attributes: disruption target, disruption frequency, disruption severity
$\mathcal{C}$&$\mathcal{R}$: client-response interaction	Examples: information sharing, location restrictions, response compatibility Adapt attributes and simulation logic as necessary	
$\mathcal{R}$&$\mathcal{D}$: response-disruption interaction	Examples: disruption-restraining effect, disruption awareness. Adapt attributes and simulation logic as necessary	
$\mathcal{D}$&$\mathcal{C}$: disruption-client interaction	Network modeling of disruption propagation as edges (proximity, connections, flows). Edges can have different types and attributes. Other examples: recurring disruption propagation, propagation speed/probability Adapt attributes and simulation logic as necessary	
$\mathcal{S}_{\#}$: design decisions	$\mathcal{S}_{\mathcal{C}}$ examples: network design $\mathcal{S}_{\mathcal{R}}$ examples: local resource allocation, backup inventory $\mathcal{S}_{\mathcal{D}}$ examples: intelligent disruption off-line targeting	
$\mathcal{S}_{\#}^{t}$: control decisions	$\mathcal{S}_{\mathcal{C}}^{t}$ examples: node movement or status change $\mathcal{S}_{\mathcal{R}}^{t}$ examples: disruption removal, detection, prevention; active negotiation $\mathcal{S}_{\mathcal{D}}^{t}$ examples: intelligent disruption on-line targeting	
$\mathcal{M}$: system performance metrics	Examples: recovery likelihood, recovery reliability, recovery time; total performance loss; maximum disruption propagation	

7.2 Summary of Original Contributions

This dissertation investigates the research problem of response to disruption propagation in complex systems of systems: cyber-physical systems, building complexes, supply networks, computer networks, and other critical infrastructures. This problem is significant due to the devastating effects of disruption propagation: fire spreading, infectious plant disease, propagating computer malware, to name a few. Furthermore, there exists no unifying framework to connect the

different RDP problem contexts and no general methodologies guide the design and control decisions. To address the aforementioned challenges, the CRDP framework and the CLOC principle are developed. The CRDP framework enables and augments the classification, categorization, and characterization of the important aspects of different RDP problems, augmenting analysis and decision-making. Building upon the CRDP framework, the CLOC guides and supports the analysis and decision-making process of the response mechanisms against disruption propagation. Specifically, the CLOC principle can be employed to support the development of CCT analytics and protocols to improve response performance and client system resilience, as shown in the numerical experiments conducted in this research. The augmented decision-making can significantly benefit practitioners, engineers, and managers involved in problem contexts dealing with disruption propagation, such as network security, supply network disruptions, fire spreading, and agricultural plant diseases (CHAPTER 2).

This dissertation addresses the challenge of responding to disruption propagation in complex systems. As discussed in CHAPTER 2, disruption propagation can cause catastrophic economic losses as well as injuries and deaths. To address this challenge, three research questions are outlined and addressed as follows:

> **Research Question 1:** What is a good framework for systematic identification and characterization of different RDP problems and their corresponding models?

Answer: The new CRDP framework is developed as a unifying framework for classification, categorization, and characterization of the important aspects of different RDP problems. Such aspects include the components, the interactions between the components, the decisions of the components, and the system performance metrics. The CRDP framework is validated by the formulations of three case studies presented in this dissertation (Section 4.2, 5.2, and 6.2).

> **Research Question 2:** What are the necessary components and interactions to be identified and characterized to enable systematic formulation of different RDP problems and their corresponding RDP models?

Answer: The necessary CRDP components are the client system $\boldsymbol{\mathcal{C}}$, the response mechanisms $\boldsymbol{\mathcal{R}}$, and the disruption propagation $\boldsymbol{\mathcal{D}}$. The necessary CRDP interactions are the client-response interactions $\boldsymbol{\mathcal{C}}\&\boldsymbol{\mathcal{R}}$, the response-disruption interaction $\boldsymbol{\mathcal{R}}\&\boldsymbol{\mathcal{D}}$, and the disruption-client interaction $\boldsymbol{\mathcal{D}}\&\boldsymbol{\mathcal{C}}$. Two types of decisions are characterized, the design decisions $\boldsymbol{\mathcal{S}}_{\#}$ and the control decisions $\boldsymbol{\mathcal{S}}_{\#}^{t}$, and the system performance metrics $\boldsymbol{\mathcal{M}}$ are specified. The aforementioned CRDP components and interactions are validated by the formulations of three case studies presented in this dissertation (Section 4.2, 5.2, and 6.2).

Research Question 3: Based on the answers to Research Questions 1 and 2, what collaborative design and control principles can be developed to provide better response against disruptions and their propagation?

Answer: The new Covering Lines of Collaboration (CLOC) principle is developed to guide and support the analysis and decision-making process of the response mechanisms against disruption propagation. The CLOC principle supports the development of CCT analytics and protocols specifically for the RDP problems, and the principle consists of 3 guidelines. The first CLOC guideline specifies the network modeling of disruption propagation behavior and patterns. The network modeling allows a better understanding of the interactions between the CRDP components, better situation awareness, and more sophisticated analysis to be employed (such as network centrality analysis and statistical inference). The second CLOC guideline specifies the analysis of the propagation-restraining effect and utilizing this effect in the development of collaborative design and control protocols for the response decisions. The third CLOC guideline specifies the development of collaborative analytics and protocols for the response decisions. Collaboration between response mechanisms can ensure the coverage of the propagation-restraining effect, improving the performance of the response mechanisms. The CLOC guideline is validated by the development of the analytics and protocols for each of the three case studies presented in this dissertation (Section 4.3, 5.3, and 6.3). Experiment results indicate that advanced CLOC-based decisions significantly outperform the baseline and less advanced protocols for all three cases, with performance superiority of 9.7-32.8% in case 1; 31.1%-56.6% in case 2; 2.1%-12.1% for teaming protocols, and at least 50% for team coordination protocols in case 3.

The mapping between the Research Questions and the developed concepts is provided in Table 7.2.

Table 7.2. Mapping between Research Questions and concepts

Research Question	Relevant Chapters/Sections
Research Question 1 – Framework	Section 3.1 - The CRDP Framework
Research Question 2 – Formulation	Subsection 3.1.1 - The CRDP components Subsection 3.1.2 - The CRDP interactions Subsection 3.1.3 - The CRDP decision spaces Subsection 3.1.4 - The CRDP system performance metrics The application of the CRDP formulation is demonstrated with Subsections 4.2, 5.2, 6.2
Research Question 3 – Design and Control Principles	Section 3.2 - The Covering Lines of Collaboration (CLOC) Principle Subsection 3.2.1 - The third CLOC guideline – collaboration between response Subsection 3.2.2 - The second CLOC guideline – restraining disruption propagation Subsection 3.2.3 - The third CLOC guideline – collaboration between response The application of the CLOC principle is validated with Subsections 4.3, 5.3, 6.3

7.3 Limitations and Future Research Directions

While this research has established the fundamental components and interactions of the CRDP framework, its limitations must be addressed by future research:

1. General Assumption 1 states that disruptions are strictly harmful to the client system, which is not necessarily applicable to the problems where disruptions can be both positive and negative to the client system. One example is a case in a supply network where production disruptions of one good X in supply networks can be economically profitable to the firms that produce the substitute goods Y of the disrupted goods X. In this case, the demand for Y increases and provides an opportunity for the firms that produce Y to increase profit,

whereas the firms that produce X would lose profit from the production disruptions. Firms for both X and Y may have the same suppliers, which could be relatively unimpacted from the disruptions, due to the loss of sales to firms producing X being offsetted by the increase of sales to firms producing Y.

2. General Assumption 3 states that responses are strictly beneficial to the client system, which is not necessarily applicable to problems where responses can bring about unintended negative consequences to the client system. In the agricultural plant disease case, the disease detection activities can inadvertently spread the diseases from one plant to other plants. This phenomenon can happen because the disease particles can latch on to the clothes (of the farmers) or the robotic arms (of robot agents), and then be brought into physical contact of other plants.
3. The case studies investigated in this research only discuss the response decisions $\boldsymbol{S_R}$ and $\boldsymbol{S_R^t}$, while assuming low-intelligence behaviors for the client system and the disruptions for the duration of the disruption attack. In reality, certain RDP problems could involve highly intelligent disruptions with advanced targeting capabilities (such as computer malware) and/or evolving behaviors (such as plant diseases) during the disruption attack, which necessitates the consideration of $\boldsymbol{S_D}$ and $\boldsymbol{S_D^t}$. Another example is the problem of propagating malware in mobile networks, in which the client system (the mobile phone users) may not be controlled, and certain client system decisions $\boldsymbol{S_C}$ and $\boldsymbol{S_C^t}$ (establishing connections, installing security updates) can both increase and decrease the disruption damage.
4. Limited characterization and classification of the CRDP interactions $\boldsymbol{C}$&$\boldsymbol{R}$, $\boldsymbol{R}$&$\boldsymbol{D}$, and $\boldsymbol{D}$&$\boldsymbol{C}$. Two important interactions are identified in this research: the disruption propagation direction in $\boldsymbol{D}$&$\boldsymbol{C}$ and the propagation-restraining effect in $\boldsymbol{R}$&$\boldsymbol{D}$. Further characterization of important interactions can overcome the research limitations of this dissertation and enrich future research on RDP problems.

The CRDP framework and the CLOC principle can be expanded in the following challenging research directions:

1. Adapting and applying the CRDP framework and the CLOC principle to specific RDP problems. This approach not only could improve the understanding of the specific RDP problems and the accompanying solution methods, but also could further enrich the CRDP framework and the CLOC principle.
2. Expanding the experiments to investigate the impacts of different types and sizes of complex networks. Although this research uses three random network models (Barabasi-Albert, Erdos-Renyi, and Watts-Strogatz) and two general network models (Grid Orthogonal and Grid Diagonal) for the experiment case studies, the network size is fixed at 100 nodes and the random network models' configuration and parameters are limited. Varying the network size and type can provide further insight into how network topology can influence the damages from disruption propagation and the effectiveness of responses.
3. Consideration of different types of disruptions. Different types of disruptions can be present at the same time, requiring different response resources. One example is in the case of agricultural plant disease, where the plants can be affected by two or more types of disease. These different types of disease can have different propagation mechanisms and directions. A possible approach is to apply multi-layer network modeling to represent the different disruption propagation mechanisms and directions.
4. Development of more advanced design/control analytics and protocols. Potential methodologies include:
    a. Statistical inference and Bayesian network to support problems with stochasticity.
    b. Game theory for the cases of adversarial intelligent decisions of $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{D}}$.
    c. Machine learning techniques for learning disruption propagation behavior.
5. Visualization of disruption propagation and response allocation to support decision making. Constructing a complex network per the CLOC first guideline can provide a convenient network map of the client system and how disruptions can propagate from one node to other nodes. Color-coding of the nodes can be added to describe the disruption and response status of the nodes, such as red for disrupted nodes, orange for imminent disruption propagations, green for pending responses, and blue for responded nodes. Visualization can support decision making in planning and allocating response decisions, and can also support the design of new analytics and protocols.

6. Investigation of more complex cases with confounding behaviors between $\boldsymbol{\mathcal{C}}$, $\boldsymbol{\mathcal{R}}$, and $\boldsymbol{\mathcal{D}}$. This means certain elements of the problem exhibit characteristics of $\boldsymbol{\mathcal{C}}$, $\boldsymbol{\mathcal{R}}$, and $\boldsymbol{\mathcal{D}}$ at the same time, and cannot be separated from one another. One such interesting case is the multiple agent path finding problem, also called the agent congestion problem. Examples of this problem include autonomous robot congestion in CPSs, traffic congestion in urban transportation, and agent congestion in computer games. This problem involves agents trying to find paths to reach their destinations, but the agents must occupy space, and can physically block each other from reaching their destinations. This problem has one unique characteristic: the agents involved exhibit characteristics of $\boldsymbol{\mathcal{C}}$ – being the victim, $\boldsymbol{\mathcal{R}}$ – being the rescuer, and $\boldsymbol{\mathcal{D}}$ – being the aggressor, at the same time. Each agent is negatively affected by the congestion (thus $\boldsymbol{\mathcal{C}}$), but is also part of the congestion due to occupying the space physically (thus $\boldsymbol{\mathcal{D}}$), and can also help resolve the congestion (thus $\boldsymbol{\mathcal{R}}$). Each agent's decision (movement) can help and/or worsen congestion.

APPENDIX A. TIE/CRDP SOFTWARE

The Teamwork Integration Evaluator (TIE) has been developed as a research software to simulate collaborative and interactive behaviors of distributed teams of agents, and to assess their performance in e-Work environments. Seven TIE software programs have been developed by PRISM Center (Production Robotics, and Integration Software for Manufacturing & Management) at Purdue University:

1. TIE 1.1 (Ceroni & Nof, 2002; Nguyen & Nof, 2019a): workflow integration, optimization, distributed parallel integration.
2. TIE/Agent (Khanna & Nof, 1994): agent-based manufacturing system, agent viability.
3. TIE/Protocol (Chin-Yin Huang & Nof, 2002): distributed resource allocation.
4. TIE/MEMS (Anussornnitisarn, Nof, & Etzion, 2005; Jeong & Nof, 2008): network communication, wireless sensor networks.
5. TIE/TAP (Y. Liu & Nof*, 2004; Ko, 2010): design of task administration protocol, task requirement analysis protocol, shared resource allocation protocol, synchronization and time-out protocol.
6. TIE/DLOC (Ko & Nof, 2010; Hao Zhong, 2016): simulation of propagating services in cyber-physical systems, activity-based scheduling.
7. TIE/CRDP (Hao Zhong & Nof, 2015): this work. The TIE/CRDP software is developed to simulate an RDP problem: the complex interactions between the client systems, response mechanisms, and disruption propagation.

TIE/CRDP is written in the object-oriented programming languages C# and has a planned version for Python 3.

1. C#, as of 2019, is a .NET compiled language with powerful computational capabilities on the Windows operating system (although slightly slower than C++ and Java). Its main drawback is the strict variable declaration and the more program structure, which leads to slower software implementation. This leads to a more challenging learning phase for new

engineering researchers, as coding in C# requires a more disciplined software engineering approach to ensure robust programming.

2. Python 3, as of 2019, is a good programming language for engineering research. Its dynamic structure and simplified variable declaration allow fast, yet brief and elegant, software implementation. It is good for researchers new to programming, and is very popular for research in engineering, mathematics, statistics, and computer science, which leads to the high availability of research libraries. Its main shortcoming is the low computational power, due to the programming language being an interpreted programming language. Python can be expected to be at least 10 times slower than C# or other compiled languages, and up to 100 or 200 times slower in certain cases.
3. A hybrid approach is also a possibility. In practice, industry and corporate researchers often employ this approach. Rapid prototyping and initial testing of each component (i.e. algorithm, subsystems) would be programmed on Python or MATLAB or another convenient programming language. The software optimization and integration would be programmed on C++ or C# to achieve much higher computational efficiency.

An abstract view of the TIE/CRDP software is provided in Figure A.1.

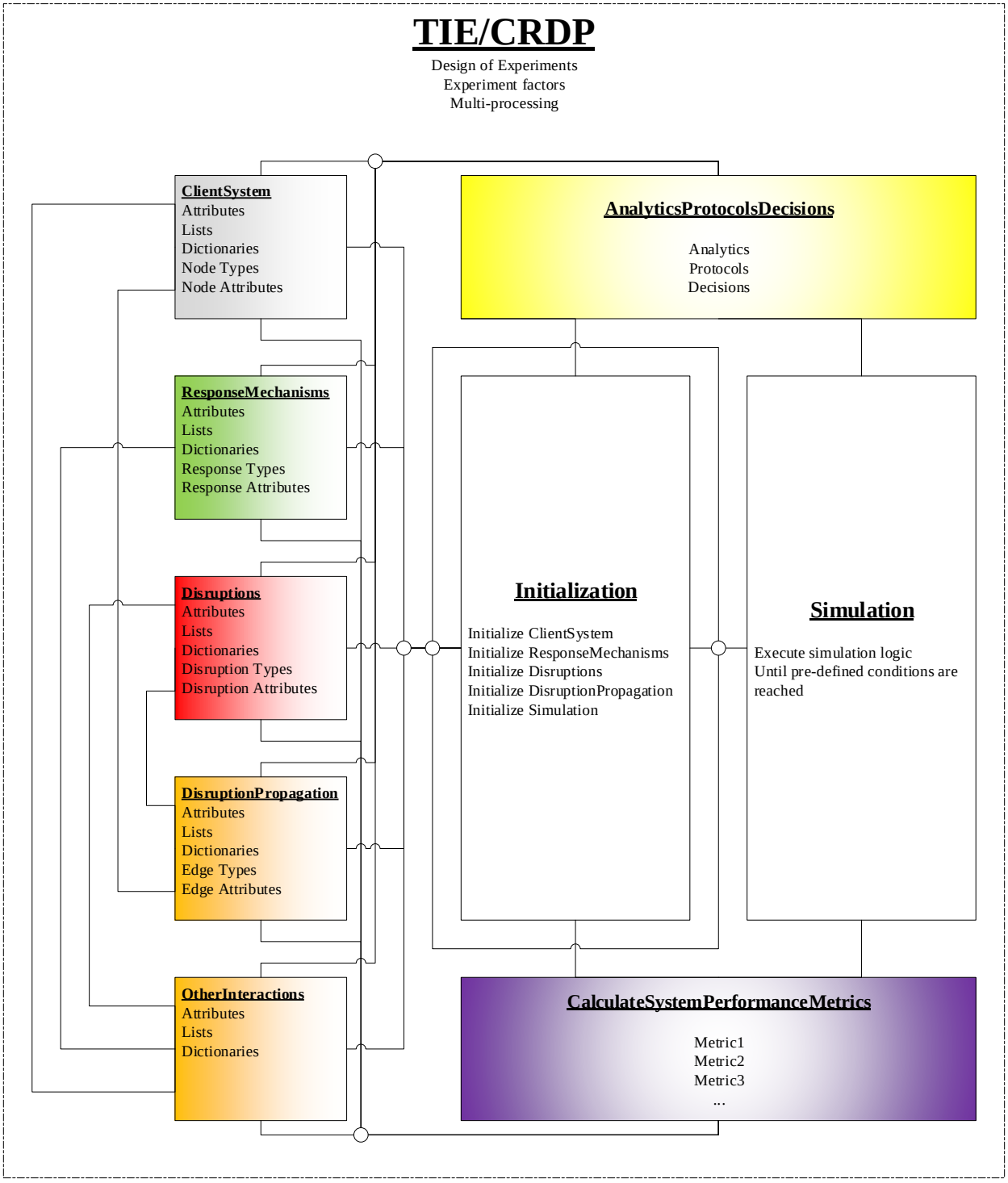

Figure A.1. TIE/CRDP software abstract view

The TIE_CRDP class:

The central controller of the software. The TIE class manages the design and execution of the different experiments. The factors of the experiments are implemented in this class. Multi-processing of different experiments is also implemented in this class.

The ClientSystem programming elements:

Include the variables, collections, and objects (together with their attributes) pertaining to the client system $\boldsymbol{\mathcal{C}}$. Each node type of the client system is defined here as a class of objects. The attributes of the nodes are either defined within the classes or as separate dictionaries.

The ResponseMechanisms programming elements:

Include the variables, collections, and objects (together with their attributes) pertaining to the response mechanisms $\boldsymbol{\mathcal{R}}$. Each response type is defined here as a class of objects. The attributes of the response mechanisms are either defined within the classes or as separate dictionaries.

The DisruptionPropagation programming elements:

Include the variables, collections, and objects (together with their attributes) pertaining to the disruptions $\boldsymbol{\mathcal{D}}$. Include the variables, collections, and objects (together with their attributes) pertaining to the disruptions $\boldsymbol{\mathcal{D}}$. Each disruption type is defined here as a class of objects. The attributes of the disruptions are either defined within the classes or as separate dictionaries.

The DisruptionPropagationDirections programming elements:

Include the variables, collections, and objects (together with their attributes) pertaining to the disruption propagation directions. Each potential disruption propagation direction is defined as an edge object. The attributes of the edges are either defined within the classes or as separate dictionaries.

The Interactions programming elements:

Other interactions (besides the edges) are defined here.

The Initialization procedure/method:

This procedure sets the initial state of the simulation based on the defined classes and variables. This includes the initialization of the client system, the response mechanisms, the initial disruptions, the potential disruption propagation directions, the other interactions, and the initial simulation parameters.

The Simulation procedure/method:

This procedure executes the simulation logic until the pre-defined conditions are met. Such conditions include maximum simulation time, removed disruptions, etc. This procedure can be divided into sub-procedures to better organize the software structure.

The AnalyticsProtocolsDecisions procedure/method:

This procedure computes the relevant analytics $\boldsymbol{\mathcal{A}}$, executes the selected protocols $\boldsymbol{\mathcal{P}}$, and sets the decisions $\boldsymbol{\mathcal{S}}_{\#}$ and $\boldsymbol{\mathcal{S}}_{\#}^{t}$. This procedure can be called by the Initialization procedure and/or the Simulation procedure. This procedure can be divided into sub-procedures to better organize the software structure.

The CalculatePerformanceMetrics programming elements and methods/procedures:

Include and keep track of system performance metrics $\boldsymbol{\mathcal{M}}$. The procedure(s) calculate the system performance metrics $\boldsymbol{\mathcal{M}}$ and can be called by the Initialization procedure and/or the Simulation procedure. This procedure can be divided into sub-procedures to better organize the software structure.

The TIE/CRDP software is adapted to simulate the three case studies discussed in this dissertation:

Case 1 – TIE/CRDP/CDUD

The TIE_CRDP class is adapted in accordance with Section 4.4. The ClientSystem, ResponseMechanism, DisruptionPropagation, and DisruptionPropagationDirections elements are adapted in accordance with Section 4.2. The Initialization, Simulation, and

CalculatePerformanceMetrics elements are adapted in accordance with Section 4.2. The AnalyticsProtocolsDecisions elements are adapted in accordance with Section 4.3.

Case 2 – TIE/CRDP/CSPD

The TIE_CRDP class is adapted in accordance with Section 5.4. The ClientSystem, ResponseMechanism, DisruptionPropagation, and DisruptionPropagationDirections elements are adapted in accordance with Section 5.2. The Initialization, Simulation, and CalculatePerformanceMetrics elements are adapted in accordance with Section 5.2. The AnalyticsProtocolsDecisions elements are adapted in accordance with Section 5.3.

Case 3 – TIE/CRDP/CTCD

The TIE_CRDP class is adapted in accordance with Section 6.4. The ClientSystem, ResponseMechanism, DisruptionPropagation, and DisruptionPropagationDirections elements are adapted in accordance with Section 6.2. The Initialization, Simulation, and CalculatePerformanceMetrics elements are adapted in accordance with Section 6.2. The AnalyticsProtocolsDecisions elements are adapted in accordance with Section 6.3.

REFERENCES

Ahmad, M. A., Woodhead, S., & Gan, D. (2016). *A Safeguard against Fast Self-propagating Malware*. Paper presented at the Proceedings of the 6th International Conference on Communication and Network Security, Singapore, Singapore.

Ahrens, M. (2011). Local Fire Department Responses to Wildfires in the US: National Estimates Based on 2004-08 Data. *Fire Safety Science, 10*, 1389-1400. doi:10.3801/IAFFS.FSS.10-1389

Albert, R., & Barabasi, A. L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics, 74*(1), 47-97. doi:DOI 10.1103/RevModPhys.74.47

Albert, R., Jeong, H., & Barabási, A.-L. (2000). Error and attack tolerance of complex networks. *Nature, 406*, 378-382.

Albert, R., Jeong, H., & Barabasi, A. L. (2000). Error and attack tolerance of complex networks. *Nature, 406*(6794), 378-382. doi:10.1038/35019019

Anantha Raj, P., & Srivani, M. (2018). *Internet of Robotic Things Based Autonomous Fire Fighting Mobile Robot.* Paper presented at the 9th IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2018, December 13, 2018 - December 15, 2018, Madurai, India.

Anussornnitisarn, P., Nof, S. Y., & Etzion, O. (2005). Decentralized control of cooperative and autonomous agents for solving the distributed resource allocation problem. *International Journal of Production Economics, 98*(2), 114-128.

Arora, V., & Ventresca, M. (2017). Action-based Modeling of Complex Networks. *Scientific Reports, 7*(1), 6673. doi:10.1038/s41598-017-05444-4

Arora, V., & Ventresca, M. (2018). Modeling topologically resilient supply chain networks. *Applied Network Science, 3*(1), 19. doi:10.1007/s41109-018-0070-7

Azad, N., Davoudpour, H., Saharidis, G. K. D., & Shiripour, M. (2014). A new model to mitigating random disruption risks of facility and transportation in supply chain network design. *The International Journal of Advanced Manufacturing Technology, 70*(9), 1757-1774. doi:10.1007/s00170-013-5404-0

Barabasi, A. L., & Albert, R. (1999). Emergence of scaling in random networks. *Science, 286*(5439), 509-512. doi:10.1126/science.286.5439.509

Basole, R. C. (2016). Topological analysis and visualization of interfirm collaboration networks in the electronics industry. *Decision Support Systems, 83*, 22-31. doi:https://doi.org/10.1016/j.dss.2015.12.005

Basole, R. C., & Bellamy, M. A. (2014). Visual analysis of supply network risks: Insights from the electronics industry. *Decision Support Systems, 67*, 109-120. doi:https://doi.org/10.1016/j.dss.2014.08.008

Bate, A. M., Jones, G., Kleczkowski, A., MacLeod, A., Naylor, R., Timmis, J., . . . White, P. C. L. (2016). Modelling the impact and control of an infectious disease in a plant nursery with infected plant material inputs. *Ecological Modelling, 334*, 27-43. doi:https://doi.org/10.1016/j.ecolmodel.2016.04.013

Batista, F. K., Martín del Rey, Á., & Queiruga-Dios, A. (2018). *Malware Propagation Software for Wireless Sensor Networks*, Cham.

Bavelas, A. (1950). Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, *22*(6), 725-730.

Behdani, B., Dam, K. H. v., & Lukszo, Z. (2011, 11-13 April 2011). *Agent-based modeling for disruption management in industrial networks and supply chains.* Paper presented at the 2011 International Conference on Networking, Sensing and Control.

Bhargava, R., Levalle, R. R., & Nof, S. Y. (2016). A best-matching protocol for order fulfillment in re-configurable supply networks. *Computers in Industry, 82*, 160-169. doi:10.1016/j.compind.2016.07.001

Buldyrev, S. V., Parshani, R., Paul, G., Stanley, H. E., & Havlin, S. (2010). Catastrophic cascade of failures in interdependent networks. *Nature, 464*(7291), 1025-1028. doi:10.1038/nature08932

Buzna, L., Peters, K., Ammoser, H., Kuhnert, C., & Helbing, D. (2007). Efficient response to cascading disaster spreading. *Phys Rev E Stat Nonlin Soft Matter Phys*, *75*(5 Pt 2), 056107. doi:10.1103/PhysRevE.75.056107

Caton, S. E., Hakes, R. S. P., Gorham, D. J., Zhou, A., & Gollner, M. J. (2017). Review of Pathways for Building Fire Spread in the Wildland Urban Interface Part I: Exposure Conditions. *Fire Technology*, *53*(2), 429-473. doi:10.1007/s10694-016-0589-z

Cencerrado, A., Cortés, A., & Margalef, T. (2014). Response time assessment in forest fire spread simulation: An integrated methodology for efficient exploitation of available prediction time. *Environmental Modelling & Software, 54*, 153-164. doi:https://doi.org/10.1016/j.envsoft.2014.01.008

Ceroni, J. A., & Nof, S. Y. (2002). A workflow model based on parallelism for distributed organizations. *Journal of Intelligent Manufacturing, 13*(6), 439-461.

Chaoqi, F., Ying, W., Kun, Z., & Yangjun, G. (2018). Complex networks under dynamic repair model. *Physica A: Statistical Mechanics and its Applications, 430*, 323-330. doi:10.1016/j.physa.2017.08.071

Chaoqi, F., Ying, W., & Xiaoyang, W. (2017). Research on complex networks' repairing characteristics due to cascading failure. *Physica A: Statistical Mechanics and its Applications, 482*, 317-324. doi:10.1016/j.physa.2017.04.086

Chaoqi, F., Ying, W., Yangjun, G., & Xiaoyang, W. (2017). Complex networks repair strategies: Dynamic models. *Physica A: Statistical Mechanics and its Applications, 482*, 401-406. doi:10.1016/j.physa.2017.04.118

Chen, T. D., & Kockelman, K. M. (2016). Management of a shared autonomous electric vehicle fleet: Implications of pricing schemes. *Transportation Research Record, 2572*, 37-46. doi:10.3141/2572-05

Chen, X. W., & Nof, S. Y. (2012). Agent-based error prevention algorithms. *Expert Systems With Applications, 39*(1), 280-287. doi:https://doi.org/10.1016/j.eswa.2011.07.018

Chen, X. W., & Nof, S. Y. (2012). Conflict and error prevention and detection in complex networks. *Automatica, 48*(5), 770-778. doi:10.1016/j.automatica.2012.02.030

Chen, X. W., & Nof, S. Y. (2012). Constraint-based conflict and error management. *Engineering Optimization, 44*(7), 821-841. doi:10.1080/0305215X.2011.613466

Cheng, S., Ao, W. C., Chen, P., & Chen, K. (2011). On Modeling Malware Propagation in Generalized Social Networks. *IEEE Communications Letters, 15*(1), 25-27. doi:10.1109/LCOMM.2010.01.100830

Chozick, A. (2007). A key strategy of Japan's car makers backfires. *Wall Street Journal, 20*, B1.

Cohen, W. W. (2005). Enron email dataset. Retrieved from http://www.cs.cmu.edu/~enron/

Cooke, E., Mao, Z. M., & Jahanian, F. (2006, 25-28 June 2006). *Hotspots: The Root Causes of Non-Uniformity in Self-Propagating Malware.* Paper presented at the International Conference on Dependable Systems and Networks (DSN'06).

Crucitti, P., Latora, V., & Marchiori, M. (2004). Model for cascading failures in complex networks. *Phys Rev E Stat Nonlin Soft Matter Phys, 69*(4 Pt 2), 045104. doi:10.1103/PhysRevE.69.045104

Datta, P. (2017). Supply network resilience: a systematic literature review and future research. *International Journal of Logistics Management, 28*(4), 1387-1424. doi:10.1108/IJLM-03-2016-0064

Day, J. M. (2014). Fostering emergent resilience: the complex adaptive supply network of disaster relief. *International Journal of Production Research, 52*(7), 1970-1988. doi:10.1080/00207543.2013.787496

del Rey, A. M., Guillén, J. D. H., & Sánchez, G. R. (2016). *Modeling Malware Propagation in Wireless Sensor Networks with Individual-Based Models*, Cham.

Diabat, A., Jabbarzadeh, A., & Khosrojerdi, A. (2019). A perishable product supply chain network design problem with reliability and disruption considerations. *International Journal of Production Economics, 212*, 125-138. doi:https://doi.org/10.1016/j.ijpe.2018.09.018

Diggle, A., Salam, M., & Monjardino, M. (2006). Individual-Based Models of the Spread of Disease, Weeds, and Insects. *The Mathematica Journal, 10*. doi:10.3888/tmj.10.2-8

Dixit, V., Seshadrinath, N., & Tiwari, M. K. (2016). Performance measures based optimization of supply chain network resilience: A NSGA-II+Co-Kriging approach. *Computers & Industrial Engineering, 93*, 205-214. doi:https://doi.org/10.1016/j.cie.2015.12.029

Donatelli, M., Magarey, R. D., Bregaglio, S., Willocquet, L., Whish, J. P. M., & Savary, S. (2017). Modelling the impacts of pests and diseases on agricultural systems. *Agricultural Systems, 155*, 213-224. doi:https://doi.org/10.1016/j.agsy.2017.01.019

Ducrot, A., & Matano, H. (2016, 2016//). *Plant Disease Propagation in a Striped Periodic Medium.* Paper presented at the Applied Analysis in Biological and Physical Sciences, New Delhi.

Dusadeerungsikul, P. O., & Nof, S. Y. (2019). A collaborative control protocol for agricultural robot routing with online adaptation. *Computers & Industrial Engineering, 135*, 456-466. doi:https://doi.org/10.1016/j.cie.2019.06.037

Dusadeerungsikul, P. O., Nof, S. Y., & Bechar, A. (2018). *Detecting stresses in crops early by collaborative robot-sensors-human system automation.* Paper presented at the 2018 Institute of Industrial and Systems Engineers Annual Conference and Expo, IISE 2018, May 19, 2018 - May 22, 2018, Orlando, FL, United states.

Eder-Neuhauser, P., Zseby, T., & Fabini, J. (2018). Malware propagation in smart grid monocultures. *e & i Elektrotechnik und Informationstechnik, 135*(3), 264-269. doi:10.1007/s00502-018-0616-5

Erdös, P., & Rényi, A. (1959). On random graphs, I. *Publicationes Mathematicae (Debrecen), 6*, 290-297. doi:citeulike-article-id:4012374

Estrada, E., Meloni, S., Sheerin, M., & Moreno, Y. (2016). Epidemic spreading in random rectangular networks. *Physical review E, 94*(5), 052316. doi:10.1103/PhysRevE.94.052316

Faghani, M. R., & Saidi, H. (2009, 13-14 Oct. 2009). *Malware propagation in Online Social Networks.* Paper presented at the 2009 4th International Conference on Malicious and Unwanted Software (MALWARE).

Fattahi, M., Govindan, K., & Keyvanshokooh, E. (2017). Responsive and resilient supply chain network design under operational and disruption risks with delivery lead-time sensitive customers. *Transportation Research Part E: Logistics and Transportation Review, 101*, 176-200. doi:https://doi.org/10.1016/j.tre.2017.02.004

Ferialdy, A. (2016). *Graph theoretical analysis of the Dynamic Lines of Collaboration model for disruption response*: Ann Arbor : ProQuest Dissertations & Theses.

Firmansyah, M. R., & Amer, Y. (2013). A review of collaborative manufacturing network models. *Int. J. Mater. Mech. Manufact, 1*(1), 6-12.

Floderus, P., Lingas, A., & Persson, M. (2013). Towards more efficient infection and fire fighting. *International Journal of Foundations of Computer Science, 24*(1), 3-14. doi:10.1142/S0129054113400017

Floyd, R. W. (1962). Algorithm 97: Shortest path. *Commun. ACM, 5*(6), 345. doi:10.1145/367766.368168

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks, 1*(3), 215-239.

Fu, N., Wang, C., & Ji, X. (2019, 4-7 Aug. 2019). *Study on Visual Detection Device of Plant Leaf Disease.* Paper presented at the 2019 IEEE International Conference on Mechatronics and Automation (ICMA).

Georgoudas, I., Sirakoulis, G., & a, I. A. (2007, 29-31 Oct. 2007). *An Intelligent Cellular Automaton Model for Crowd Evacuation in Fire Spreading Conditions.* Paper presented at the 19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007).

Ghavamifar, A., Makui, A., & Taleizadeh, A. A. (2018). Designing a resilient competitive supply chain network under disruption risks: A real-world application. *Transportation Research Part E: Logistics and Transportation Review, 115*, 87-109. doi:https://doi.org/10.1016/j.tre.2018.04.014

Giachetti, B., Couton, D., & Plourde, F. (2017). Smoke spreading analyses in a subway fire scale model. *Tunnelling and Underground Space Technology, 70*, 233-239. doi:https://doi.org/10.1016/j.tust.2017.08.008

Gibson, G. J., Otten, W., N. Filipe, J. A., Cook, A., Marion, G., & Gilligan, C. A. (2006). Bayesian estimation for percolation models of disease spread in plant populations. *Statistics and Computing, 16*(4), 391-402. doi:10.1007/s11222-006-0019-z

Gilligan, C. A. (2008). Sustainable agriculture and plant diseases: an epidemiological perspective. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences, 363*(1492), 741-759. doi:10.1098/rstb.2007.2181

Gong, J., Mitchell, J. E., Krishnamurthy, A., & Wallace, W. A. (2014). An interdependent layered network model for a resilient supply chain. *Omega, 46*, 104-116. doi:10.1016/j.omega.2013.08.002

Gu, X., Jin, X., & Ni, J. (2014). Resilience Measures of Manufacturing Systems Under Disruptions. (45806), V001T004A007. doi:10.1115/MSEC2014-4047

Guariniello, C., & DeLaurentis, D. (2017). Supporting design via the system operational dependency analysis methodology. *Research in Engineering Design, 28*(1), 53-69. doi:10.1007/s00163-016-0229-0

Guo, H., Cheng, H. K., & Kelley, K. (2016). Impact of Network Structure on Malware Propagation: A Growth Curve Perspective. *Journal of Management Information Systems, 33*(1), 296-325. doi:10.1080/07421222.2016.1172440

Guo, P., Dusadeerungsikul, P. O., & Nof, S. Y. (2018). Agricultural cyber physical system collaboration for greenhouse stress management. *Computers and Electronics in Agriculture, 150*, 439-454. doi:https://doi.org/10.1016/j.compag.2018.05.022

Han, J., & Shin, K. (2016). Evaluation mechanism for structural robustness of supply chain considering disruption propagation. *International Journal of Production Research, 54*(1), 135-151. doi:10.1080/00207543.2015.1047977

Hasani, A., & Khosrojerdi, A. (2016). Robust global supply chain network design under disruption and uncertainty considering resilience strategies: A parallel memetic algorithm for a real-life case study. *Transportation Research Part E: Logistics and Transportation Review, 87*, 20-52. doi:https://doi.org/10.1016/j.tre.2015.12.009

He, F., He, R. C., Sun, S. N., & Chen, J. D. (2014). Research on the Model of Emergency Fire Rescue Vehicle Routing Selection and Resource Allocation. *Applied Mechanics and Materials, 641-642*, 824-828. doi:10.4028/www.scientific.net/AMM.641-642.824

He, L., Fan, M., Liu, B., Chen, S., & Li, H. (2014, 15-16 June 2014). *Fire Control Emergency Rescue Resources Allocation Based on Calamities and Accidents Risk Evaluation.* Paper presented at the 2014 Fifth International Conference on Intelligent Systems Design and Engineering Applications.

Hernández Encinas, L., Hoya White, S., Martín del Rey, A., & Rodríguez Sánchez, G. (2007). Modelling forest fire spread using hexagonal cellular automata. *Applied Mathematical Modelling, 31*(6), 1213-1227. doi:https://doi.org/10.1016/j.apm.2006.04.001

Himoto, K., & Tanaka, T. (2012). *A model for the fire-fighting activity of local residents in urban fires*.

Huang, C.-Y., Cheng, K., & Holt, A. (2007). An integrated manufacturing network management framework by using mobile agent. *The International Journal of Advanced Manufacturing Technology, 32*(7), 822-833. doi:10.1007/s00170-005-0378-1

Huang, C.-Y., Holt, A., Monk, J., & Cheng, K. (2007). The application of dependency management in an integrated manufacturing network framework. *The International Journal of Advanced Manufacturing Technology, 33*(3-4), 354-364.

Huang, C.-Y., & Nof, S. Y. (2002). Evaluation of agent-based manufacturing systems based on a parallel simulator. *Computers & Industrial Engineering, 43*(3), 529-552.

Ingram, J. (2011). A food systems approach to researching food security and its interactions with global environmental change. *Food Security, 3*. doi:10.1007/s12571-011-0149-9

Ismail, H. S., Poolton, J., & Sharifi, H. (2011). The role of agile strategic capabilities in achieving resilience in manufacturing-based small companies. *International Journal of Production Research, 49*(18), 5469-5487.

Jabbarzadeh, A., Haughton, M., & Khosrojerdi, A. (2018). Closed-loop supply chain network design under disruption risks: A robust approach with real world application. *Computers & Industrial Engineering, 116*, 178-191. doi:https://doi.org/10.1016/j.cie.2017.12.025

Jeong, W., & Nof, S. Y. (2008). Performance evaluation of wireless sensor network protocols for industrial applications. *Journal of Intelligent Manufacturing, 19*(3), 335-345.

Jia, P., Liu, J., Fang, Y., Liu, L., & Liu, L. (2018). Modeling and analyzing malware propagation in social networks with heterogeneous infection rates. *Physica A: Statistical Mechanics and its Applications, 507*, 240-254. doi:https://doi.org/10.1016/j.physa.2018.05.047

Jiao, Y., Wang, J., Xiao, M., Xu, T., & Chen, W. (2014, 25-26 Oct. 2014). *Development of Field-Zone-Net Model for Fire Smoke Propagation Simulation in Ships.* Paper presented at the 2014 7th International Conference on Intelligent Computation Technology and Automation.

Khanna, N., & Nof, S. Y. (1994). *TIE: Teamwork Integration Evaluation Simulator, a Preliminary User Manual for TIE 1.1*: School of Industrial Engineering, Purdue University.

Kim, J., Dietz, J. E., & Matson, E. T. (2016, 10-11 May 2016). *Simulation modeling of a statistical fire spread to respond fire accident in buildings.* Paper presented at the 2016 IEEE Symposium on Technologies for Homeland Security (HST).

Kim, Y., Chen, Y.-S., & Linderman, K. (2015). Supply network disruption and resilience: A network structural perspective. *Journal of Operations Management, 33-34*, 43-59. doi:10.1016/j.jom.2014.10.006

Ko, H. S. (2010). *Design of protocols for task administration in collaborative e-work systems.* Purdue University,

Ko, H. S., & Nof, S. Y. (2010). Design of protocols for task administration in collaborative production systems. *International Journal of Computers Communications & Control, 5*(1), 91-105.

Landegren, F. E., Johansson, J., & Samuelsson, O. (2016). A Method for Assessing Margin and Sensitivity of Electricity Networks With Respect to Repair System Resources. *IEEE Transactions on Smart Grid, 7*(6), 2880-2889. doi:10.1109/Tsg.2016.2582080

Liu, B., Zhou, W., Gao, L., Zhou, H., Luan, T. H., & Wen, S. (2018). Malware Propagations in Wireless Ad Hoc Networks. *IEEE Transactions on Dependable and Secure Computing, 15*(6), 1016-1026. doi:10.1109/TDSC.2016.2642191

Liu, F. L., & Wang, L. J. (2012). Study on Task Allocation Model of Forest Fire Fighting. *Advanced Materials Research, 457-458*, 1129-1136. doi:10.4028/www.scientific.net/AMR.457-458.1129

Liu, W., Liu, C., Yang, Z., Liu, X., Zhang, Y., & Wei, Z. (2016). Modeling the propagation of mobile malware on complex networks. *Communications in Nonlinear Science and Numerical Simulation, 37*, 249-264. doi:https://doi.org/10.1016/j.cnsns.2016.01.019

Liu, W., & Zhong, S. (2018). Modeling and analyzing the dynamic spreading of epidemic malware by a network eigenvalue method. *Applied Mathematical Modelling, 63*, 491-507. doi:https://doi.org/10.1016/j.apm.2018.07.010

Liu, W. P., Liu, C., Yang, Z., Liu, X. Y., Zhang, Y. H., & Wei, Z. X. (2016). Modeling the propagation of mobile malware on complex networks. *Communications in Nonlinear Science and Numerical Simulation, 37*, 249-264. doi:10.1016/j.cnsns.2016.01.019

Liu, Y., & Nof*, S. (2004). Distributed microflow sensor arrays and networks: Design of architectures and communication protocols. *International Journal of Production Research, 42*(15), 3101-3115.

Lu, J., Guo, J., Jian, Z., & Xu, X. (2018). Optimal Allocation of Fire Extinguishing Equipment for a Power Grid Under Widespread Fire Disasters. *IEEE Access, 6*, 6382-6389. doi:10.1109/ACCESS.2017.2788893

Manes, M., & Rush, D. (2019). A Critical Evaluation of BS PD 7974-7 Structural Fire Response Data Based on USA Fire Statistics. *Fire Technology, 55*(4), 1243-1293. doi:10.1007/s10694-018-0775-2

Marchiori, M., & Latora, V. (2000). Harmony in the small-world. *Physica A: Statistical Mechanics and its Applications, 285*(3), 539-546. doi:https://doi.org/10.1016/S0378-4371(00)00311-3

Mari, S. I., Young Hae, L., & Memon, M. S. (2014). Sustainable and Resilient Supply Chain Network Design under Disruption Risks. *Sustainability, 6*(10), 6666-6686. doi:10.3390/su6106666

Moghaddam, M., & Nof, S. Y. (2014). Combined demand and capacity sharing with best matching decisions in enterprise collaboration. *International Journal of Production Economics, 148*, 93-109. doi:10.1016/j.ijpe.2013.11.015

Moghaddam, M., & Nof, S. Y. (2015). Real-time administration of tool sharing and best matching to enhance assembly lines balanceability and flexibility. *Mechatronics, 31*, 147-157. doi:10.1016/j.mechatronics.2014.10.001

Moghaddam, M., & Nof, S. Y. (2016). *Best Matching Theory & Applications*. In *Automation, Collaboration, & E-Services*, (pp. 1 online resource (XVI, 231 p. 257 illus., 248 illus. in color.)).

Mohapatra, P., Nanda, S., & Adhikari, T. (2015, 9-10 Jan. 2015). *Resilience measurement of a global supply chain network.* Paper presented at the 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO).

Motter, A. E., & Lai, Y. C. (2002). Cascade-based attacks on complex networks. *Phys Rev E Stat Nonlin Soft Matter Phys, 66*(6 Pt 2), 065102. doi:10.1103/PhysRevE.66.065102

Musa, A., Almohannadi, H., & Alhamar, J. (2018, 6-8 Aug. 2018). *Malware Propagation Modelling in Peer-to-Peer Networks: A Review.* Paper presented at the 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW).

Nair, A., & Vidal, J. M. (2011). Supply network topology and robustness against disruptions – an investigation using multi-agent model. *International Journal of Production Research, 49*(5), 1391-1404.

Nguyen, W. P. V., Nair, A. S., & Nof, S. Y. (2019). *Advancing Cyber-Physical Systems Resilience: The Effects of Evolving Disruptions* Paper presented at the ICPR-25, Chicago, IL.

Nguyen, W. P. V., & Nof, S. Y. (2018). Resilience Informatics for Cyber-augmented Manufacturing Networks (CMN): Centrality, Flow and Disruption. *Studies in Informatics and Control, 27*(4), 377-384. doi:https://doi.org/10.24846/v27i4y201801

Nguyen, W. P. V., & Nof, S. Y. (2019a). Collaborative response to disruption propagation (CRDP) in cyber-physical systems and complex networks. *Decision Support Systems, 117*, 1-13. doi:https://doi.org/10.1016/j.dss.2018.11.005

Nguyen, W. P. V., & Nof, S. Y. (2019b). *Collaborative Response to Disruption Propagation with Established Lines of Collaboration (CRDP/ESLOC) in Cyber-Physical Systems: Informatics for Decision Support.* Paper presented at the ICPR-25, Chicago, IL.

Nof, S. Y. (2007). Collaborative control theory for e-Work, e-Production, and e-Service. *Annual Reviews in Control, 31*(2), 281-292. doi:10.1016/j.arcontrol.2007.08.002

Nof, S. Y. (2013). Sustainability and resiliency in supply networks. *Plenary Talk—14th Asia Pacific Industrial Engineering and Management Society*.

Nof, S. Y., Ceroni, J., Jeong, W., & Moghaddam, M. (2015). *Revolutionizing Collaboration through e-Work, e-Business, and e-Service*: Berlin, Heidelberg : Springer Berlin Heidelberg : Imprint: Springer.

Ocampo, A. L. P. d., & Dadios, E. P. (2018, 29 Nov.-2 Dec. 2018). *Mobile Platform Implementation of Lightweight Neural Network Model for Plant Disease Detection and Recognition.* Paper presented at the 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM).

Orozco-Fuentes, S., Griffiths, G., Holmes, M. J., Ettelaie, R., Smith, J., Baggaley, A. W., & Parker, N. G. (2019). Early warning signals in plant disease outbreaks. *Ecological Modelling, 393*, 12-19. doi:https://doi.org/10.1016/j.ecolmodel.2018.11.003

Osorio, A. F., Fernandez-Pello, C., Urban, D. L., & Ruff, G. A. (2013). Limiting conditions for flame spread in fire resistant fabrics. *Proceedings of the Combustion Institute, 34*(2), 2691-2697. doi:https://doi.org/10.1016/j.proci.2012.07.053

Parajuli, A., Kuzgunkaya, O., & Vidyarthi, N. (2017). Responsive contingency planning of capacitated supply networks under disruption risks. *Transportation Research Part E: Logistics and Transportation Review, 102*, 13-37. doi:https://doi.org/10.1016/j.tre.2017.03.010

Park, Y., Nicol, D. M., Zhu, H., & Lee, C. W. (2013, 21-24 Oct. 2013). *Prevention of malware propagation in AMI.* Paper presented at the 2013 IEEE International Conference on Smart Grid Communications (SmartGridComm).

Paul, S. K., Sarker, R., & Essam, D. (2017). A quantitative model for disruption mitigation in a supply chain. *European Journal of Operational Research, 257*(3), 881-895. doi:https://doi.org/10.1016/j.ejor.2016.08.035

Peng, S., Wang, G., & Yu, S. (2013, 16-18 July 2013). *Modeling Malware Propagation in Smartphone Social Networks.* Paper presented at the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications.

Peng, W., Feng, C., Che, A., & MengChu, Z. (2018). Bi-objective scheduling of fire engines for fighting forest fires: new optimization approaches. *IEEE Transactions on Intelligent Transportation Systems, 19*(4), 1140-1151. doi:10.1109/TITS.2017.2717188

Piraveenan, M., Prokopenko, M., & Hossain, L. (2013). Percolation centrality: Quantifying graph-theoretic impact of nodes during percolation in networks. *PloS one, 8*(1), e53095.

Ponnambalam, L., Wenbin, L., Fu, X., Yin, X. F., Wang, Z., & Goh, R. S. M. (2013, 10-13 Dec. 2013). *Decision trees to model the impact of disruption and recovery in supply chain networks.* Paper presented at the 2013 IEEE International Conference on Industrial Engineering and Engineering Management.

Queiruga-Dios, A., Hernández Encinas, A., Martín-Vaquero, J., & Hernández Encinas, L. (2017, 2017//). *Malware Propagation Models in Wireless Sensor Networks: A Review.* Paper presented at the International Joint Conference SOCO'16-CISIS'16-ICEUTE'16, Cham.

Reyes Levalle, R. (2018). *Resilience by Teaming in Supply Chains and Networks*: Springer.

Reyes Levalle, R., & Nof, S. Y. (2015a). A resilience by teaming framework for collaborative supply networks. *Computers & Industrial Engineering, 90*, 67-85. doi:10.1016/j.cie.2015.08.017

Reyes Levalle, R., & Nof, S. Y. (2015b). Resilience by teaming in supply network formation and re-configuration. *International Journal of Production Economics, 160*, 80-93. doi:10.1016/j.ijpe.2014.09.036

Reyes Levalle, R., & Nof, S. Y. (2017). Resilience in supply networks: Definition, dimensions, and levels. *Annual Reviews in Control, 43*, 224-236. doi:10.1016/j.arcontro1.2017.02.003

Rossi, R., & Ahmed, N. (2015). *The network data repository with interactive graph analytics and visualization.* Paper presented at the Twenty-Ninth AAAI Conference on Artificial Intelligence.

Sajadi, S. M., Esfahani, M. M. S., & Sorensen, K. (2011). Production control in a failure-prone manufacturing network using discrete event simulation and automated response surface methodology. *International Journal of Advanced Manufacturing Technology, 53*(1-4), 35-46. doi:10.1007/s00170-010-2814-0

Savary, S., Ficke, A., Aubertot, J.-N., & Hollier, C. (2012). Crop losses due to diseases and their implications for global food production losses and food security. *Food Security, 4*. doi:10.1007/s12571-012-0200-5

Sawik, T. (2019). Two-period vs. multi-period model for supply chain disruption management. *International Journal of Production Research, 57*(14), 4502-4518. doi:10.1080/00207543.2018.1504246

Scheibe, K. P., & Blackhurst, J. (2018). Supply chain disruption propagation: a systemic risk and normal accident theory perspective. *International Journal of Production Research, 56*(1-2), 43-59. doi:10.1080/00207543.2017.1355123

Schmitt, A. J., & Singh, M. (2012). A quantitative analysis of disruption risk in a multi-echelon supply chain. *International Journal of Production Economics, 139*(1), 22-32. doi:https://doi.org/10.1016/j.ijpe.2012.01.004

Schor, N., Bechar, A., Ignat, T., Dombrovsky, A., Elad, Y., & Berman, S. (2016). Robotic Disease Detection in Greenhouses: Combined Detection of Powdery Mildew and Tomato Spotted Wilt Virus. *IEEE Robotics and Automation Letters, 1*(1), 354-360. doi:10.1109/LRA.2016.2518214

Seok, H., Kim, K., & Nof, S. Y. (2016). Intelligent contingent multi-sourcing model for resilient supply networks. *Expert Systems With Applications, 51*, 107-119. doi:10.1016/j.eswa.2015.12.026

Shah, N., Shah, H., Malensek, M., Pallickara, S. L., & Pallickara, S. (2016, 5-8 Dec. 2016). *Network analysis for identifying and characterizing disease outbreak influence from voluminous epidemiology data.* Paper presented at the 2016 IEEE International Conference on Big Data (Big Data).

Shanmugam, L., Adline, A. L. A., Aishwarya, N., & Krithika, G. (2017, 7-8 April 2017). *Disease detection in crops using remote sensing images.* Paper presented at the 2017 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR).

Shen, S., Li, H., Han, R., Vasilakos, A. V., Wang, Y., & Cao, Q. (2014). Differential Game-Based Strategies for Preventing Malware Propagation in Wireless Sensor Networks. *IEEE Transactions on Information Forensics and Security, 9*(11), 1962-1973. doi:10.1109/TIFS.2014.2359333

Shen, S., Smith, J. C., & Goli, R. (2012). Exact interdiction models and algorithms for disconnecting networks via node deletions. *Discrete Optimization*, *9*(3), 172-188.

Shen, S. Q. (2013). Optimizing designs and operations of a single network or multiple interdependent infrastructures under stochastic arc disruption. *Computers & Operations Research, 40*(11), 2677-2688. doi:10.1016/j.cor.2013.05.002

Shen, S. Q., Smith, J. C., & Goli, R. (2012). Exact interdiction models and algorithms for disconnecting networks via node deletions. *Discrete Optimization, 9*(3), 172-188. doi:10.1016/j.disopt.2012.07.001

Sherstjuk, V., Zharikova, M., & Sokol, I. (2018, 24-26 April 2018). *Forest Fire-Fighting Monitoring System Based on UAV Team and Remote Sensing.* Paper presented at the 2018 IEEE 38th International Conference on Electronics and Nanotechnology (ELNANO).

Simão, A., Coutinho-Rodrigues, J., & Current, J. R. (2004). Minimizing network disruption for planned and emergency repairs of water supply systems. *Journal of Infrastructure systems, 10*(4), 176-180.

Singh, K. K. (2018, 23-24 Nov. 2018). *An Artificial Intelligence and Cloud Based Collaborative Platform for Plant Disease Identification, Tracking and Forecasting for Farmers.* Paper presented at the 2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM).

Snediker, D. E., Murray, A. T., & Matisziw, T. C. (2008). Decision support for network disruption mitigation. *Decision Support Systems, 44*(4), 954-969. doi:https://doi.org/10.1016/j.dss.2007.11.003

Strange, R. N., & Scott, P. R. (2005). Plant Disease: A Threat to Global Food Security. *Annual Review of Phytopathology, 43*(1), 83-116. doi:10.1146/annurev.phyto.43.113004.133839

Tan, C. S., Tan, P. S., & Lee, S. S. G. (2015, 6-9 Dec. 2015). *Quantifying disruptions propagation in a supply chain.* Paper presented at the 2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM).

Tan, P. S., Lee, S. G., & Tan, C. S. (2016, 4-7 Dec. 2016). *Modeling disruption propagation in a complex Supply Chain.* Paper presented at the 2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM).

Thompson, B., Morris-King, J., & Cam, H. (2016, 1-3 Nov. 2016). *Controlling risk of data exfiltration in cyber networks due to stealthy propagating malware.* Paper presented at the MILCOM 2016 - 2016 IEEE Military Communications Conference.

Tidy, L., & Woodhead, S. (2018, 10-12 Jan. 2018). *The effect of datagram size and susceptible population on the epidemiology of fast self-propagating malware.* Paper presented at the 2018 International Conference on Information Networking (ICOIN).

Topal, B., & Sahin, H. (2018). The Influence of Information Sharing in the Supply Chain Process on Business Performance: An Empirical Study. *Studies in Informatics and Control, 27*(2), 201-212.

Tsai, M.-K. (2016). Improving efficiency in emergency response for construction site fires: an exploratory case study. *Journal of Civil Engineering and Management, 22*(3), 322-332. doi:10.3846/13923730.2014.897980

U.S. Fire Administration. (2017). U.S. Fire Statistics.

Valizadeh, S., & van Dijk, M. (2019). On the Convergence Rates of Learning-based Signature Generation Schemes to Contain Self-propagating Malware. *arXiv preprint arXiv:1905.00154*.

Van Weyenberge, B., Criel, P., Deckers, X., Caspeele, R., & Merci, B. (2017). Response surface modelling in quantitative risk analysis for life safety in case of fire. *Fire Safety Journal, 91*, 1007-1015. doi:https://doi.org/10.1016/j.firesaf.2017.03.020

Velasquez, J. D., & Nof, S. Y. (2008a). A best-matching protocol for collaborative e-work and e-manufacturing. *International Journal of Computer Integrated Manufacturing, 21*(8), 943-956. doi:10.1080/09511920802014896

Velasquez, J. D., & Nof, S. Y. (2008b). Integration of machine-vision inspection information for best-matching of distributed components and suppliers. *Computers in Industry, 59*(1), 69-81. doi:10.1016/j.compind.2007.06.007

Velasquez, J. D., & Nof, S. Y. (2009). Best-matching protocols for assembly in e-work networks. *International Journal of Production Economics, 122*(1), 508-516. doi:10.1016/j.ijpe.2009.06.018

Velasquez, J. D., Yoon, S. W., & Nof, S. Y. (2010). Computer-based collaborative training for transportation security and emergency response. *Computers in Industry, 61*(4), 380-389. doi:10.1016/j.compind.2009.12.007

Vurro, M., Bonciani, B., & Vannacci, G. (2010). Emerging infectious diseases of crop plants in developing countries: impact on agriculture and socio-economic consequences. *Food Security*, *2*(2), 113-132. doi:10.1007/s12571-010-0062-7

Wang, C., Fu, S., Bai, X., & Bai, L. (2009, 31 March-2 April 2009). *Risk Perception in Modeling Malware Propagation in Networks.* Paper presented at the 2009 WRI World Congress on Computer Science and Information Engineering.

Wang, D., Vinson, R., Holmes, M., Seibel, G., Bechar, A., Nof, S., . . . Tao, Y. (2018). *Early Tomato Spotted Wilt Virus Detection using Hyperspectral Imaging Technique and Outlier Removal Auxiliary Classifier Generative Adversarial Nets (OR-AC-GAN)*. Paper presented at the 2018 ASABE Annual International Meeting, St. Joseph, MI. http://elibrary.asabe.org/abstract.asp?aid=49283&t=5

Wang, S. L., Hong, L., Ouyang, M., Zhang, J. H., & Chen, X. G. (2013). Vulnerability analysis of interdependent infrastructure systems under edge attack strategies. *Safety Science*, *51*(1), 328-337. doi:10.1016/j.ssci.2012.07.003

Wang, T. Y., Zhang, J., Sun, X. Q., & Wandelt, S. (2017). Network repair based on community structure. *Epl*, *118*(6). doi:10.1209/0295-5075/118/68005

Warshall, S. (1962). *A theorem on boolean matrices.* Paper presented at the Journal of the ACM.

Watts, D. J. (2002). A simple model of global cascades on random networks. *Proc Natl Acad Sci U S A*, *99*(9), 5766-5771.

Xanthopoulos, A., Vlachos, D., & Iakovou, E. (2012). Optimal newsvendor policies for dual-sourcing supply chains: A disruption risk management framework. *Computers & Operations Research*, *39*(2), 350-357. doi:https://doi.org/10.1016/j.cor.2011.04.010

Xu, H., Fu, X., Ponnambalam, L., Namatame, A., Yin, X. F., & Goh, R. S. M. (2015, 6-9 Dec. 2015). *A model to evaluate risk propagation considering effect of dynamic risk information sharing and multi-sourcing in supply chain networks.* Paper presented at the 2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM).

Yavari, M., & Zaker, H. (2019). An integrated two-layer network model for designing a resilient green-closed loop supply chain of perishable products under disruption. *Journal of Cleaner Production*, *230*, 198-218. doi:https://doi.org/10.1016/j.jclepro.2019.04.130

Yin, R.-R., Liu, B., Liu, H.-R., & Li, Y.-Q. (2016). Research on invulnerability of the random scale-free network against cascading failure. *Physica A*, *444*, 458-465.

Yoon, S. W., Velasquez, J. D., Partridge, B. K., & Nof, S. Y. (2008). Transportation security decision support system for emergency response: A training prototype. *Decision Support Systems, 46*(1), 139-148. doi:10.1016/j.dss.2008.06.002

Yu, S., Gu, G., Barnawi, A., Guo, S., & Stojmenovic, I. (2015). Malware Propagation in Large-Scale Networks. *IEEE Transactions on Knowledge and Data Engineering, 27*(1), 170-179. doi:10.1109/TKDE.2014.2320725

Yurong, S., Guo-Ping, J., & Yiran, G. (2008, 30 Nov.-3 Dec. 2008). *Modeling malware propagation in complex networks based on cellular automata.* Paper presented at the APCCAS 2008 - 2008 IEEE Asia Pacific Conference on Circuits and Systems.

Zhan, G., Qingbo, Z., & Tingxin, S. (2014). *Analysis and research on dynamic models of complex manufacturing network cascading failures.* Paper presented at the Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2014 Sixth International Conference on.

Zhang, L., Gier, J. d., & Garoni, T. M. (2014). Traffic disruption and recovery in road networks. *Physica A: Statistical Mechanics and its Applications, 401*, 82-102. doi:doi.org/10.1016/j.physa.2014.01.034

Zhang, S.-P., Huang, Z.-G., Dong, J.-Q., Eisenberg, D., Seager, T. P., & Lai, Y.-C. (2015). Optimization and resilience of complex supply-demand networks. *New Journal of Physics, 17*(6), 063029. doi:10.1088/1367-2630/17/6/063029

Zhang, Y., Bhargava, B., & Hurni, P. (2009, 27-30 Sept. 2009). *The Effects of Threading, Infection Time, and Multiple-Attacker Collaboration on Malware Propagation.* Paper presented at the 2009 28th IEEE International Symposium on Reliable Distributed Systems.

Zhao, K., Scheibe, K., Blackhurst, J., & Kumar, A. (2019). Supply Chain Network Robustness Against Disruptions: Topological Analysis, Measurement, and Optimization. *IEEE Transactions on Engineering Management, 66*(1), 127-139. doi:10.1109/TEM.2018.2808331

Zhao, K., Zuo, Z., & Blackhurst, J. V. (2019). Modelling supply chain adaptation for disruptions: An empirically grounded complex adaptive systems approach. *Journal of Operations Management, 65*(2), 190-212. doi:10.1002/joom.1009

Zharikova, M., & Sherstjuk, V. (2018, 11-14 Sept. 2018). *The Hybrid Intelligent Diagnosis Method for the MultiUAV-Based Forest Fire-Fighting Response System.* Paper presented at the 2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT).

Zheng, Y., Jia, B., Li, X.-G., & Jiang, R. (2017). Evacuation dynamics considering pedestrians' movement behavior change with fire and smoke spreading. *Safety Science*, *92*, 180-189. doi:https://doi.org/10.1016/j.ssci.2016.10.009

Zhong, H. (2016). *Dynamic lines of collaboration in e-Work systems*: Ann Arbor : ProQuest Dissertations & Theses.

Zhong, H., & Nof, S. Y. (2015). The dynamic lines of collaboration model: collaborative disruption response in cyber–physical systems. *Computers & Industrial Engineering*, *87*, 370-382. doi:doi.org/10.1016/j.cie.2015.05.019

Zhong, H., & Nof, S. Y. (2020). *Dynamic Lines of Collaboration: Disruption Handling & Control*: Springer International Publishing.

Zhong, H., Nof, S. Y., & Filip, F. G. (2014). Dynamic lines of collaboration in CPS disruption response. *IFAC Proceedings Volumes*, *47*(3), 7855-7860. doi:10.3182/20140824-6-ZA-1003.02403

Zhong, H., Wachs, J. P., & Nof, S. Y. (2013). HUB-CI Model for Collaborative Telerobotics in Manufacturing. *IFAC Proceedings Volumes*, *46*(7), 63-68. doi:https://doi.org/10.3182/20130522-3-BR-4036.00059

Zhou, S., Zhang, T., & Qin, H. (2008, 1-3 Nov. 2008). *The Fire Emergency Response Plan Simulation Based on Particle System.* Paper presented at the 2008 First International Conference on Intelligent Networks and Intelligent Systems.

Zyba, G., Voelker, G. M., Liljenstam, M., Mehes, A., & Johansson, P. (2009, 19-25 April 2009). *Defending Mobile Phones from Proximity Malware.* Paper presented at the IEEE INFOCOM 2009.

VITA

WIN P. V. NGUYEN

EDUCATION

Purdue University, West Lafayette, IN, USA

Ph.D., Industrial Engineering (Major Advisor: Professor Shimon Y. Nof) – GPA: 3.71 – Fall 2016 to Spring 2020

Purdue University, West Lafayette, IN, USA

M.S., Engineering Technology (Major Advisor: Professor Duane D. Dunlap) – GPA: 3.81 – Fall 2015 to Fall 2016

Purdue University, West Lafayette, IN, USA

B.S., Industrial Engineering – GPA: 3.72 – Fall 2010 to Spring 2014

PROFESSIONAL EXPERIENCES

Researcher, PRISM Center, Purdue University – Spring 2017 to Present

- Software, programming, and simulation specialist: C#, Python, MATLAB.
- Development of collaborative production systems simulation for research projects.
- Development of software tools for research projects: HUB-Collaborative Intelligence software for Agricultural Robotic Systems.
- Development of Teamwork Integration Evaluator (TIE) software:
    - TIE/Dynamic Lines of Collaboration experiment designer tool.
    - TIE/Collaborative Response to Disruption Propagation software.

System Engineer, Lap Phuc Co. Ltd, Vietnam – 2011 to 2018 (summers only)

- Development of system software and automation software.

- Process analysis and process improvement.
- Implemented ISO 9001:2015.
- Development of workflow management processes and software.

Mentor for New Industrial Engineers, Lap Phuc Co. Ltd, Vietnam – 2017 to 2018 (summers only)

- Supervise and provide guidance for Industrial Engineer interns.
- Supervise and provide guidance for new Industrial Engineers.

TEACHING EXPERIENCES

Purdue University – Course Management Assistant for the interdisciplinary *Seminar Global Policy Issues (ME 497/NUCL 497/CNIT 499/AGEC 498/POL 493)* – Spring 2012, Spring 2013, Spring 2014. Instructors: Professor Arden L. Bement, Professor Dennis R. Depew, and Professor Suresh Garimella.

Purdue University – Teaching Assistant for the School of Engineering Technology 2-semester *Capstone Projects I and II (ECET 430/460, MET 401/402)* – Fall 2013 to Fall 2015. Instructor: Professor Phillip A. Sanger.

Purdue University – **Teaching Assistant and Assistant Mentor** for the School of Engineering Technology 2-semester *Capstone Projects I and II (ECET 430/460, MET 401/402)* – Fall 2016 to Spring 2017. Instructors: Professor Phillip A. Sanger and Professor Frederick C. Berry. Assistant mentor for 2 teams with Professor Duane D. Dunlap.

Purdue University – Teaching Assistant for the School of Industrial Engineering.
Fall 2017 – Design and Control of Production and Manufacturing Systems (IE 579). Instructor: Professor Shimon Y. Nof. Roles: grading assignments; assisting the development of the new class project; providing critical feedback to student projects.
Spring 2017 – Integrated Production System II (IE 484). Instructor: Professor Seokcheon Lee. Role: grading assignments and grading exams.

Fall 2018 – Integrated Production System II (IE 484). Instructor: Professor Shimon Y. Nof. Roles: grading; providing critical feedback to open-ended case study reports; development of one new case study.
Spring 2019 – Industrial Engineering Senior Design (IE 431). Instructor: Dr. Patrick Brunese. Roles: mentoring and supervising 12 teams, 60 students in total.
Fall 2019 – Integrated Production System II (IE 484). Instructor: Professor Shimon Y. Nof. Roles: grading; providing critical feedback to open-ended case study reports; development of case study problems for the new Production System Serious Game Simulation.
Spring 2020 – Industrial Engineering Senior Design (IE 431). Instructor: Professor Steven Landry, Dr. Patrick Brunese. Roles: mentoring and supervising 15 teams, 74 students in total.

RESEARCH EXPERIENCES

Grant Proposal Preparation

- 2019 - National Science Foundation – Robust Intelligence Program - Scalable Congestion Intelligence for Autonomous Mobile Robots.
- 2019 – AI-based State Estimation and Control for Nuclear Hybrid Energy System.

Reviewer

- International Journal of Production Research
- Computers and Industrial Engineering
- International Journal of Industrial Engineering: Theory, Applications and Practice
- IFAC Conference on Manufacturing Modelling, Management and Control
- Journal of Intelligent Manufacturing
- 25th International Conference on Production Research

PUBLICATIONS

Nguyen, W. P. V., & Nof, S. Y. (2018). Resilience Informatics for Cyber-augmented Manufacturing Networks (CMN): Centrality, Flow and Disruption. *Studies in Informatics and Control, 27*(4), 377-384. doi:https://doi.org/10.24846/v27i4y201801

Nguyen, W. P. V., Nair, A. S., & Nof, S. Y. (2019). *Advancing Cyber-Physical Systems Resilience: The Effects of Evolving Disruptions.* Paper presented at the ICPR-25, Chicago, IL.

Nguyen, W. P. V., & Nof, S. Y. (2019a). Collaborative response to disruption propagation (CRDP) in cyber-physical systems and complex networks. *Decision Support Systems, 117*, 1-13. doi:https://doi.org/10.1016/j.dss.2018.11.005

Nguyen, W. P. V., & Nof, S. Y. (2019b). *Collaborative Response to Disruption Propagation with Established Lines of Collaboration (CRDP/ESLOC) in Cyber-Physical Systems: Informatics for Decision Support.* Paper presented at the ICPR-25, Chicago, IL.

Nguyen, W. P. V., & Nof, S. Y. (2020). Strategic Lines of Collaboration in Response to Disruption Propagation (CRDP) through Cyber-physical Systems. *International Journal of Production Economics.* (Under review)