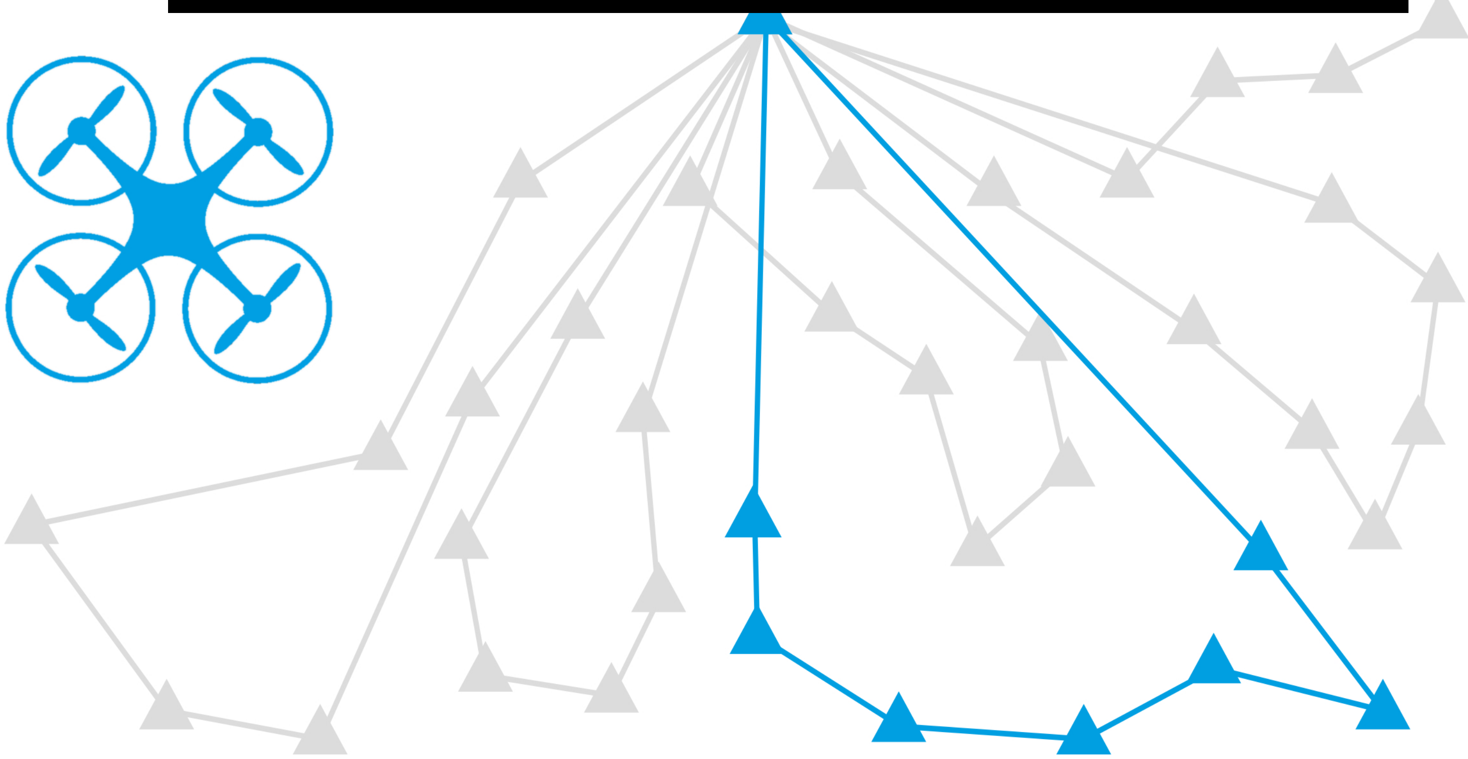
Faculty of Aerospace Engineering

Human Control Performance in Solving Multi-UAV Dynamic Vehicle Routing Problems Using an Ecological Interface

N. W. Klein Koerkamp

September 14, 2017



Delft University of Technology

Challenge the future

Human Control Performance in Solving Multi-UAV Dynamic Vehicle Routing Problems Using an Ecological Interface

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

N. W. Klein Koerkamp

September 14, 2017

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Delft University Of Technology
Department Of
Control and Simulation

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Human Control Performance in Solving Multi-UAV Dynamic Vehicle Routing Problems Using an Ecological Interface"** by **N. W. Klein Koerkamp** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: September 14, 2017

Readers:

Dr.ir. C. Borst

Dr. G. C. H. E. de Croon

Dr.ir. M. M. van Paassen

Prof.dr.ir. M. Mulder

Dr.ir. B. F. Santos

Acknowledgments

I had an amazing time over the past three and a half years during which I have been part of the Control & Simulation Section at the Faculty of Aerospace Engineering at TU Delft as an MSc student. During all the lectures, the many coffee breaks, vrimibo and wobo drinks, and the past four C&S BBQs, I have gotten to know many of our staff, PhD candidates and fellow students. I would like to thank everyone for the great atmosphere you created, the many discussions we have had and the laughs we shared over the past years.

I was very fortunate to be given the opportunity to spend time abroad twice during my time as an MSc student. Max, thank you for your instrumental role in arranging both my internship and my thesis abroad. I would like to thank the team at Boeing Research & Technology Europe for an amazing time in Spain during my internship and the team at the Humans and Autonomy Lab at Duke University for a wonderful time in the United States of America during my short thesis abroad period.

During my thesis work, I received great support from my supervisors, which has been essential to its successful completion. Clark, thank you for all the time you invested in our meetings over the many months we have worked together, for your critical reflections on my work and for all the fun times. Guido, Rene and Max, thank you all for the feedback you provided on my work.

Finally, I would like to thank my family for all the support and motivation you provided during all 7 years I spent as a student in Delft.

Niek

Delft, 14 September 2017

Contents

Acknowledgments **v**

Acronyms **xiii**

List of Figures **xix**

List of Tables **xxi**

1 Introduction **1**

1-1 Background 1

1-2 Problem Statement 2

1-3 Research Question, Aims and Objectives 3

1-4 Scope and Approach 4

1-5 Report Structure 5

I Master of Science Thesis Paper **7**

II Thesis Book of Appendices **25**

A Literature Study **27**

A-1 Vehicle Routing Problem 27

A-2 Human Supervisory Control of Multiple UAVs 28

A-2-1 Supervisory Control 28

A-2-2 Operator Workload 30

A-2-3 Situation Awareness 31

A-2-4 Effects of Automation 32

A-3 Ecological Interface Design 32
A-3-1 Skill, Rule, Knowledge Taxonomy 33
A-3-2 Ecological Interface Design Framework 34
A-3-3 Cognitive Work Analysis 35
A-4 Hierarchical Task Analysis 39
A-5 Previous Research on Multi-UAV EID 40
A-6 Conclusions 41

B Work and Task Analysis 43
B-1 Scope 43
B-2 Work Domain Analysis 44
B-3 Control Task Analysis 47
B-4 Strategies Analysis 49
B-5 Social Organization and Cooperation Analysis 49
B-6 Worker Competencies Analysis 50
B-7 Hierarchical Task Analysis 51
B-8 Conclusions 52

C Preliminary Ecological Interface 53
C-1 Map View 53
C-2 Mission View 56
C-3 Example Case 58

D Final Ecological Interface 65
D-1 Map, Payload and Timeline Views 65
D-2 Flight Time Constraint Visualization 70

E Experiment Design 71
E-1 Experiment Conditions 71
E-2 Experiment Matrix 74
E-3 Training Scenarios 76
E-4 Experiment Scenarios 80

F Experiment Briefing 85
F-1 Introduction 85
F-2 Experiment Setup 87
F-3 Training Scenarios 90
F-4 Training 1: Flight Time Constraint 90
F-5 Training 2: Payload Constraint 91
F-6 Training 3: Depot Congestion 92
F-7 Training 4: DVRP 92
F-8 Training 5-9: DVRP 93

G Experiment Survey **95**
G-1 Intake Survey 96
G-2 Post Scenario Survey 100
G-3 Post Experiment Survey 102

H Experiment Results **107**
H-1 Participant Characteristics 107
H-2 Optimized Solutions 108
H-3 Participant Solutions 112
H-3-1 Participant 1 112
H-3-2 Participant 2 116
H-3-3 Participant 3 120
H-3-4 Participant 4 124
H-3-5 Participant 5 128
H-3-6 Participant 6 132
H-3-7 Participant 7 136
H-3-8 Participant 8 140
H-3-9 Participant 9 144
H-3-10 Participant 10 148
H-3-11 Participant 11 152
H-3-12 Participant 12 156
H-3-13 Participant 13 160
H-3-14 Participant 14 164
H-3-15 Participant 15 168
H-3-16 Participant 16 172
H-4 Training Effects 176
H-5 Post-Survey Participant Feedback 183

I Code Architecture **193**
I-1 Vehicle Routing Problem Optimization 193
I-2 Multi-UAV Simulator 193
I-3 Experiment Survey 194
I-4 Post Experiment Data Processing 194

J Concluding Remarks and Recommendations **199**
J-1 Interface Design 199
J-2 Experiment Design 200
J-3 Future Work 200

Bibliography **206**

Acronyms

ADS Abstraction Decomposition Space

ATM Air Traffic Management

C3 Command, Control, Communication

CONOPS Concept of Operations

CTA Control Task Analysis

CWA Cognitive Work Analysis

DCVRP Distance-Constrained Capacitated Vehicle Routing Problem

EID Ecological Interface Design

HTA Hierarchical Task Analysis

NASA National Aeronautics and Space Administration

RPAS Remotely Piloted Aircraft Systems

RTA Required Time of Arrival

SA Situation Awareness

SOC State of Charge

SRK Skill, Rule, Knowledge

TLX Task Load Index

TSPs Traveling Salesman Problems

UAV Unmanned Areal Vehicle

UAVs Unmanned Areal Vehicles

VRP Vehicle Routing Problem

VRPs Vehicle Routing Problems

WDA Work Domain Analysis

List of Figures

1-1 Forms of Unmanned Areal Vehicle (UAV) control related to the number of Unmanned Areal Vehicles (UAVs) controlled by a single operator. 2
1-2 Research framework 3

A-1 Human supervisory control, adapted from Sheridan and Verplank (1978). 29
A-2 Human supervisor functions as nested loops, adapted from Sheridan (1992). . . . 29
A-3 Multi-UAV system control loops, adapted from Cummings, Bruni, Mercier, and Mitchell (2007). 30
A-4 Human operator workload attributes and performance measures, adapted from Jahns (1973) and Johannsen (1976). 31
A-5 Situation awareness inputs, adapted from Endsley (1990). 32
A-6 Simplified representation of the three levels of performance of skilled human operators, adapted from Rasmussen (1983). 33
A-7 Structure of the interface design problem for complex human-machine systems, adapted from Vicente and Rasmussen (1992). 34
A-8 Five concepts of the Cognitive Work Analysis (CWA) framework and transition from ecological to cognitive considerations, adapted from Vicente (1999). 35
A-9 Abstraction-decomposition space modeling tool, adapted from Rasmussen (1985). 36
A-10 Decision ladder with some possible shortcuts indicated, adapted from Rasmussen (1976). 37
A-11 Strategies analysis, adapted from Vicente (1999). 38
A-12 Hierarchical task analysis diagram 40

B-1 Example multi-UAV payload delivery mission, with black diamonds indicating delivery locations, aircraft symbols indicating UAV positions and solid lines indicating guidance reference. 44

B-2 Abstraction Decomposition Space (ADS) representing the structure of the multi-UAV work domain. 45

B-3 Functional purpose and abstract function levels with means-ends links. 45

B-4 Abstract function and generalized function levels with means-ends links. 46

B-5 Abstract function level at both fleet and UAV level with decomposition links. . . 46

B-6 Generalized function and physical function levels with means-ends links. 47

B-7 Physical function and physical form levels with means-ends links. 47

B-8 Decision ladder for mission planning and disturbance and failure management. . . 48

B-9 Identified strategies for transition of infeasible mission plan to feasible mission plan. 49

B-10 Abstraction Decomposition Space (ADS) with human-automation task division. . 50

B-11 Worker competencies analysis, listing information processing steps, knowledge states, and corresponding skill-, rule-, and knowledge-based behavior, adapted from Kilgore and St-Cyr (2006). 51

B-12 Hierarchical task analysis for mission management and disturbance and failure management. 52

C-1 Waypoint structure as used for endurance envelope derivation. 54

C-2 Geometric relationships between ground speed, wind speed, and airspeed vectors. 55

C-3 Map view display for UAV level control (zero wind). 56

C-4 Endurance envelopes for varying wind conditions (airspeed: 8 [m/s]). 57

C-5 Mission view display for fleet level control. 58

C-6 Map display and energy display prototypes corresponding to an example multi-UAV payload delivery mission case. 64

D-1 Final Interface 65

D-2 Final Interface (continued) 66

D-2 Final Interface (continued) 67

D-2 Final Interface (continued) 68

D-2 Final Interface (continued) 69

D-3 Ellipse definition 70

E-1 Training scenarios 77

E-1 Training scenarios (continued) 78

E-1 Training scenarios (continued) 79

E-2 Experiment scenarios 80

E-2 Experiment scenarios (continued) 81

E-2 Experiment scenarios (continued) 82
E-2 Experiment scenarios (continued) 83

F-1 Example mission of a multi-UAV vehicle routing problem, with the depot at the center (20;25), 5 vehicles and 14 customer locations. 85
F-2 Experiment setup consisting of keyboard, mouse, display and desktop PC (not shown). 87
F-3 Interface views for an example inactive and active UAV case. 89

H-1 Optimized solutions to experiment scenarios. 108
H-1 Optimized solutions to experiment scenarios (continued). 109
H-1 Optimized solutions to experiment scenarios (continued). 110
H-1 Optimized solutions to experiment scenarios (continued). 111
H-2 Participant 1 solutions 112
H-2 Participant 1 solutions (continued) 113
H-2 Participant 1 solutions (continued) 114
H-2 Participant 1 solutions (continued) 115
H-3 Participant 2 solutions 116
H-3 Participant 2 solutions (continued) 117
H-3 Participant 2 solutions (continued) 118
H-3 Participant 2 solutions (continued) 119
H-4 Participant 3 solutions 120
H-4 Participant 3 solutions (continued) 121
H-4 Participant 3 solutions (continued) 122
H-4 Participant 3 solutions (continued) 123
H-5 Participant 4 solutions 124
H-5 Participant 4 solutions (continued) 125
H-5 Participant 4 solutions (continued) 126
H-5 Participant 4 solutions (continued) 127
H-6 Participant 5 solutions 128
H-6 Participant 5 solutions (continued) 129
H-6 Participant 5 solutions (continued) 130
H-6 Participant 5 solutions (continued) 131
H-7 Participant 6 solutions 132
H-7 Participant 6 solutions (continued) 133

H-7 Participant 6 solutions (continued) 134
H-7 Participant 6 solutions (continued) 135
H-8 Participant 7 solutions 136
H-8 Participant 7 solutions (continued) 137
H-8 Participant 7 solutions (continued) 138
H-8 Participant 7 solutions (continued) 139
H-9 Participant 8 solutions 140
H-9 Participant 8 solutions (continued) 141
H-9 Participant 8 solutions (continued) 142
H-9 Participant 8 solutions (continued) 143
H-10 Participant 9 solutions 144
H-10 Participant 9 solutions (continued) 145
H-10 Participant 9 solutions (continued) 146
H-10 Participant 9 solutions (continued) 147
H-11 Participant 10 solutions 148
H-11 Participant 10 solutions (continued) 149
H-11 Participant 10 solutions (continued) 150
H-11 Participant 10 solutions (continued) 151
H-12 Participant 11 solutions 152
H-12 Participant 11 solutions (continued) 153
H-12 Participant 11 solutions (continued) 154
H-12 Participant 11 solutions (continued) 155
H-13 Participant 12 solutions 156
H-13 Participant 12 solutions (continued) 157
H-13 Participant 12 solutions (continued) 158
H-13 Participant 12 solutions (continued) 159
H-14 Participant 13 solutions 160
H-14 Participant 13 solutions (continued) 161
H-14 Participant 13 solutions (continued) 162
H-14 Participant 13 solutions (continued) 163
H-15 Participant 14 solutions 164
H-15 Participant 14 solutions (continued) 165
H-15 Participant 14 solutions (continued) 166

H-15 Participant 14 solutions (continued) 167
H-16 Participant 15 solutions 168
H-16 Participant 15 solutions (continued) 169
H-16 Participant 15 solutions (continued) 170
H-16 Participant 15 solutions (continued) 171
H-17 Participant 16 solutions 172
H-17 Participant 16 solutions (continued) 173
H-17 Participant 16 solutions (continued) 174
H-17 Participant 16 solutions (continued) 175
H-18 Total distance flown, with first and second runs separated. 176
H-19 Extra distance flown, with first and second runs separated. 176
H-20 Extra distance flown as a percentage of optimized solutions, with first and second runs separated. 177
H-21 Extra distance flown as a percentage of optimized solutions for each experiment run, with ordering according to experiment matrix (different experiment conditions are combined in runs, since each participant had a different scenario ordering (latin square)). 177
H-22 Time to first solution, with first and second runs separated. 178
H-23 Time to last solution, with first and second runs separated. 178
H-24 Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). 179
H-24 Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). (continued) 180
H-24 Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). (continued) 181
H-24 Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). (continued) 182
H-25 Figure corresponding to post-survey question 5. 183
H-26 Figure corresponding to post-survey question 6. 183

I-1 Vehicle Routing Problem optimization code architecture. 195
I-2 Multi-UAV simulator code architecture. 196
I-3 Survey web form code architecture. 197
I-4 Post processing code architecture. 197

List of Tables

E-1 Experiment Conditions . . . 71

E-2 Customers and Vehicles per Condition . . . 72

E-3 Control Variables . . . 72

E-4 Number of failures for various numbers of customers and numbers of payload. . . 73

E-5 Experiment matrix for the conducted experiment. Each participant first goes through 9 training runs, followed by 2x8 experiment runs, with one break halfway the experiment. . . . 75

E-6 Training Runs . . . 76

H-1 Participant Characteristics . . . 107

H-2 Q1: How do you assess the usefulness and the functionality of the map view? Please provide examples in your elaboration. . . . 184

H-3 Q2: How do you assess the usefulness and the functionality of the timeline view? Please provide examples in your elaboration. . . . 185

H-4 Q3: How do you assess the usefulness and the functionality of the payload view? Please provide examples in your elaboration. . . . 186

H-5 Q4: How do you assess the usefulness and clarity of the color use in the display? Please provide examples in your elaboration. . . . 188

H-6 Q5: How would you solve the scenario depicted below (high-level description)? NOTE: each vehicle only has one spare payload item on board. . . . 189

H-7 Q6: How would you solve the scenario depicted below (high-level description)? NOTE: each vehicle only has one spare payload item on board. . . . 190

H-8 Q7: Do you have any other comments or suggestions with respect to the interface or the experiment? . . . 192

Chapter 1

Introduction

1-1 Background

UAVs are a class of airplanes that operate without a human pilot on board. Civil, commercial and military applications are common, for example in aerial photography, cargo delivery and surveillance tasks (Koldaev, 2007). UAVs are also the subject of many research activities, for example on vision-based collision avoidance (Tijmons, de Croon, Remes, de Wagter, & Mulder, 2016) and advanced flight control (Smeur, Chu, & de Croon, 2016). The global UAV market is expected to grow significantly over the coming years, with UAV sales revenues in 2024 expected to be over double the revenues of 2013 (BI Intelligence, 2016).

Historically, UAVs were controlled from the ground by a pilot manually flying the vehicle, relying on other operators for sensor and mission management. Due to the high workload associated with manual flight control a many-to-single operator to UAV ratio generally results. Recently, research has been conducted to invert this operator-to-vehicle ratio, which is important for further improvement of military and commercial operations (Cummings, Bertucelli, Macbeth, & Surana, 2014). Introducing automation in the flight control loop and, hence, shifting the role of the human operator to supervisory control offers opportunities to invert the operator-to-vehicle ratio (Mouloua, Gilson, & Hancock, 2003). As part of a future vision for multi-UAV control, a transition to swarms and autonomous control is envisioned. However, currently, this form of control is not only technically challenging, due to the requirement for automation to deal with disturbances, unexpected situations and fault management, but also comes with legal and ethical implications (Lazarski, 2002) that further complicate this type of control.

As discussed, the form of control will have an influence on the number of UAVs that can be effectively controlled. This is further illustrated by Figure 1-1, where the relationship between the different forms of control (manual, supervisory and swarm) are related to the number of UAVs controlled by a single operator. The figure shows that for increasing numbers of UAVs, higher levels of automation and autonomy are required for a single human operator to achieve sufficient control performance, which is supported by the results of a meta-analysis performed by Cummings, Bruni, et al. (2007).

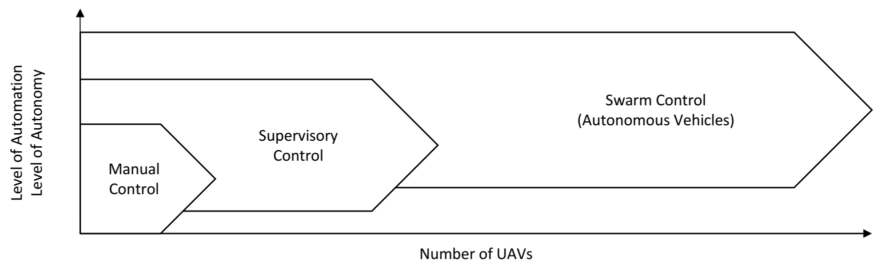


Figure 1-1: Forms of UAV control related to the number of UAVs controlled by a single operator.

1-2 Problem Statement

For multi-UAV control, supervisory control is currently considered to be the best mix of the humans ability to deal with unanticipated events and the workload reduction effects that are associated with the introduction of automation (Cummings, 2006). Also, UAVs are generally costly vehicles, which leads to the desire to allow the human operator to stay involved in the flight execution and be able to intervene when deemed necessary. Much research is focused on the best level of automation and how to achieve good human-automation interaction (Cummings, Clare, & Hart, 2010; Prinet, Terhune, & Sarter, 2012; Ruff, Narayanan, & Draper, 2002), however, the introduction of higher levels of automation also brings downsides. Automation "brittleness" leads to problems with high levels of automation in uncertain and unforeseen situations (Guerlain et al., 1995; Smith, McCoy, & Layton, 1997) and humans tend to be susceptible to automation bias, which leads to both errors of omission and commission (Cummings, 2004). Omission errors relate to human failure to identify problems because automation does not explicitly alert them, and commission errors relate to humans following incorrect automated directions or recommendations.

Compared to increasing levels of automation, much less attention is given to the human-machine interface used to control the vehicles and the positive influence on control performance good visualizations can have (Cummings, Brzezinski, & Lee, 2007). Recently, some work on multi-UAV interfaces has been performed by Fuchs, Borst, de Croon, van Paassen, and Mulder (2014) and by van Lochem, Borst, de Croon, van Paassen, and Mulder (2015), where ecological interface design (Vicente & Rasmussen, 1992) techniques were used to develop interfaces that improve single-operator multi-UAV mission control. These studies focused on generic ground-surveillance missions, where much effort was put towards the representation of low level information. Improvements in mission control are envisioned possible by putting more focus towards the integration of information to yield representations that assist in higher level mission planning. Therefore, the goal of this research is to develop an ecological interface for a multi-UAV ground control station, with a focus on human operator mission planning and disturbance and failure management for a payload delivery mission.

The payload delivery mission under consideration is consistent with a Distance-Constrained Capacitated Vehicle Routing Problem (DCVRP). Where a fleet of vehicles is used to deliver payload to customers. The solution space is constrained by the flight time of the UAVs in the fleet, their payload carrying capacity and the capacity at the depot from where all vehicles start and end their routes. The overall control goal is to determine the most efficient routing, such that all customers are served, while satisfying the aforementioned constraints.

1-3 Research Question, Aims and Objectives

To aid in the structuring of the research efforts, a research framework is created, which is depicted in Figure 1-2. In the research framework, several relevant fields are identified: multi-UAV supervisory control, ecological interface design and cognitive work and task analysis. These research areas will serve as the basis for an interface design framework that will be used for the to be developed interface. This interface framework combined with theory on Vehicle Routing Problems (VRPs) and the multi-UAV mission definition will yield the tasks the human operator is expected to perform and associated work domain constraints. The execution of these tasks are what needs to be supported by the interface and are, in combination with the work domain constraints, used as inputs for the creation of interface visualizations. The effectiveness of these visualizations is then assessed in a human-in-the-loop evaluation study.

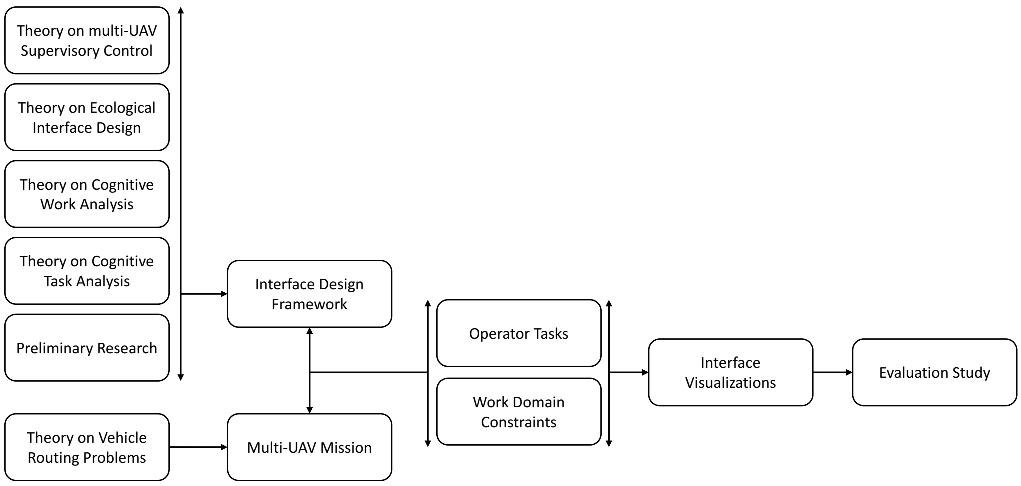


Figure 1-2: Research framework

From the information provided by the research framework combined with the research goal, research questions can be formulated. The main research question of the proposed research is the following:

What ecological ground control station visualizations support human operator mission planning and disturbance and failure management for a multi-UAV payload delivery mission?

Since this main research question would be too complex to answer directly, the following research sub-questions are formulated:

1. (a) *What contributions does the theory on VRPs have for the multi-UAV mission definition?*
   (b) *What contributions does the theory on multi-UAV supervisory control have for an interface design framework?*
   (c) *What contributions does the theory on ecological interface design have for an interface design framework?*
   (d) *What contributions does a symbiosis of cognitive work and task analysis have for an interface design framework?*
2. (a) *What tasks are assigned to the human operator according to the interface design framework for a multi-UAV payload delivery mission?*
   (b) *What work domain constraints are relevant according to the interface design framework for the human operator tasks in a multi-UAV payload delivery mission?*
3. *What visualizations show relevant information and constraints for support of human operator tasks?*
4. *How do the visualizations influence human operator mission planning and disturbance and failure management?*

The objective of the research project is to develop and evaluate an ecological interface for a multi-UAV ground control station for a DCVRP mission by using an interface design framework based on supervisory control, ecological interface design and a symbiosis of cognitive work and tasks analysis. The research objective can be split up into several parts, first the interface design framework needs to be assembled from the theories on supervisory control, ecological interface design and cognitive work and task analysis. Then, the ecological interface will be developed and implemented and finally, the interface is evaluated.

Novel about this research project is the use of ecological interface design instead of introducing high levels of automation for the human operator to achieve good mission planning and control. Supporting this approach are the results from research performed by Ruff et al. (2002), which showed that in an experiment where human operators performed supervisory control of multiple simulated UAVs with varying levels of automation and decision-aid fidelity, human operators should have an integrated role in the decision making process, which is associated with an intermediate level of automation. Also, previous research (Dry, Lee, Vickers, & Hughes, 2006; Macgregor & Chu, 2011) showed good human performance in visually solving Traveling Salesman Problems (TSPs), which is a specific case of the more generic Vehicle Routing Problem (VRP).

1-4 Scope and Approach

As discussed in Section 1-3, the research objective is to develop and evaluate an ecological interface for a multi-UAV ground control station using an interface design framework that is a symbiosis of various theories. First, this symbiosis of supervisory control, ecological interface design and cognitive work and task analysis needs to be performed by means of a literature study. Theory on multi-UAV supervisory control is important for the development of the ecological interface. It is essential that the human-machine interface is designed in such a way as to keep operator workload at an acceptable level and provide sufficient situation awareness for effective control. Also, theory on ecological interface design and related subjects cognitive work and task analysis are important for obtaining cues about the information that should be presented in the ecological interface.

The literature study will extensively cover the aforementioned subjects, whereby requirements or tools that are relevant for ecological interface design are considered for the interface design framework. This framework will guide in the identification of tasks that the human operator should perform and the work domain constraints associated with those tasks and the environment.

Important remark is that this framework will not give guidance on how this information should be represented. This is the next phase of the project where, based on the information from the interface design framework, visualizations are created that communicate the required information effectively to the human operator. This phase is often referred to as "overcoming the creative gap" and it is therefore difficult to formulate a strict procedure or methodology for this phase. This phase will be characterized by iterations where ideas are quickly prototyped, evaluated and improved. Then, these visualizations will be implemented in a simulated environment.

Final step of this research project will be to evaluate the visualizations. It is hypothesized that these visualizations will allow for better human operator mission planning and disturbance and failure management. To asses this hypothesis, an evaluation study will be conducted where mission planning and disturbance and failure management for some human-in-the-loop scenarios will be assessed. Performance metrics will be automatically recorded during the evaluation study and participants will be requested to fill out a questionnaire regarding the effectiveness of the visualizations.

Some assumptions are made on mission and UAV characteristics to limit the scope of the current research. As mentioned before, the UAV mission under consideration is a DCVRP mission in which UAVs will each carry a number of payload items that have to be delivered to pre-specified delivery locations. The fleet is assumed homogeneous, flying at constant airspeed, with automatic obstacle and traffic avoidance. Furthermore, with respect to atmospheric conditions, no wind is taken into account. The mission starts from a pre-optimized schedule, which contains information on which delivery each UAV should make. During mission execution, disturbances from the pre-optimized schedule are introduced by vehicle failures. Since the mission starts from a pre-optimized schedule, the human operator control task will be focused on perturbation management.

1-5 Report Structure

This thesis report is structured as follows. Part I contains the thesis paper. Part II contains all appendices to the paper. The appendices are structured as follows. Appendix A contains an elaborate literature study, which discusses the following relevant research fields as aforementioned in the research framework: VRPs, multi-UAV supervisory control, ecological interface design and cognitive work and task analysis. Then, in Appendix B, a cognitive work and task analysis is presented focused on the multi-UAV payload delivery mission that is the subject of this research. This will result in information on what should be represented in the to be developed ecological interface and associated work domain constraints. Appendix C elaborates on the preliminary visualizations that were developed to effectively present the previously identified information to the human operator. Appendix D presents the final ecological interface design. Then, Appendix E contains a detailed description of the human-in-the-loop experiment design. The experiment will be used to assess the effectiveness of the developed ecological interface as well as to assess human performance in solving dynamic VRPs. Appendix F contains the experiment briefing that was provided to participants. Appendix G presents the experiment survey that was used to gather information during the experiment. Appendix H contains detailed experiment results. Appendix I presents the code architecture of the optimization script, simulator, survey and post-processing scripts. Finally, Appendix J will discuss concluding remarks and recommendations for improvements and future work.

Part I

Master of Science Thesis Paper

Human Control Performance in Solving Multi-UAV Dynamic Vehicle Routing Problems Using an Ecological Interface

Niek W. Klein Koerkamp
Supervisors: Clark Borst, Guido C. H. E. de Croon, Marinus M. van Paassen, and Max Mulder

***Abstract*—Real-time optimization of Vehicle Routing Problems during mission operations raises concerns regarding reliability of obtaining a solution and solution time. Improvements in control performance by having a human-in-the-loop might be possible by leveraging human visual pattern recognition qualities. By developing an ecological interface, supporting the operator in controlling multiple Unmanned Aerial Vehicles in a simulated payload delivery mission, and by conducting a human-in-the-loop experiment, interface effectiveness and human control performance in Dynamic Vehicle Routing Problems was investigated. Results show the ecological interface offers good support and scales well with problem size. Results also show participants can in some cases achieve solutions faster and more reliably compared to an optimization algorithm, although generally yielding less efficient solutions. Having a human-in-the-loop can thus offer improved control performance over relying on pure automation, especially in time critical situations.**

***Index Terms*—Vehicle Routing Problem, Ecological Interface Design, Human Control Performance, Unmanned Aerial Vehicle, Perturbation Management.**

I. INTRODUCTION

VEHICLE routing problems (VRP) are at the core of many logistics applications [1]. With the current focus on just-in-time logistics and data-driven analysis techniques, cost-efficient routing of a fleet of vehicles plays an important role in many industries [2]. A practical example of a VRP is courier mail service companies offering mail and package pick-up and delivery services [3]. Determining efficient routes for each vehicle in the fleet, such that the overall mission goal is achieved within distance, capacity and time constraints, defines the VRP [4]. Although much attention is given in literature to obtaining routes for static problems, in many real-life situations these problems become dynamic due to changing customer locations, vehicle failures or uncertain service and travel times caused by perturbations during mission operations [5], [6].

Generally, automation works well in the expected situations for which it was designed, but performance in uncertain and unforeseen situations can be problematic [7]. Regarding automation for solving VRPs, various challenges can be identified. First, the optimization problem needs to be explicitly formulated. Although a wide range of VRP types has been studied in literature, constructing an algorithm that takes into account all possible disturbances and stochastic properties of a real life application is challenging [3]. When not taking into account all constraints inflicted by these disturbances and properties, a theoretical optimal solution will not transfer well to real life operations. Second, due to the complexity of the VRP optimization problem and current computational limitations, finding an optimal solution might take a long time (in the order of hours or sometimes even days) [8]. Although this might not be a problem for generating schedules well before mission execution, it is a concern when a need for re-optimization arises due to a perturbation during mission operations. Third, for some scenarios there might not even be a solution. For example for overconstrained problems, VRP algorithms might be unable to find a solution at all, without first performing some kind of constraint relaxation [9].

N. W. Klein Koerkamp is a graduate student, C. Borst, G. C. H. E. de Croon and M. M. van Paassen are assistant professors, and M. Mulder is a professor with the Control & Simulation Section, Faculty of Aerospace Engineering, TU Delft, Delft, The Netherlands (e-mail: niek@kleinkoerkamp.com, c.borst@tudelft.nl, g.c.h.e.decroon@tudelft.nl, m.m.vanpaassen@tudelft.nl, m.mulder@tudelft.nl).

High levels of automation might be problematic for decision making in dynamic environments, such as during mission operations, due to risks and inability of the automation to be perfectly reliable [10]. Reducing the level of automation and putting a human in the control loop can offer increased control performance in these situations. Previous research on human performance in solving Traveling Salesman Problems (TSP), shows good human performance in creating optimal routes purely based on a visual representation of the customer locations and vehicle routes [11]–[21]. By leveraging these human visual pattern recognition qualities, good performance in solving real-time dynamic VRPs might be achieved.

To investigate this, this research comprises the design of an interface to be able to effectively present the VRP to a human operator as well as a human-in-the-loop experiment to evaluate the interface design and human control performance. For the interface design, use is made of the Ecological Interface Design (EID) framework [22]. EID is characterized by a constraint-based approach, where both environment as well as cognitive constraints are analyzed to aid in the design of interfaces that support the operator's mental model of the work domain [23]. This approach differs from other interface design frameworks that are generally task based and focus on procedures and advisories. This constraint-based approach is deemed essential, since human visual pattern recognition qualities will be leveraged to come up with solutions, the process of which, given the aforementioned challenges related to automation, is difficult to capture in explicit procedures or advisories. Furthermore, the interface is intended to support the adaptive problem solving required for dealing with perturbations during mission operations, which the EID framework explicitly aims to support [24].

The human-in-the-loop experiment will focus on multi-UAV (Unmanned Aerial Vehicle) payload delivery as a specific application of the VRP. Vehicle failures will be introduced to the mission to make the scenarios dynamic. The task of the human operator will be to perform real-time perturbation management to ensure mission success. This application has been chosen due to the inherent time pressure introduced with flight operations. In addition, flying vehicles are often unable to stop mid-air to wait for new instructions, which makes this application especially challenging. Furthermore, there has been interest from industry to use fleets of UAVs for such payload delivery applications [25], [26].

This paper is structured as follows. In Section II, background information on the VRP is provided. Then, Section III presents the Cognitive Work Analysis (CWA) that provides the theoretical basis for the ecological interface that is discussed in Section IV. Section V covers the human-in-the-loop experiment. Then, the experiment results are presented in Section VI. These results are discussed in Section VII. Finally, Section VIII covers the conclusions.

II. The Vehicle Routing Problem

The generic family of Vehicle Routing Problems can be defined as follows:

"Given a set of transportation requests and a fleet of vehicles, determine a set of vehicle routes to perform all transportation requests with the given vehicle fleet at minimum cost; in particular, decide which vehicle handles which requests in which sequence so that all vehicle routes can be feasibly executed." [8]

Various types of VRPs exist, each with its own specific set of constraints [8]. Under consideration for this research is the Distance-Constrained Capacitated Vehicle Routing Problem (DCVRP) with resource constraints at the depot, which, in addition to the generic VRP attributes, includes a distance constraint for each vehicle, modeling fuel limitations, a capacity constraint, modeling payload capacity, and a depot capacity limit (only a finite number of vehicles are allowed to depart and arrive at the depot simultaneously). More formally, the transportation requests in the DCVRP consist of the distribution of goods from a single depot, denoted as point 0, to customers, which are defined as a set of n other points, with $N = \{1, 2, ..., n\}$. The customer demand, $q_i \geq 0$, is defined as the number of goods that needs to be delivered to the customer $i \in N$. The fleet of vehicles used to distribute the goods, defined as $K = \{1, 2, ..., |K|\}$, is assumed homogeneous. The homogeneity entails that the $|K|$ vehicles in the fleet have identical distance constraints, capacity $Q > 0$ constraints, and associated cost. Each vehicle starts at the depot, delivers goods to a subset of customers $S \subset N$ visiting customer locations only once, then returns to the depot. When traveling from customer i to customer j, the vehicle incurs the travel cost c_{ij}. Cost is assumed symmetric, where the cost of traveling from i to j is equal to the cost of traveling from j to i.

The VRP can be formulated as follows [5]:

$$\text{minimize} \quad \sum_{i \neq j} c_{ij} x_{ij} \tag{1}$$

$$\text{subject to} \quad \sum_{j=1}^{n} x_{ij} = 1 \quad (i = 1, ..., n), \tag{2}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad (j = 1, ..., n), \tag{3}$$

$$\sum_{j=1}^{n} x_{0j} = |K| \quad (j = 1, ..., n), \tag{4}$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - v(S) \quad (S \subset N; |S| \neq 0), \tag{5}$$

$$x_{ij} \in \{0, 1\} \quad (i, j = 1, ..., n; i \neq j). \tag{6}$$

Equation 1 represents the total cost of all vehicle routes and is the cost function to be minimized. Then, Equation 2 and Equation 3 ensure that each customer vertex is connected to two other vertices. Similarly, Equation 4 dictates there are K vertices leaving the depot, corresponding to one route for each vehicle in the fleet. Equation 5 serves as the capacity constraint and subtour elimination constraint (ensuring a vehicle route is joined and does not consist of separated tours), where $v(S)$ represents the minimum number of vehicles required to serve the customers in S. Finally, Equation 6 defines the decision variable x_{ij}, which is equal to 1 if and only if the arc (i, j) is part of the optimal solution.

Although many algorithms, both exact and (meta)heuristic, already exist to optimize and approximate solutions to VRPs, it is still a topic that is heavily researched [27], [28]. Not only does it draw attention because of its notorious difficulty as a combinatorial optimization problem, but also because of its practical relevance [8].

III. Cognitive Work Analysis

Cognitive Work Analysis (CWA) is a framework for the identification of requirements for effective work support [29]. To successfully conduct a CWA, a precise description of the system boundaries is required. Here, the system under consideration is a fleet of UAVs, departing from a single depot, delivering payload to numerous customer locations in a scenario consistent with the Distance-Constrained Capacitated Vehicle Routing Problem (DCVRP) with resource constraints at the depot.

A. Scope

The purpose of the system under consideration is to serve all customers, minimize cost (by flying the most efficient routes) and to execute the mission in a safe manner. To further analyze the work domain, the constraints that affect the system's purpose are identified. Since the mission corresponds to a DCVRP, high level constraints are the UAV payload capacity limit, the UAV flight time limit and the depot capacity limit. Also, communication range (both related to ground station and UAV), airspace restrictions, vehicle separation requirements

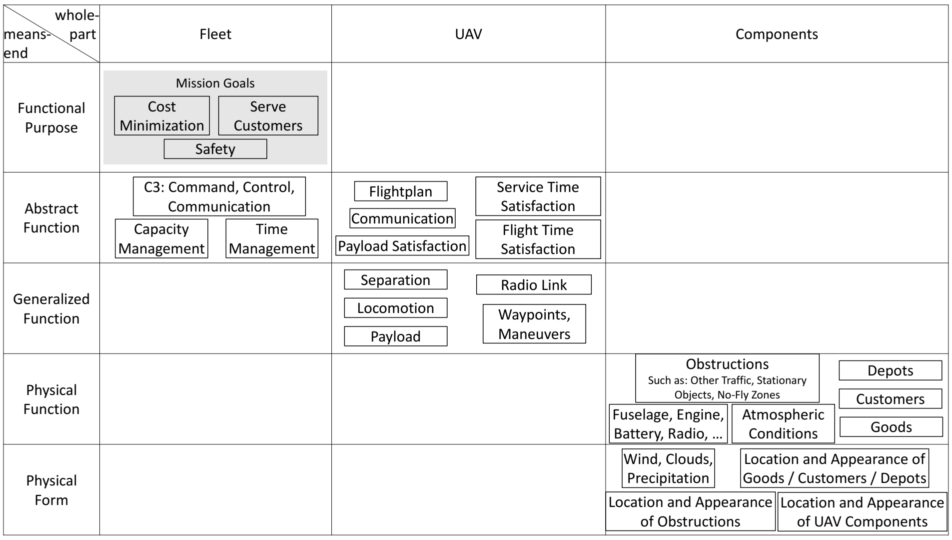


Fig. 1: Abstraction Decomposition Space for the multi-UAV DCVRP work domain.

(with respect to both terrain and other vehicles), weather (such as wind), and UAV flight performance characteristics introduce work domain constraints.[1]

B. Work Domain Analysis

In the EID framework, the Abstraction Decomposition Space (ADS) [30] is used as a representation of the complex work domain. It offers hierarchical structuring along two dimensions: the part-whole dimension, which relates parts and subsystems to the system as a whole, and the means-ends dimension, which offers levels of abstraction for the purposes of the system. Bottom-up, the ADS provides a physical basis for capabilities, resources and causes of malfunction, whereas top-down, it provides reasons for proper function, requirements and a purpose basis [30].

The ADS constructed for the Multi-UAV DCVRP scenario under consideration is depicted in Figure 1, a more detailed description of the different levels is provided below.

1) Functional Purpose: The purpose of the system is to serve all customers, minimize cost and to execute the mission in a safe manner. The focus of this research is on the first two aspects.

2) Abstract Function: This level describes the required underlying principles to achieve the purpose of the system. These are, C3 (Command, Control, Communication) capability, capacity management and time management. In this context, C3 considers the ability to command and control the vehicles that make up the fleet of UAVs and the communication infrastructure that supports this functionality. Furthermore, capacity management considers the ability of the fleet of UAVs to serve customers, and hence comprises the vehicles in the fleet as well as the payload capacity of those vehicles. Finally, time management considers the ability of the fleet of UAVs to handle all customer demands in time, and hence comprises flight time characteristics as well as depot service time requirements.

3) Generalized Function: The generic functions that enable the abstract functions are described at this level. Separation, locomotion and waypoints and maneuvers facilitate the flight plan and service time and flight time satisfaction. The radio link is a means for providing communication and payload facilitates customer payload satisfaction.

4) Physical Function: This level describes the functions provided by the system components. Obstructions are introduced by, for example, other traffic, stationary objects and no-fly zones. The UAV platform is defined by its components, such as the fuselage, engine, battery and radio. Weather is dictated by the atmospheric conditions. Finally, the depots function as the UAV launch sites, the customers as the delivery locations, and the goods as the items to be transported.

5) Physical Form: The spatial location and appearance of the system components are represented at this level.

C. Control Task Analysis

The Control Task Analysis (CTA) complements the WDA [29]. Whereas the WDA is focused on describing the work domain and associated constraints, the CTA offers a description of known tasks associated with a particular application and identifies constraints and requirements associated with those. The multi-UAV DCVRP control task is mapped onto the decision ladder, see Figure 2, which serves as a template for decision processes [31]. Although the discussion on the decision ladder below starts from the activation activity and follows the structure of the decision ladder all the way to procedure execution, it is important to remark that decision processes can start and end anywhere and thus do not necessarily traverse the entire structure of the decision ladder [29].

Activation takes place by means of identification of perturbations affecting mission operations. Observations of the depots, vehicles and customer locations serve as input to the

[1]Although these constraints are considered in the WDA, the subsequent discussions regarding the interface design assume infinite communication range, no airspace restrictions, guaranteed vertical separation, no wind and instantaneous turn dynamics, since they are not required for the experiment.

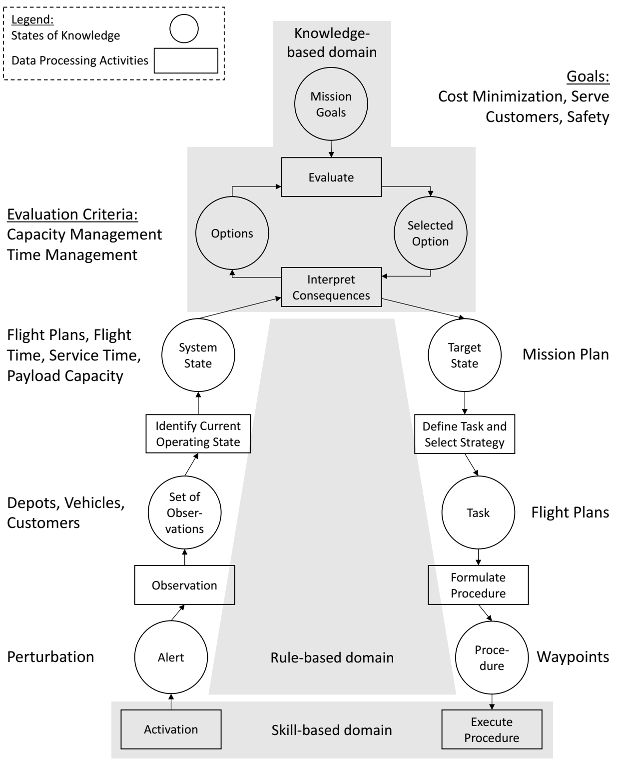
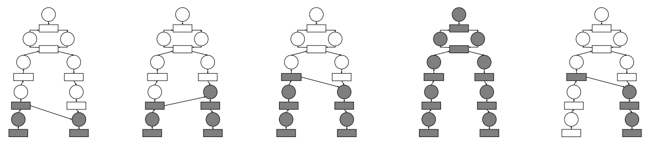


Fig. 2: Decision ladder for the multi-UAV DCVRP control task.



(a) Tactical – satisfice (b) Tactical – satisfice (c) Strategic – satisfice (d) Strategic – optimize (e) Post-optimization

Fig. 3: Envisioned DCVRP decision process shortcuts mapped onto the decision ladder.

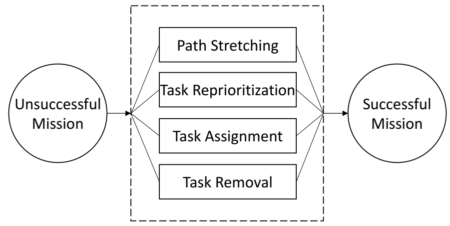


Fig. 4: Information flow map depicting envisioned strategies.

identification of the current operating state. The operating state is defined by the current flight plans, flight and service times and vehicle payload capacities. The information on the system state is used to evaluate different options for dealing with the perturbation. These options are evaluated using criteria based on capacity and time management and are driven by the mission goals to minimize cost, serve all customers and execute the mission in a safe manner. The selected option results in an explicit target state, or mission plan, which is subsequently used to define the desired flight plans. To be able to implement these flight plans, the waypoints that make up the flight plan need to be formulated. The final step that is left is to execute the procedure to implement these waypoints.

Several shortcuts in the decision process are envisioned. These shortcuts have been mapped onto the decision ladder as depicted in Figure 3. The first strategy, depicted in Figure 3a, is tactical – satisfice, where the observations of the UAVs and customer locations are directly interpreted as desired waypoints. This strategy can be used when, for example, a vehicle flies right past a customer location that is, because of its spatial proximity, desired to be part of the vehicle's flight plan. Another variant of this shortcut, depicted in Figure 3b, is when the observations are interpreted in terms of desired flight plans instead of waypoints. This strategy can be used when, for example, a set of waypoints in close spatial proximity is desired to be part of the vehicle's flight plan. The third shortcut, depicted in Figure 3c, is strategic – satisfice, where the current operating state, in terms of flight plans, flight time, service time and payload capacity, is directly interpreted as desired flight plans. This strategy can be used when, for example, a pre-defined strategy is used to reroute UAVs in case of a failure, by means of interpreting the current state in terms of the desired flight plans. The fourth shortcut, depicted in Figure 3d, corresponds to a strategic – optimize strategy where the current system state serves as an input to a decision process, where different options are explored, evaluated towards the mission goals and consequences interpreted, leading to a target state in terms of a desired mission plan. This strategy can be used when strategies are developed and evaluated to come up with optimized solutions satisfying the mission goals. Finally, depicted in Figure 3e is the post-optimization strategy, where an already achieved first solution and direct observations of the current flight plans are used to identify optimization opportunities, which are directly perceived as desired flight plans. Some signs of optimization opportunities are crossing flight paths and inefficient clustering of customer locations [21].

D. Strategies Analysis

The core of the control problem discussed here is to transition from an unsuccessful mission, caused by UAV failures, to a successful mission. Mission success is dictated by the mission goals and work domain constraints previously identified. Four different strategies to achieve the control objective are considered. Path stretching can be used to solve mission plan infeasibilities related to, for example, too many vehicles arriving at the depot at the same time. Also, path stretching is an important strategy for ensuring vehicle separation and avoiding areas of bad weather or no-fly zones. To facilitate serving all customers, taking into account the mission constraints, customer locations can be assigned or removed from a UAV's flightplan. Also, customer locations can be reprioritized to yield more efficient routes or to deal with possible customer

Information Processing Step	Resultant Knowledge State	Skill-Based Behavior	Rule-Based Behavior	Knowledge-Based Behavior
Observe future flightplan for each UAV	Whether any UAVs have infeasible flightplans	Monitor UAV flight plans	Perceive explicit indication that UAVs have infeasible flightplans	Reason, based on proposed flightplans, that UAVs may have infeasible flightplans
Predict time-based states for UAVs with infeasible flightplans	Whether UAVs will run out of energy, penetrate no-fly zones, have insufficient payload, etc.	Perceive energy levels, no-fly zones, payload levels, etc.	Use heuristics to estimate whether UAVs will run out of energy, penetrate no-fly zones, have insufficient payload, etc.	Calculate, based on current state and flightplan, if UAV has sufficient resources to complete flightplan
Determine the criticality of infeasible flightplans	Which UAV flightplan needs to be addressed in what order	Perceive which UAV is in most critical state	Use heuristics to estimate when UAV resources are depleted	Calculate, based on consumption of resources when UAV resources are depleted
Choose to modify UAV flightplan to address infeasibility	Which UAV flightplans need to be modified to yield mission feasibility	Directly perceive that one or more UAVs must be redirected	Apply doctrine: (e.g. If insufficient resources, MUST change UAV flightplan)	Reason from knowledge of proposed flightplans, current state, and expected future use of resources that flightplans must be altered
Select method for accomplishing feasible flightplan	Operator awareness of new flightplan	Respond automatically to insufficient resources by directly manipulating a representation of UAV flightplan(s)	Classify insufficient resources within a set of generalized scenarios and select appropriate stereotypical control rule	Develop new, optimized flightplan(s) based on weighted criteria including urgency, priority, efficiency, safety, etc.
Convey flightplan modifications to UAV for execution	UAV awareness of new flightplan	Direct, simultaneous interaction with communication equipment through control interface through input of updated flightplan information	Apply stereotypical control rules to select method/sequencing for conveying proposed flightplans	Reason using knowledge of UAV systems, priorities, urgency, etc., the best means and order for contacting each UAV to convey proposed flightplan(s)

Fig. 5: Worker competencies analysis, adapted from [32].

related delivery time requirements. These strategies are not mutually exclusive, most likely a combination needs to be used to solve a given unsuccessful mission. A summarized representation of the different control strategies is depicted in Figure 4.

E. Social Organization and Cooperation Analysis

Two actors can be identified that are active in the work domain currently under consideration: the human operator and automation. Automation is required to relieve the operator of some tasks, such that effective single-operator multi-UAV control can be achieved [33]. Automation is well suited for tasks that are repetitive and predictable, whereas humans are good at finding solutions to unforeseen and unpredictable problems [34]. Hence, tasks related to low level UAV flight control, separation from terrain and collision avoidance are allocated to automation. This allows the operator to focus on higher level mission and perturbation management.

F. Worker Competencies Analysis

Worker competencies are assessed based on the Skill, Rule, Knowledge (SRK) taxonomy [35]. The SRK taxonomy is a qualitative human performance model that is used to identify how information should be communicated to the operator. Figure 5 provides an overview of information processing steps, resulting knowledge states, and corresponding skill-, rule-, and knowledge-based behavior for achieving single-operator multi-UAV mission management for a DCVRP mission.

G. Hierarchical Task Analysis

Although work domain centered approaches offer tools for analyzing the functional structure of a work domain, the actions an actor can or should take to accomplish the control objective are not explicitly considered. This is the focus of task analysis techniques. Since work and task analysis approaches offer different perspectives on the knowledge requirements for human-centered system design, also performing a task analysis in addition to the CWA yields an integrated approach where a more complete set of knowledge is obtained [36]. Ecological interfaces have been designed in the past based on such an integrated approach, using a Hierarchical Task Analysis (HTA) [37] to complement the CWA [38].

The HTA is focused on an analysis of hierarchical means-ends relationships, relating to how a set of subtasks allows for achieving a higher level goal, and sequential relationships, which consider task related temporal requirements. The result of the analysis is depicted in Figure 6. From the overall goal of the mission at the top of the diagram, tasks are formulated that need to be performed to successfully achieve this goal. For these tasks, several subtasks are identified that, together, accomplish the higher level tasks. Also, temporal information is included by means of the numbers between the brackets, that indicate whether tasks are to be executed concurrently or consecutively.

IV. Interface Design

A. Layout, Structure and Functionality

Figure 7 provides a representation of the layout and structure of the interface for a simple scenario. The scenario consists of three UAVs delivering payload to six customers from a single depot location. The interface consists of three separate views, namely the map A, payload B, and timeline view C, where the map view presents information from a

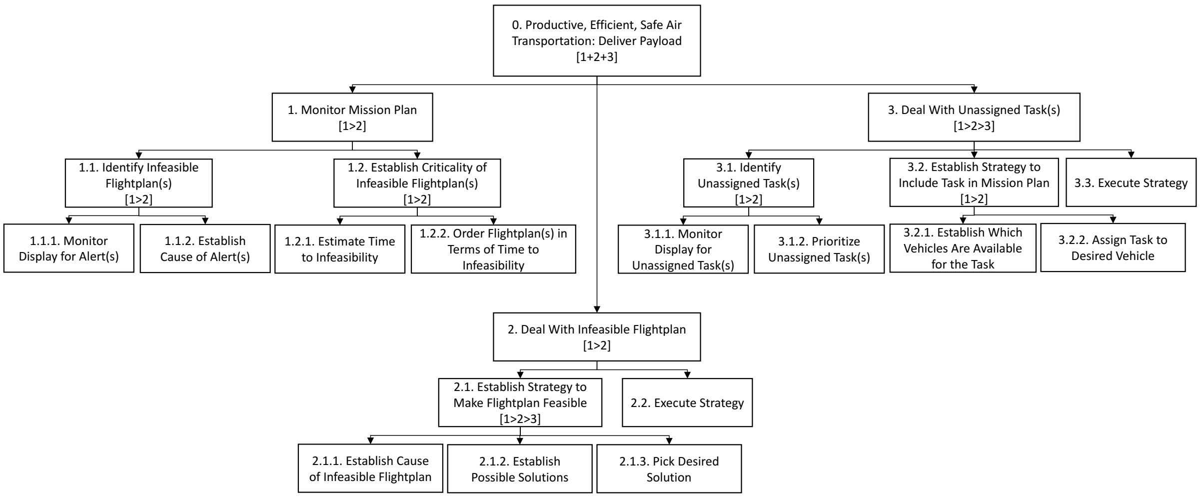


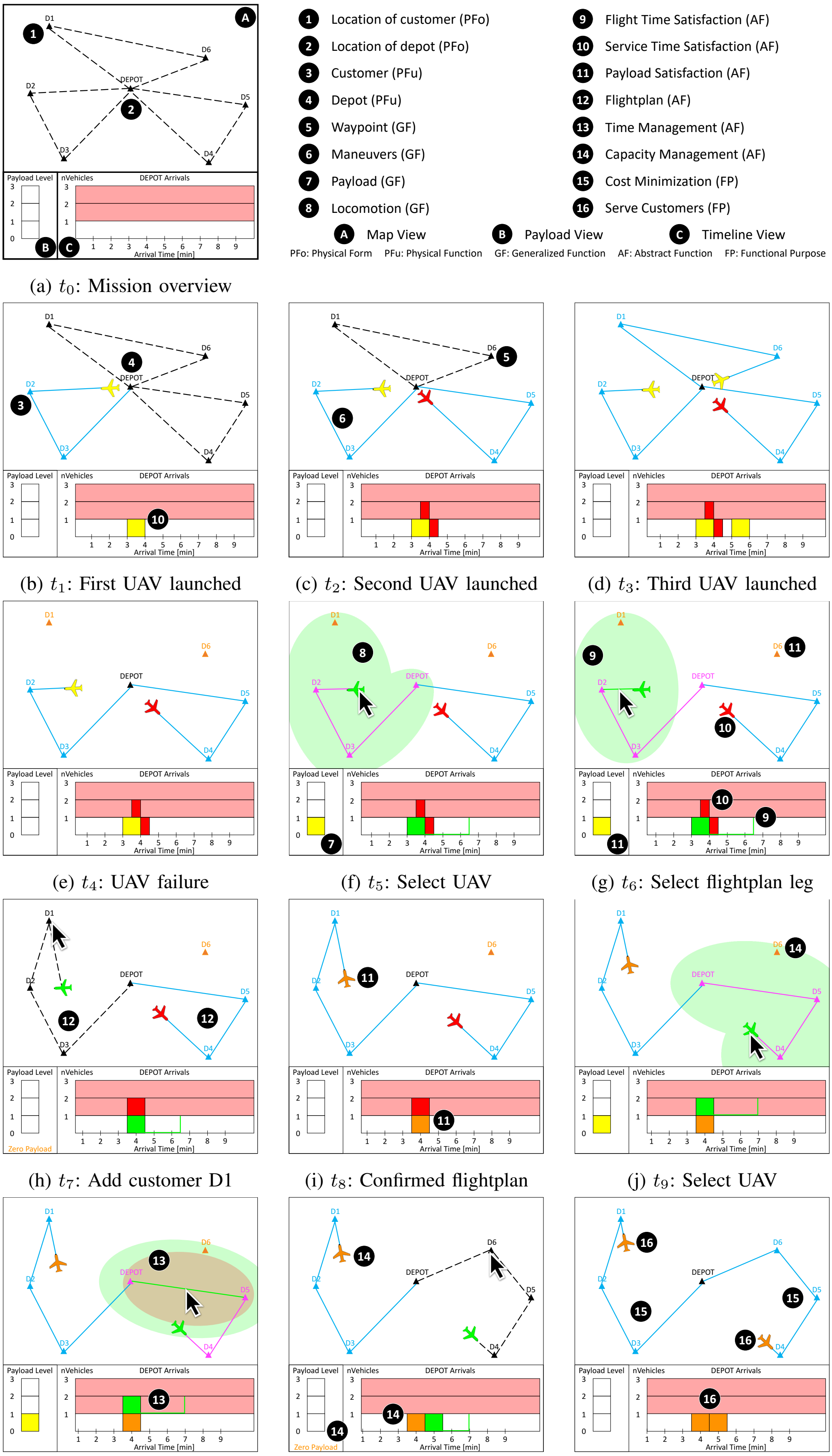
Fig. 6: Hierarchical Task Analysis for mission and perturbation management.

spatial perspective and the timeline view presents information from a temporal perspective. The red zone in the timeline view represents the depot capacity constraint.

At t_0 (Figure 7a) the interface provides a mission overview by displaying customer 1 and depot 2 locations and the pre-optimized flightplans 12 for each UAV. The flightplans of vehicles that have not yet left the depot are drawn using a dashed line to differentiate from the vehicles that have. The first UAV is launched from the depot 4 and flies to its first customer 3 at time t_1 (Figure 7b). The arrival time at the depot and corresponding service time is now also indicated 10 for this vehicle. The color of the UAV icon and the arrival time block correspond to the payload level of the vehicle, where bright yellow is used when all payload is available, dark yellow when payload capacity is reduced and amber is used when none is available. At t_2 (Figure 7c) the second UAV is launched, which arrival time overlaps with the first UAV launched and hence exceeds the depot capacity. Therefore, both the arrival time block in the timeline view as well as the UAV icon in the map view are colored red to assist in activating the operator to identify the problem and take action. Also, the vehicle's flightplan consists of waypoints 5 at the customer locations as well as any diversion waypoints that the operator my include for path stretching purposes. The future maneuvers a UAV will make are depicted by the guidance reference 6. The third and final UAV is launched at t_3 (Figure 7d). At t_4 (Figure 7e) one of the vehicles fails, disappearing from the interface and resulting in two customers not being served.

One vehicle is selected at time t_5 (Figure 7f), indicated by the green coloring of the UAV icon and the arrival time block. If a vehicle is selected, the payload window indicates the payload available 7. An envelope 8 around the guidance reference indicates what locations can be reached given the vehicle's energy status, which is a visualization of the vehicle's locomotion constraint. At t_6 (Figure 7g) a flightplan leg is selected and the corresponding flight time constraint is indicated 9 both in the map view and in the timeline view, where the vertical line indicates the maximum flight time. The red UAV icon and the arrival time overlap 10 indicate service time issues. The payload level of the vehicles in the fleet and the unvisited customers 11 yield information on the payload satisfaction. Customer D1 is included in the flightplan at t_7 (Figure 7h), where the updated flightplan 12 is indicated with a dashed line and the arrival time is updated. At t_8 (Figure 7i) the modified flightplan is confirmed and the UAV icon and arrival time block 11 color change to amber to indicate no more payload is available after visiting all assigned customers.

At t_9 (Figure 7j) the other UAV has sufficient payload capacity 14 to cover the remaining customer and is selected. Once a flightplan leg is selected at time t_{10} (Figure 7k), in addition to the flight time constraint, the required delay to solve the depot arrival time overlap is visualized 13. This combination gives an integrated overview of time management affordances. At t_{11} (Figure 7l) the modified flightplan is shown. Also, it can be observed that all UAVs in the fleet have now used up their full payload capacity 14. Finally, at t_{12} (Figure 7l) the flightplan is confirmed and the solution to the scenario is visible. Not only are all customers served 16, but this is also achieved using efficient routing of the vehicles 15, while adhering to all applicable constraints.



(a) t_0: Mission overview

(b) t_1: First UAV launched (c) t_2: Second UAV launched (d) t_3: Third UAV launched

(e) t_4: UAV failure (f) t_5: Select UAV (g) t_6: Select flightplan leg

(h) t_7: Add customer D1 (i) t_8: Confirmed flightplan (j) t_9: Select UAV

(k) t_{10}: Select flightplan leg (l) t_{11}: Add customer D6 (m) t_{12}: Confirmed flightplan

Fig. 7: Step-by-step overview of the interface workings and Abstraction Hierarchy links for a simple scenario.

B. SRK-Based Control Support

Simultaneous support for cognitive control at the skill-, rule- and knowledge-based levels is an important property of an ecological interface. Therefore, the way in which the levels of cognitive control are supported by the proposed interface is discussed below.

1) Skill-Based Behavior: SBB is driven by the perception of time-space signals that are directly used for control [35]. The interface is designed to allow for the operator to act directly on the interface by selecting and interacting with the various display elements, supporting SBB.

2) Rule-Based Behavior: RBB is dictated by signs that when perceived, trigger previously stored rules for familiar situations [35]. An example of an RBB sign is that by comparing the location of an unvisited customer with the ellipses around the vehicle's flight plan, the feasibility of visiting this location with respect to the vehicle's energy limitation can be directly perceived. If the customer location is covered by the ellipses, then it can be included in the flight plan.

3) Knowledge-Based Behavior: KBB is based on the perception of symbols that communicate meaningful information related to problem-solving activities for unfamiliar situations [35]. For example, the visualization of all vehicle routes, combined with the information on payload capacity and fleet energy status, allows for the development of rerouting strategies in case of any perturbations to the mission plan. Also, the edit mode gives the opportunity to evaluate different options and interpret the consequences.

V. Human-in-the-Loop Experiment

A human-in-the-loop experiment was performed to evaluate the functionality and scalability of the proposed interface. Also, human control performance in solving multi-UAV DCVRPs was investigated. Relevant DCVRP scenarios were created and simulated, where the interface was expected to offer support in achieving successful mission execution. Both objective and subjective experimental data are captured and analyzed to gather information on display usage and mission performance.

A. Participants

The experiment was performed with a total of sixteen participants, all of which are graduate students at TU Delft, with an average age of 25.38 (SD = 1.67). The group consisted of one female and fifteen males, also, fourteen out of sixteen participants were native Dutch speakers. Finally, seven out of sixteen participants considered themselves a regular video game player.

B. Independent Variables

The experiment design consisted of two within-subject independent variables, namely:

1) Payload capacity: The payload capacity of a single UAV, serving as a metric for DCVRP problem size, consisting of four levels: 4, 5, 6 and 7 payload items. All vehicles in a scenario have the same payload capacity.
2) Perturbation severity: The perturbation severity dictates how many UAVs will fail in the scenario, consisting of two levels: single and double failure. All vehicle failures occur at the same time at five seconds into the scenario.

TABLE I: Experiment Conditions

	Payload 4	Payload 5	Payload 6	Payload 7
Single UAV Failure	F1P4	F1P5	F1P6	F1P7
Double UAV Failure	F2P4	F2P5	F2P6	F2P7

The payload capacity variable dictates the size of the scenario, since from this metric, the amount of vehicles and the amount of customer locations are determined. Scenarios are designed in such a way as to require the full payload capacity of all post-failure vehicles to be able to serve all customers. Increasing the vehicle payload capacity will also result in an increased number of UAVs and an increased number of customers. The rationale for this variable is to, first, be able to investigate the scalability of the interface, second, to investigate human performance over a range of problems from simple to complex, and third, to investigate whether operator strategies differ as a function of problem size. Also, some of these severely constrained cases have been found to be challenging for the optimization algorithm, which sometimes requires impractically long solution times given the real-time application under consideration, or does not yield a solution at all. Therefore, it is of interest to investigate whether human performance for these cases is superior or possibly complementary to optimization.

The perturbation severity variable dictates the spatial spread of the replanning induced by the vehicle failures. In case of a single vehicle failure, focus can be put on accommodating the single cluster of customer locations that needs to be taken over by the remaining UAVs in the fleet. Whereas the double vehicle failure case results in two clusters of unassigned customer locations that need to be attended to. In case of the dual failure case, the customer clusters corresponding to the failed UAVs were positioned opposite of each other to prevent the two clusters from being combined into a single larger one. The rationale for this variable is to, first, be able to investigate the effects of spatial spread of vehicle failures on solution quality, and second, investigate the effects of spatial spread of vehicle failures on solution time.

C. Scenarios

Participants were asked to mitigate the effects caused by UAV failures during several multi-UAV payload delivery missions under the eight different experiment conditions, see Table I. The vehicle failures resulted in unassigned customer locations and the task of the participant was to include the unassigned locations into the flight plans of the remaining vehicles, while satisfying all constraints (flight time, payload capacity and depot capacity). Table II lists the number of customer locations and the number of vehicles per condition, both of which are uniquely dictated by the payload capacity, number of vehicle failures, and payload margin.

TABLE II: Customers and Vehicles per Condition

	nCustomers	nVehicles
F1P4	12	4
F1P5	20	5
F1P6	30	6
F1P7	42	7
F2P4	24	8
F2P5	40	10
F2P6	60	12
F2P7	84	14

TABLE III: Control Variables

Variable	Value
Max Flight Time (s)	750
Airspeed (m/s)	13
Service Time (s)	30
Scenario Duration (s)	360
Failure Times (s)	5
Payload Margin (-)	1
Sector Size (m^2)	5000 x 5000
Depot Capacity (-)	30% of nVehicles[2]

A balanced latin square design was used to order the experiment conditions such that carry-over effects between the scenarios are minimized. All participants performed each experiment condition twice to reduce variance in the results, resulting in 16 runs per participant. To prevent scenario recognition, the repeated scenarios were rotated 180-degrees with respect to the originals.

All scenarios lasted six minutes. In every scenario, UAVs were deployed in batches from the depot every thirty seconds (equal to the depot service time), with the batch size equaling the depot capacity. Only lateral control was available, by means of flight plan waypoint modification. Any control actions taken by the participant could influence the solution space later in the scenario. Finally, all UAV flight performance characteristics were simulated as a single generic aircraft type.

The scenarios were created by using an off-line VRP optimization algorithm. First, customer locations were generated using a random number generator, where a minimum distance criterion was implemented to prevent location clustering. Then, vehicle routes were obtained by optimizing the DCVRP that the scenario is based on. For the optimization, use was made of the Google Optimization Tools, which is a software suite for solving combinatorial optimization problems [39]. The algorithm finds a first solution by starting from a route start node, connecting it to the node which produces the cheapest route segment, then extending the route by iterating on the last node added to the route. Starting from this initial solution, a guided local search algorithm is used as the local search meta-heuristic, which is generally considered the most efficient for vehicle routing problems [40]. The use of a meta-heuristic local search algorithm requires the use of a time limit to stop the search. A time limit of two hours per scenario was used to generate the optimal routes for the experiment scenarios.

D. Control Variables

The control variables in the experiment are the UAV flight time, airspeed, and depot service time, the sector size, depot capacity, scenario duration, the time at which UAVs fail, and the amount of excess payload (i.e. payload margin) each vehicle has available. Table III summarizes the values associated with these control variables.

[2]Rounded to the nearest integer.

E. Dependent Measures

The following dependent variables were used to investigate the functionality and scalability of the interface, as well as human control performance:

1) Workload: Rating Scale Mental Effort (RSME) scores and the total number of clicks on the map view.
2) Control performance: Feasibility of the final solution, extra distance flown by the rerouted UAVs with respect to the optimized solution and solution time.
3) Strategy: Satisfice versus optimize and tactical versus strategic self assessments and the number of clicks on the map view as a function of time.
4) Interface usage: Display usage ratings for the map, timeline and payload views and participant comments.

F. Procedure

First, participants were requested to complete a short intake survey. This survey consisted of questions regarding age, gender, language and gaming activities. Also, a short spatial reasoning test consisting of six questions was performed as part of the survey to assess if any large differences in participant's spatial reasoning capabilities existed, which was considered to have potential impact on performance in the experiment. With a mean score of 5.56 (SD = 0.73) correct questions out of six, and an average time to complete the survey of 216.50 seconds (SD = 79.29), the spatial reasoning performance of the participant group was considered uniform.[3]

After the intake survey, participants were requested to read the experiment briefing, explaining the VRP, control goals, experiment setup, control inputs, and the interface. The task of the participant during the experiment was to mitigate the effects caused by UAV failures during several multi-UAV payload delivery missions. The control goals were twofold. First, assign all customer locations to the fleet of UAVs, while satisfying flight time, payload capacity and depot capacity constraints. Second, optimize all UAV routes for shortest distance. Also contained in the briefing package was step-by-step instructions describing the training scenarios. Participants used these instructions, together with the simulator to perform the training scenarios.

The training consisted of nine untimed scenarios. The first three training scenarios were used to familiarize the participant with all the features of the interface and the control inputs.

[3]The survey was not time limited and participants were also not instructed to complete the questions as fast as possible.

The remaining six training scenarios were used to train the participant in solving DCVRPs. Also, starting with training four, participants were instructed to complete a short post scenario survey after every run. This post scenario survey consisted of a Rating Scale Mental Effort (RSME) score [41], and two questions regarding the manner in which they solved the scenario, namely whether they would classify their approach as satisfice (achieving a solution that achieves the overall goal) or optimize (achieving the best solution to achieve the overall goal) and as tactical (using local solutions to achieve the overall goal) or strategic (executing a pre-defined plan). The six DCVRP training scenarios started out simple, with just three UAVs and six customers, and progressively got more complex, ending with a scenario consisting of nine UAVs and seventy two customers. The complexity of the last two training scenarios was approximately equivalent to the most complex scenarios in the experiment.

After the training was completed, participants started the experiment. In contrast to the training runs, the experiment runs had a fixed duration of six minutes. After each run, the participants were instructed to complete the same post scenario survey as was used during training.

The experiment ended with a post experiment survey. This survey required the participant to answer questions about the usefulness of the map, timeline and payload views. Also, the participant was asked to assess the usefulness and clarity of the color use in the display. Then, two scenarios, a simple and a complex one, were graphically presented and the participant was requested to provide a high-level description of how to solve the scenario. Finally, the participant was given the opportunity to provide comments or suggestions with respect to the interface or the experiment that were not yet covered in the preceding questions.

G. Apparatus

The experiment was conducted in the Air Traffic Management Laboratory (ATM Lab) of the Faculty of Aerospace Engineering at TU Delft. A dedicated software-based simulator, running on a single computer, was created that was used to present the interface and the scenario to the participant. The interface was presented on a 30-inch display (60-HZ LED, 2560 x 1600 pixels) positioned in front of the participant. Control inputs were given by means of a standard computer mouse and keyboard. A second display was used to present the surveys.

H. Hypotheses

It was hypothesized that with increasing payload capacity and increasing perturbation severity, the workload would increase, since more rerouting needs to be performed by the operator. Also, a decrease in control performance was expected, since the increased rerouting activities would leave less time for focusing on route efficiency. Furthermore, a shift from strategic and optimize to tactical and satisfice was hypothesized due to more control actions being required, leaving less time to reason about solution strategy and optimality. Finally, the interface usage was expected to become less efficient due to interface clutter caused by the visual representation of all vehicles, customer locations and routes.

I. Data Analysis

To test for within-group effects, the two-way repeated measures ANOVA is used in case the data are normally distributed. If not, the Friedman's ANOVA is used, with the Wilcoxon test as follow-up. For nominal data, the Pearson's chi-squared test is used. The significance level (α) has been set at 0.05 for all tests. A Bonferroni correction is used to control the Type I error when multiple hypotheses (m) are tested, by means of adjusting the significance level to α/m. For each participant, the data corresponding to the dependent variables associated with repeated conditions were averaged. Also, the results have been corrected for between participant variability, by subtracting the participant mean and adding the grand mean for each dataset.

VI. Results

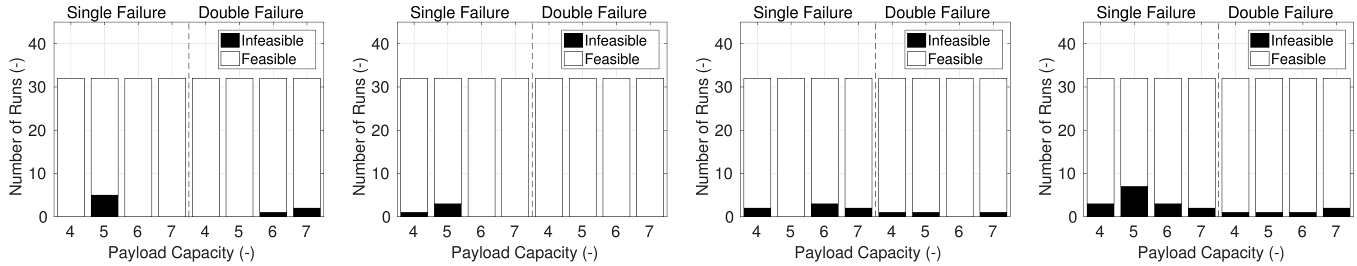
A. Workload

Figure 8 shows a box plot of the RSME scores per condition. During the experiment, it was observed that the main source of workload was the rerouting activities that were performed to obtain and initial solution after the UAV failures occurred. After the initial solution was obtained, only monitoring and post-optimization activities were observed. Analysis of the results shows that the RSME scores are not significantly affected by either the number of vehicle failures, the payload capacity, or the combination of payload capacity and failures.

Figure 9 shows a box plot of the total number of clicks in the map view per condition. It was observed during the experiment that participants required more interaction with the interface if the problem size increased (increase in payload capacity and increase in perturbation severity). From the data, it can be concluded that the experimental conditions significantly affect the total map view clicks (Friedman $\chi^2(7) = 74.479$, $p < .001$), reflecting this observation. The total map view clicks for the double failure case are found to be significantly different from the single failure case for the four (Bonferroni correction: $\alpha = .05/4 = .0125$, Wilcoxon: $z = -2.896$, $p = .004$, $r = -.51$), six (Wilcoxon: $z = -3.051$, $p = .002$, $r = -.54$) and seven (Wilcoxon: $z = -3.207$, $p = .001$, $r = -.57$) payload capacity conditions. Given the aforementioned results on the RSME scores, it can be concluded that although increasing the problem size leads to more clicks, it does not result in a significant increase in mental effort.

B. Control performance

Figure 10 shows a bar chart of the number of infeasible and feasible runs per condition. Infeasible runs were caused by not meeting the control goals: serving all customers, not overrunning the available flight time and not overrunning the depot capacity. Analysis shows that the experimental conditions do not significantly affect the number of infeasible and feasible runs.



(a) Not satisfying the goal of serving all customers.

(b) Not satisfying the goal of not overrunning the available flight time.

(c) Not satisfying the goal of not overrunning the depot capacity.

(d) Total count due to any combination of goals not being met.

Fig. 10: Bar chart of the infeasibility count per condition.

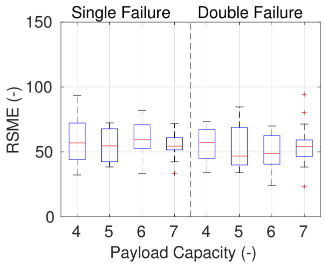


Fig. 8: Box plot of the Rating Scale Mental Effort (RSME) scores per condition.

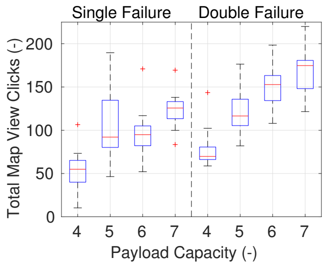
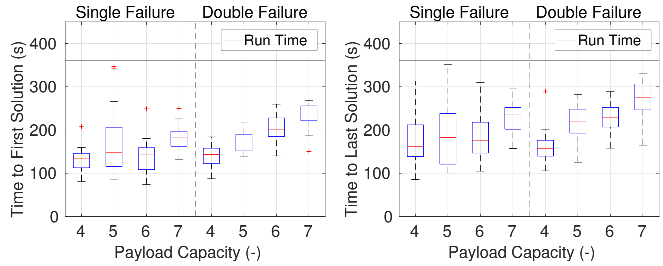


Fig. 9: Box plot of the total number of clicks in the map view per condition.



(a) Time to first solution

(b) Time to last solution

Fig. 11: Box plot of the time to obtain the first and last solution per condition.

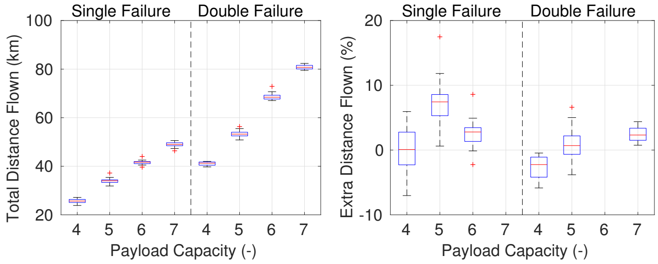
Figure 11a shows a box plot of the time to the first solution per condition. The time to the first solution is defined as the amount of time between the start of the scenario and the moment the first feasible solution was achieved. During the experiment, it was observed that the time to the first solution increased with increasing problem size. Analysis of the results shows that the experimental conditions significantly affect the time to the first solution (Friedman $\chi^2(7) = 56.354$, $p < .001$), supporting this observation. The time to the first solution for the double failure case is significantly different from the single failure case for the six (Bonferroni correction: $\alpha = .05/4 = .0125$, Wilcoxon: $z = -3.154$, $p = .002$, $r = -.56$) and seven (Wilcoxon: $z = -3.051$, $p = .002$, $r = -.54$) payload capacity conditions. Together with observing the trends in the figure and the analysis of the results, it can be concluded that for small problem sizes the time to first solution is relatively constant, whereas it increases for larger problem sizes.

Figure 11b shows a box plot of the time to the last solution per condition. The time to the last solution is defined as the amount of time between the start of the scenario and the moment the last changes were made to the solution. Some participants used a post-optimization phase in their strategy, which creates a difference between the time to the first and the time to the last solution. Analysis of the results shows that the time to the last solution is significantly affected by the amount of vehicle failures ($F(1, 15) = 11.289$, $p < .01$, $r = .64$) and the payload capacity ($F(3, 45) = 9.757$, $p < .001$, $r = .41$), but not by the combination of payload capacity and failures. Regarding the effect of payload capacity, only the difference between the four and five payload capacity conditions was found to be significantly different ($F(1, 15) = 49.277$, $p < .001$). From these results, it can be concluded that the time to the last solution is generally larger for the double failure conditions compared to the single failure conditions.

Figure 12 shows box plots of the total distance flown by the UAVs in the fleet per condition and the percentage extra distance flown over a condition specific optimized solution generated by the DCVRP optimization algorithm that was also used to generate the scenarios.[4] Figure 12a shows an increase in total distance flown with increasing payload capacity and perturbation severity. This is expected, since an increase in either variable results in a larger DCVRP problem size. It can also be observed that the rate of increase is higher for the double failure versus the single failure condition.

Figure 12b generally shows an increase in the percentage of extra distance flown with increasing payload capacity. The optimization of the F1P7 and F2P6 conditions yielded no feasible solutions and hence no extra distance flown can be obtained for these conditions. As can be observed, participant solutions outperform the optimized solutions for some conditions. Since for the optimization use is made of a meta-heuristic algorithm, finding the global optimum is not guaranteed. Therefore, it is possible for participants to find a more efficient solution with respect to the optimized solution. However, especially in the low payload capacity conditions, participants were observed to be using path stretching to solve depot capacity conflicts. The optimization algorithm did not have this capability and

[4]The optimization for each condition had a time limit of 30 hours.



(a) Total distance flown for participant solutions.

(b) Percentage of extra distance flown over optimized solutions.

Fig. 12: Box plots of the sum of the distance flown and the sum of the extra distance flown over the optimized solution by the UAVs in the fleet per condition. No optimized solution was obtained for condition F1P7 and F2P6.

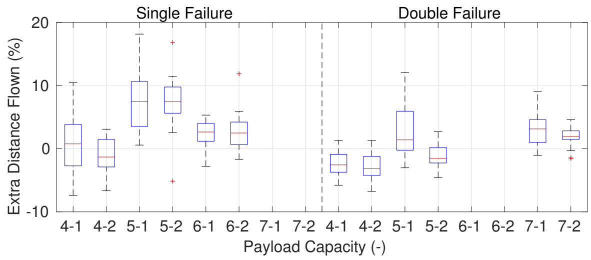


Fig. 13: Percentage of extra distance flown over optimized solutions, with first and second experiment condition runs separated to assess training effects.

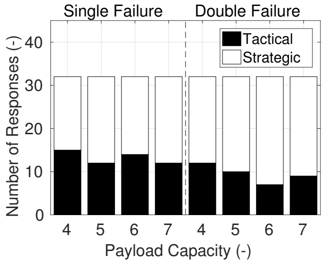


Fig. 14: Bar chart of the tactical versus strategic participant ratings per condition.

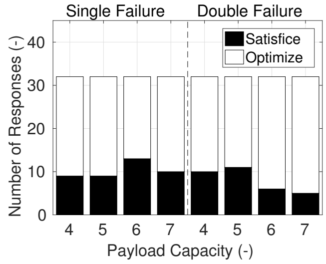
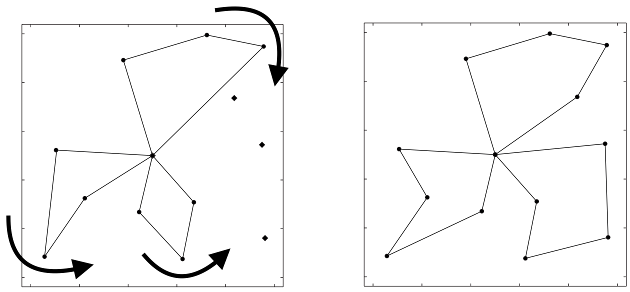


Fig. 15: Bar chart of the satisfice versus optimize participant ratings per condition.



(a) Before rerouting

(b) After rerouting

Fig. 16: Mission status for an example scenario with a single vehicle failure with arrows illustrating identified effective control strategy.

has to settle for a less efficient solution to comply with the depot capacity constraint, which is probably the case for the majority of cases where the participant solution outperforms the optimized solution. On average, the extra distance flown in the participant solutions is less than 2% of the optimized solution distances.

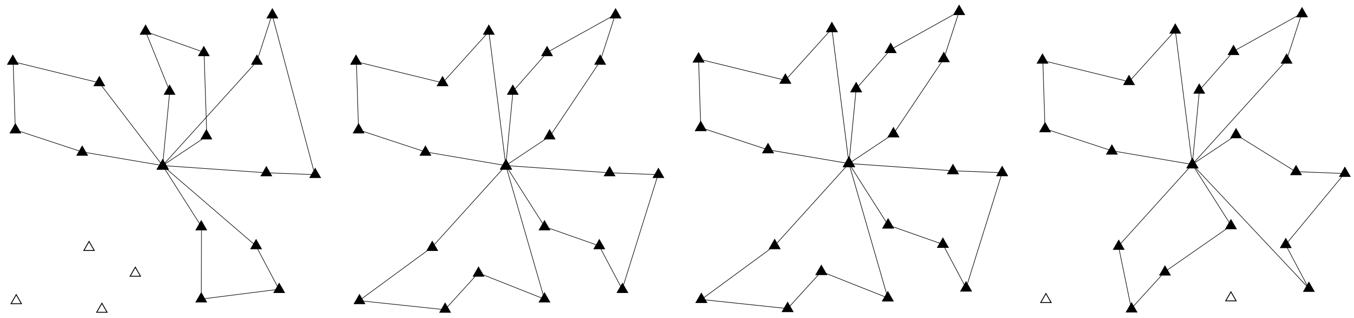
To address possible training effects during the experiment, Figure 13 shows the percentage of extra distance flown with the repeated conditions split up. Some training effects can be observed, since the variance for the first run is clearly larger compared to the second run for conditions F1P4, F1P5, F2P5 and F2P7. This indicates the training runs that were conducted before the start of the experiment might not have been sufficient to sufficiently stabilize participant performance.

C. Strategy

Figure 14 shows a bar chart of the tactical versus strategic ratings per condition and Figure 15 shows a bar chart of the satisfice versus optimize ratings per condition. During the experiment and from post-survey data it was observed that participants generally first start with a strategic plan with the goal to satisfy the mission objectives. Then, during the scenario, participants transitioned to a tactical approach when they found their strategy to be ineffective. Some participants were satisfied by just satisfying mission objectives, whereas others were observed to also perform an active optimization phase after achieving a first solution. However, analysis of the results shows that the experimental conditions do not significantly affect the tactical versus strategic and satisfice versus optimize self assessments.

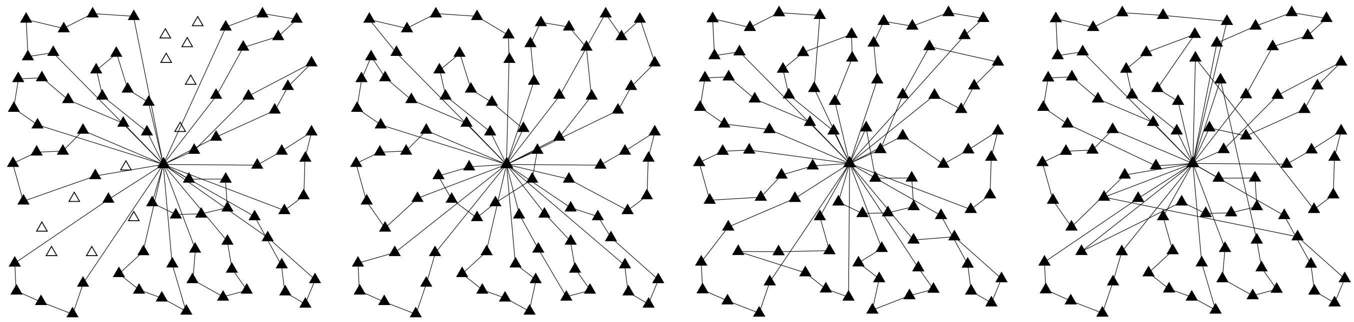
The strategy that most participants converged to was dictated by the realization that if a vehicle fails, the flight plans of the remaining vehicles need to be shifted in the direction of the cluster of unvisited customers that results from the UAV failure. Effectively, this means the routes next to the unvisited customer cluster need to be changed to include those locations. This route-shifting strategy is repeated for the remaining available UAVs until all customer locations are included in the mission plan, see Figure 16 for an illustration of this process.

Figure 17 and Figure 18 depict the scenario, optimized solution and two participant solutions for condition F1P5 and F2P7, respectively. Due to the UAV flight time constraints, the conditions with few customers have a smaller solution space compared to conditions with many customers. Hence, participants generally either found one out of a small set of solutions, or were unable to solve the scenario. Figure 17c shows a participant solution that is the same as the optimized solution. Figure 17d shows a participant's best attempt, but in this case a solution was not obtained because the participant got stuck due to ineffective customer clustering. For conditions with many customers, participants with a good strategy were able to come up with solutions that are visually similar to the optimized solution, see Figure 18c, but rarely exactly the same because of the large solution space. Participants with a bad strategy, or participants who focused on satisfice over



(a) Perturbed scenario (b) Optimized solution (c) Participant 13 solution (d) Participant 4 solution

Fig. 17: Selection of results for condition F1P5.



(a) Perturbed scenario (b) Optimized solution (c) Participant 6 solution (d) Participant 1 solution

Fig. 18: Selection of results for condition F2P7.

optimize generally opted for solutions that visually look more chaotic, as depicted in Figure 18d.

Figure 19 shows box plots of the map view clicks over time for each condition. The time axis is discretized into thirty second time windows, each representing the amount of clicks that were measured in the corresponding time frame. Also indicated is the time after which all UAVs have been launched from the depot. As can be observed from the figures, each click distribution roughly has the same shape, a quick ramp-up at the start of the scenario, followed by a slow ramp-down starting around the time all vehicles have been launched. Increasing payload capacity stretches the ramp-down and increasing the perturbation severity from single to dual failure generally increases the magnitude. The first part of the curve (ramp-up and and ramp-down) corresponds to the input required to obtain the first solution, the remainder (flatter section) of the curve corresponds to post-optimization activity. It can be concluded that the peaks in control input take place just before or around the time all vehicles are launched.

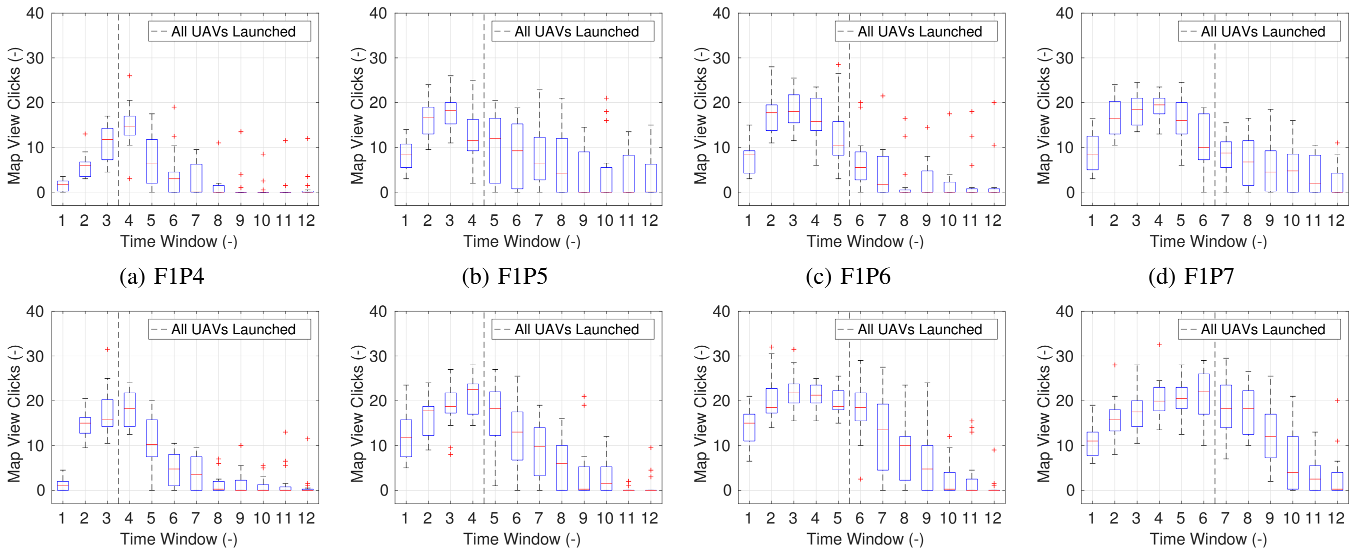
D. Interface Usage

Figure 20a shows a box plot of the map view display usage rating per condition. Analysis of the results shows that the map view display usage is significantly affected by the payload capacity ($F(3, 45) = 3.208$, $p = .032$, $r = .22$), but not by the number of vehicle failures and the combination of payload capacity and failures. Regarding the effect of payload capacity, only the difference between the four and five payload capacity conditions was found to be significantly different ($F(1, 15) = 6.784$, $p = .020$). From these results it can be concluded that for the four payload capacity cases, more attention was paid to the payload and timeline views than for the five payload capacity cases.

Figure 20b shows a box plot of the payload view display usage rating per condition. Analysis of the results shows that the payload view display usage is not significantly affected by either the number of vehicle failures, the payload capacity or the combination of payload capacity and failures. Figure 20c shows a box plot of the time line view display usage rating per condition. Analysis of the results shows that the time line view display usage is not significantly affected by either the number of vehicle failures, the payload capacity or the combination of payload capacity and vehicle failures.

VII. Discussion

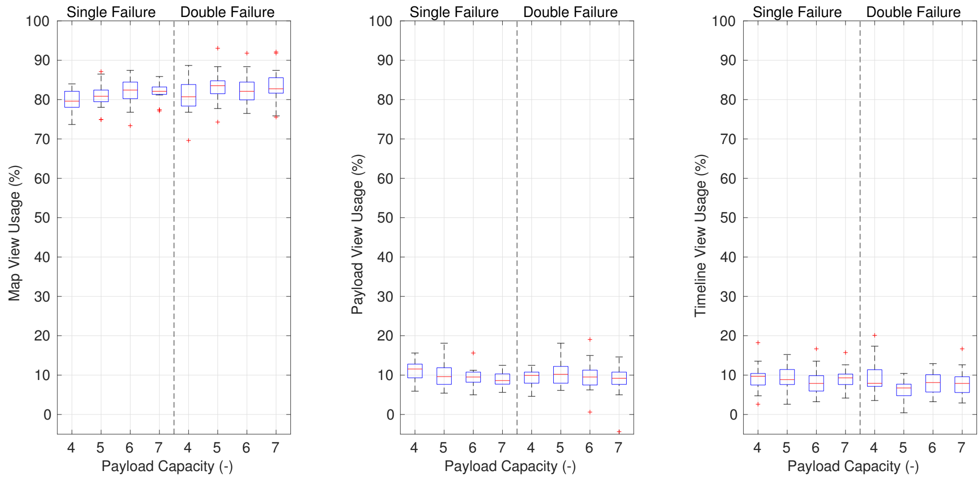
The developed interface has shown to be generally effective for human control in solving multi-UAV VRPs. The constraint-based EID approach to the design of the interface allowed operators to choose their own desired way to solve the control task, although many converged to a similar strategy. Some participants focused on satisfying the mission objectives and did not extensively consider solution optimality. So although the EID-based interface allows operators to choose their own strategy, this does in this case affect the optimality of the solutions. On the other hand, participants were able to come up with solutions in a short amount of time for all scenarios, whereas the optimization algorithm was not. Clearly, a trade-off is to be made between solution time and optimality.



(a) F1P4 (b) F1P5 (c) F1P6 (d) F1P7

(e) F2P4 (f) F2P5 (g) F2P6 (h) F2P7

Fig. 19: Box plots of the map view clicks versus time window (30 seconds) for all conditions.



(a) Map view (b) Payload view (c) Timeline view

Fig. 20: Box plots of the view usage ratings for all conditions.

Whether or not a human-in-the-loop offers an advantage is largely dictated by solution time available and solution time required by the optimization.

The interface elements were generally considered useful by the participants. Especially the map view with the ellipses visualizing the vehicle flight time constraint was essential for participants to effectively perform reroutings. The payload and timeline view however, were considered less useful. The payload view was located too far from the map view, requiring excessive attention shifting to observe the indications. Also, some participants did not look at this view at all, since they fully relied on the UAV color coding indicating payload capacity in the map view. Others did use the information, primarily to reason about fleet payload capacity and used this information in their rerouting activities. Hence, relocating this information closer to the map view, or including it in the map view itself (for example by having info labels attached to each vehicle) could be a possible improvement. The timeline view was only used for getting an overview of any depot capacity constraint violations, and thus was of limited use. For scenarios requiring 4D navigation involving for example Required Time of Arrival (RTA) constraints at customer locations, the timeline view could probably more useful. In these cases, temporal constraints play a more prominent role and appropriate mapping of these constraints onto the timeline view could be essential for effective mission control.

With regards to participant strategies, the satisfice versus optimize and tactical versus strategic self assessments were observed to be an unreliable way to gain insight into them. Although specific definitions of each term were provided to participants, during the experiment they were sometimes observed to be choosing the opposite answer from the strategy they were observed to be executing. Post-survey data and observations made during the experiment did yield an insight into participant problem solving strategies.

Some participants were observed to not fully converge, or converge slowly to the most used route shifting control strategy. Hence, to increase human performance, more strict training might be employed, where operators are specifically instructed on possible successful strategies. Also, since it was observed that not all participants reached steady-state performance at the start of the experiment runs, an increase in training volume should be considered.

Although it was expected that participants would have difficulty solving the complex scenarios, while the number of control inputs and solution time increased, workload remained constant. Based on post-survey feedback, the amount of control actions required to reroute UAVs was increased unnecessarily by system design. To be able to reassign waypoints between UAVs, they first needed to be unassigned, then assigned. By changing this such that waypoints can be directly exchanged between the vehicles, some portion of control inputs can be avoided, resulting in some reduction in solution time, and hence, better interface scalability. Furthermore, more complex scenarios should be investigated to better determine the limits of human control performance.

Participant solutions were compared with optimized solutions, but not only the optimality of the solution is relevant for operational mission success. For example, solution robustness was not considered, which is generally an important metric for the safety of flight operations. More control performance metrics could be developed and investigated to better assess the advantages of having a human operator in the control loop.

Finally, only a single meta-heuristic optimization algorithm was used to construct baseline optimized solutions to the experiment scenarios. However, other types of algorithms, not necessarily optimization based, could yield improvements in solution time and reliability compared to the algorithm currently used. Further investigation into alternative algorithms for automated vehicle rerouting and comparing them to human operator performance could provide more insight in automation related challenges related to dynamic VRPs as well as the potential control performance improvements obtained by having a human-in-the-loop.

VIII. Conclusion

The goal of this study was to investigate human control performance in multi-UAV DCVRPs. First, an ecological interface was designed and implemented to allow for human control of VRPs. Then, this interface was used in a simulated environment to assess human control performance in various DCVRP scenarios of varying problem size. In the experiment, the payload capacity and perturbation severity were manipulated. The interface design and human control actions were evaluated in terms of workload, control performance, strategy, and interface usage. Results show that the developed interface scales well from small to large problem sizes, but at a price of losing some efficiency. On average, vehicles in the fleet were found to fly less than 2% extra distance in the participant solutions compared to the optimized solutions. It was shown that human operators were able to effectively control perturbed DCVRPs across the problem size range. For at least some of the conditions under consideration, the optimization algorithm needed 30 hours of solution time, and for other conditions was unable to find a solution at all. This indicates that having a human-in-the-loop using an ecological interface can be beneficial in some cases to achieving real-time VRP solutions faster and more reliably over relying purely on optimization algorithms, especially in time critical situations.

References

[1] J.-f. Cordeau, G. Laporte, M. W. P. Savelsbergh, and D. Vigo, "Vehicle Routing," *Handbooks in operations research and management science*, vol. 14, pp. 367–428, 2007.

[2] C. Lin, K. L. Choy, G. T. S. Ho, S. H. Chung, and H. Y. Lam, "Survey of Green Vehicle Routing Problem: Past and future trends," *Expert Systms with Appications*, vol. 41, no. 1, pp. 1118–1138, 2014.

[3] A. Larsen and O. B. G. Madsen, "The dynamic vehicle routing problem," Ph.D. dissertation, 2000.

[4] G. Laporte, "What You Should Know about the Vehicle Routing Problem," *Naval Research Logistics (NRL)*, vol. 54, no. 8, pp. 811–819, 2007.

[5] ——, "The vehicle routing problem: An overview of exact and approximate algorithms," *European journal of operational research*, vol. 59, no. 3, pp. 345–358, 1992.

[6] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia, "A review of dynamic vehicle routing problems," *European Journal of Operational Research*, vol. 225, no. 1, pp. 1–11, 2013.

[7] P. J. Smith, E. C. McCoy, and C. Layton, "Brittleness in the Design of Cooperative Problem-Solving Systems: The Effects on User Performance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 27, no. 3, pp. 360–371, 1997.

[8] P. Toth and V. Daniele, *Vehicle routing: problems, methods, and applications*. Society for Industrial and Applied Mathematics, 2014.

[9] H. C. Lau, M. Sim, and K. M. Teo, "Vehicle routing problem with time windows and a limited number of vehicles," *European journal of operational research*, vol. 148, no. 3, pp. 559–569, 2003.

[10] N. B. Sarter and B. Schroeder, "Supporting Decision Making and Action Selection under Time Pressure and Uncertainty: The Case of In-Flight Icing," *Human factors*, vol. 43, no. 4, pp. 573–583, 2001.

[11] P. Krolak, W. Felts, and G. Marble, "A Man-Machine Approach Toward Solving the Traveling Salesman Problem," *Communications of the ACM*, vol. 14, no. 5, pp. 327–334, 1971.

[12] P. Krolak, W. Felts, and J. Nelson, "A Man-Machine Approach Toward Solving the Generalized Truck-Dispatching Problem," *Transportation Science*, vol. 6, no. 2, pp. 149–170, 1972.

[13] A. V. Hill, "An Experimental Comparison of Human Schedulers and Heuristic Algorithms for the Traveling Salesman Problem," *Journal of Operations Management*, vol. 2, no. 4, pp. 215–223, 1982.

[14] J. N. Macgregor and T. Ormerod, "Human performance on the traveling salesman problem," *Attention, Perception, & Psychophysics*, vol. 58, no. 4, pp. 527–539, 1996.

[15] D. Anderson, E. Anderson, N. Lesh, J. Marks, K. Perlin, D. Ratajczak, and K. Ryall, "Human-Guided Simple Search: Combining Information Visualization and Heuristic Search," in *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference on Information and knowledge maangement. ACM*, 1999, pp. 21–25.

[16] D. Anderson, E. Anderson, N. Lesh, J. Marks, B. Mirtich, D. Ratajczak, and K. Ryall, "Human-Guided Simple Search," in *AAAI/IAAI*, 2000, pp. 209–216.

[17] D. Vickers, M. Butavicius, M. Lee, and A. Medvedev, "Human performance on visually presented Traveling Salesman problems," *Psychological Research*, vol. 65, no. 1, pp. 34–45, 2001.

[18] S. D. Scott, N. Lesh, and G. W. Klau, "Investigating Human-Computer Optimization," in *Proceedings of the SIGCHI conference on Human factors in computing systems. ACM*, 2002, pp. 155–162.

[19] M. Dry, M. D. Lee, D. Vickers, and P. Hughes, "Human Performance on Visually Presented Traveling Salesperson Problems with Varying Numbers of Nodes," *The Journal of Problem Solving*, vol. 1, no. 1, pp. 20–32, 2006.

[20] G. Y. Tütüncü, C. A. C. Carreto, and B. M. Baker, “A visual interactive approach to classical and mixed vehicle routing problems with backhauls,” *Omega*, vol. 37, no. 1, pp. 138–154, 2009.

[21] J. N. Macgregor and Y. Chu, “Human Performance on the Traveling Salesman and Related Problems,” *The Journal of Problem Solving*, vol. 3, no. 2, pp. 1–29, 2011.

[22] K. J. Vicente and J. Rasmussen, *A Theoretical Framework for Ecological Interface Design*, 1988.

[23] ——, “Ecological Interface Design : Theoretical Foundations,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, no. 4, pp. 589–606, 1992.

[24] K. J. Vicente, “Ecological interface design: progress and challenges.” *Human factors*, vol. 44, no. 1, pp. 62–78, 2002.

[25] C. C. Murray and A. G. Chu, “The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery,” *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 86–109, 2015.

[26] D. Bamburry, “Drones: Designed for product delivery,” *Design management Review*, vol. 26, no. 1, pp. 40–48, 2015.

[27] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, “The vehicle routing problem: State of the art classification and review,” *Computers & Industrial Engineering*, vol. 99, pp. 300–313, 2016.

[28] H. N. Psaraftis, M. Wen, and C. A. Kontovas, “Dynamic Vehicle Routing Problems: Three Decades and Counting,” *Networks*, vol. 67, no. 1, pp. 3–31, 2016.

[29] K. J. Vicente, *Cognitive work analysis: Toward safe, productive, and healthy computer-based work*. CRC Press, 1999.

[30] J. Rasmussen, “The role of hierarchical knowledge representation in decisionmaking and system management,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-15, no. 2, pp. 234–243, 1985.

[31] ——, “Outlines of a hybrid model of the process plant operator,” in *Monitoring behavior and supervisory control*. Springer US, 1976, pp. 371–383.

[32] R. Kilgore and O. St-Cyr, “The SRK inventory: a tool for structuring and capturing a worker competencies analysis,” pp. 506–509, 2006.

[33] M. Mouloua, R. Gilson, and P. Hancock, “Human-Centered Design Of Unmanned Aerial Vehicles,” *Ergonomics in Design*, no. Winter 2003, pp. 6–11, 2003.

[34] R. Parasuraman and V. Riley, “Humans and automation: Use, misuse, disuse, abuse,” *Human factors*, vol. 39, no. 2, pp. 230–253, 1997.

[35] J. Rasmussen, “Skills Rules and Knowledge, Other Distinctions in Human Performance Models,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 13, no. 3, pp. 257–266, 1983.

[36] C. A. Miller and K. J. Vicente, “Toward an integration of task-and work domain analysis techniques for human-computer interface design,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 42, no. 3, pp. 336–340, 1998.

[37] J. Annett, “Hierarchical task analysis,” in *Handbook of cognitive task design*, 2003, pp. 17–35.

[38] G. A. Jamieson, C. A. Miller, W. H. Ho, and K. J. Vicente, “Integrating task- and work domain-based work analyses in ecological interface design: A process control case study,” *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, vol. 37, no. 6, pp. 887–905, 2007.

[39] “Google Optimization Tools.” [Online]. Available: https://developers.google.com/optimization/

[40] C. Voudouris and E. Tsang, “Guided local search and its application to the traveling salesman problem,” *European Journal of Operational Research*, vol. 113, no. 2, pp. 469–499, 1999.

[41] F. R. H. Zijlstra, “Efficiency in work behaviour: A design approach for modern tools,” Ph.D. dissertation, TU Delft, 1993.

Part II

Thesis Book of Appendices

The content in this chapter has been graded as part of the preliminary thesis report under AE4020.

Appendix A

Literature Study

This chapter presents a review of literature on the vehicle routing problem in Section A-1, human supervisory control in Section A-2, ecological interface design in Section A-3 and hierarchical task analysis in Section A-4. Also, previous work on multi-UAV ecological interfaces is discussed in Section A-5. At the end of this chapter, conclusions are presented in Section A-6.

A-1 Vehicle Routing Problem

The VRP is at the core of many logistics applications. The generic family of VRPs can be defined as follows:

"Given a set of transportation requests and a fleet of vehicles, determine a set of vehicle routes to perform all transportation requests with the given vehicle fleet at minimum cost; in particular, decide which vehicle handles which requests in which sequence so that all vehicle routes can be feasibly executed." (Toth & Daniele, 2014)

Various types of VRPs exist, each with its own specific set of constraints. The DCVRP includes, in addition to the generic VRP attributes, a distance constraint for each vehicle modeling fuel limitations and a capacity constraint modeling payload capacity. More formally, the transportation requests in the DCVRP consist of the distribution of goods from a single depot, denoted as point 0, to customers, which are defined as a set of n other points, with $N = \{1, 2, ..., n\}$. The customer demand, $q_i \geq 0$, is defined as the amount of goods that needs to be delivered to the customer $i \in N$. The fleet of vehicles used to distribute the goods, defined as $K = \{1, 2, ..., |K|\}$, is assumed homogeneous. The homogeneity entails that the $|K|$ vehicles in the fleet each have identical distance constraints, capacity $Q > 0$ constraints, and associated cost. Each vehicle starts at the depot, delivers goods to a subset of customers $S \subset N$ visiting customer locations only once, then returns to the depot. When traveling from customer i to customer j, the vehicle incurs the travel cost c_{ij}. Cost is assumed symmetric, where the cost of traveling from i to j is equal to the cost of traveling from j to i.

The VRP can be formulated as follows (Laporte, 1992):

$$\text{minimize} \quad \sum_{i \neq j} c_{ij} x_{ij} \tag{A-1}$$

$$\text{subject to} \quad \sum_{j=1}^{n} x_{ij} = 1 \quad (i = 1, ..., n), \tag{A-2}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad (j = 1, ..., n), \tag{A-3}$$

$$\sum_{j=1}^{n} x_{0j} = |K| \quad (j = 1, ..., n), \tag{A-4}$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - v(S) \quad (S \subset N; |S| \neq 0), \tag{A-5}$$

$$x_{ij} \in \{0, 1\} \quad (i, j = 1, ..., n; i \neq j). \tag{A-6}$$

Equation A-1 represents the total cost of all vehicle routes and is the cost function to be minimized. Then, Equation A-2 and A-3 ensure that each customer vertex is connected to two other vertices. Similarly, Equation A-4 dictates there is K vertices leaving the depot, corresponding to one route for each vehicle in the fleet. Equation A-5 serves as the capacity constraint and subtour elimination constraint (ensuring a vehicle route is joint and does not consist if separated tours), where $v(S)$ represents the minimum amount of vehicles required to serve the customers in S. Finally, Equation A-6 defines the decision variable x_{ij}, which is equal to 1 if an only if the arc (i, j) is part of the optimal solution.

Many algorithms exist, both exact and (meta)heuristic, with the goal of optimizing and approximating solutions to VRPs. Typically, performance of these algorithms is good for problems with a sufficiently large solution space and when sufficient time is available to perform the computations. Severely constrained problems, over-constrained problems, and little available computation time are generally problematic for the algorithms, which results in the inability to obtain a solution.

A-2 Human Supervisory Control of Multiple UAVs

This section provides information on literature relevant to the topic of human supervisory control of multiple UAVs. First, the supervisory control loops are presented in Section A-2-1, followed by discussions on operator workload and situation awareness in Section A-2-2 and Section A-2-3, respectively. Finally, the effects of introducing automation will be reviewed in Section A-2-4.

A-2-1 Supervisory Control

In supervisory control, as illustrated in Figure A-1, a human operator supervises a computer system, which in turn controls a process. Since the human is not directly controlling a process, some form of information processing and automation is present on the computer system. Usually, for some variables at least some of the time, the computer system itself closes an automatic control loop, leading to the human operator's role of guiding the automation (Sheridan, 1992). In this role, the human observes the actions performed by the automation, assesses the quality and desirability of the actions and if necessary intervenes and reprograms the automation for more appropriate behavior.

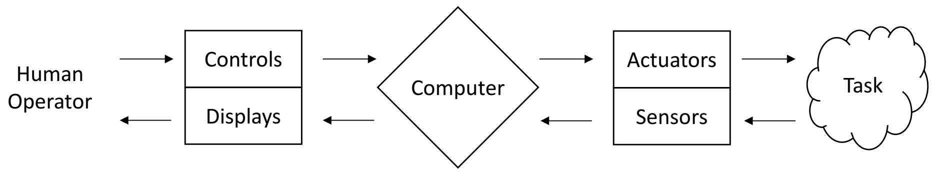


Figure A-1: Human supervisory control, adapted from Sheridan and Verplank (1978).

Sheridan (1992) identifies five time-sequential functions of the human supervisor: planning, teaching, monitoring, intervening and learning. In planning, the human decides on a strategy and allocates and prioritizes tasks. Then, it teaches or instructs the computer about the plan, monitors the automation and identifies anomalies in the plan execution and possible failures of the system. At operators discretion interventions might take place to re-guide or overrule the automation with the objective to increase performance or in case of emergency situations. Finally, the operator learns from the experiences as to increase performance in the future. Figure A-2 visualizes these human supervisor functions as nested loops. Clearly, any interface between human and automated system should aim to support the identified functions.

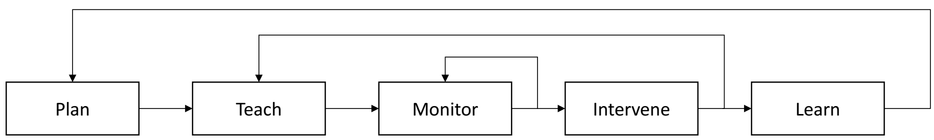


Figure A-2: Human supervisor functions as nested loops, adapted from Sheridan (1992).

Not only can control loops be defined for human supervisory functions, but also the command and control of a multi-UAV system can be represented by equivalent means. Cummings, Bruni, et al. (2007) present such a control architecture, which is visualized in Figure A-3. The figure presents N UAVs with accompanying local low level control loops and an overarching global mission and payload management loop. In the local loops, the most inner loop is for basic guidance and flight control. The second loop is the navigation loop and is responsible for obstacle avoidance and routes to waypoints. The global loop offers mission-level control where information from sensors and payload is interpreted and guidance on the overall goals and execution of the mission is provided. Also, global system health and status monitoring is represented by comparing UAV state information with nominal performance models.

For this research, which is focused on the mission planning and disturbance and failure man-

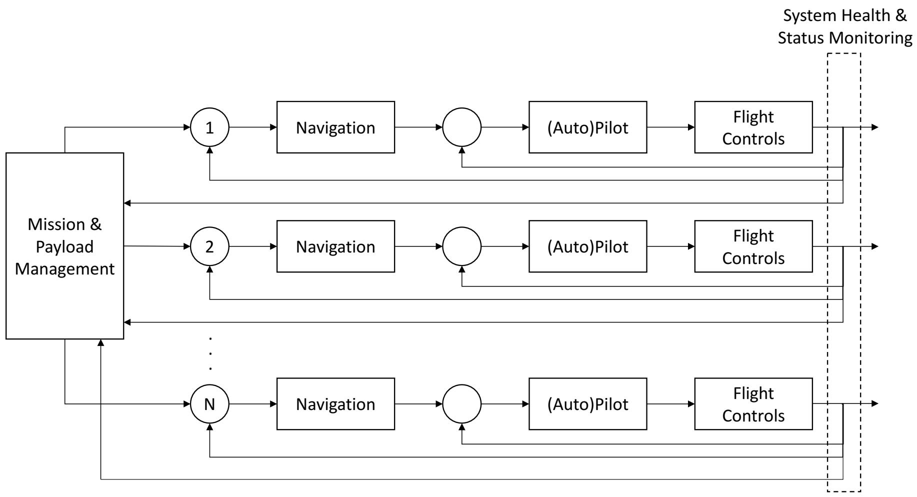


Figure A-3: Multi-UAV system control loops, adapted from Cummings, Bruni, et al. (2007).

agement by the human operator, the navigation and flight control loops are automated. Not only does this limit the scope of the research, but it is also a requirement for the single human operator to have sufficient cognitive resources available for satisfactory control performance.

A-2-2 Operator Workload

In the field of multi-UAV supervisory control, achieving acceptable operator workload is of significant importance for overall mission effectiveness and deserves critical attention in the design of interfaces and displays (Mouloua, Gilson, Kring, & Hancock, 2001). Workload is really an overarching concept, a broad area, which Jahns (1973) subdivides into three functionally relatable attributes: input load, operator effort and performance. Input load relates to factors or events that are external to the human operator, operator effort is concerned with internal factors, and performance is relates to adequacy of the data outputs generated by the human operator (Johannsen, 1977). Figure A-4 illustrates the workload attributes together with a number of measures of performance.

In man-machine systems evaluation, the input load is mainly a predetermined factor, dictated by experiment and man-machine system design, whereas effort and performance are generally subject of assessment (Johannsen, 1977). Performance measures are related to well-established and well-known signal processing and data analysis techniques and hence will not be further discussed here. As can be seen from Figure A-4, operator effort measures can be subdivided into four categories: time-line analyses, information processing studies, operator activation-level studies, and subjective effort ratings.

In time-line analyses, execution times of all events related to task execution, such as task recognition and task duration, are analyzed. Information processing studies represent the human operator as an information processing element and assess effort by introducing secondary tasks or by the application of control or information theory. Operator activation-level studies consider the use of physiological cues for estimating effort. Finally, subjective effort

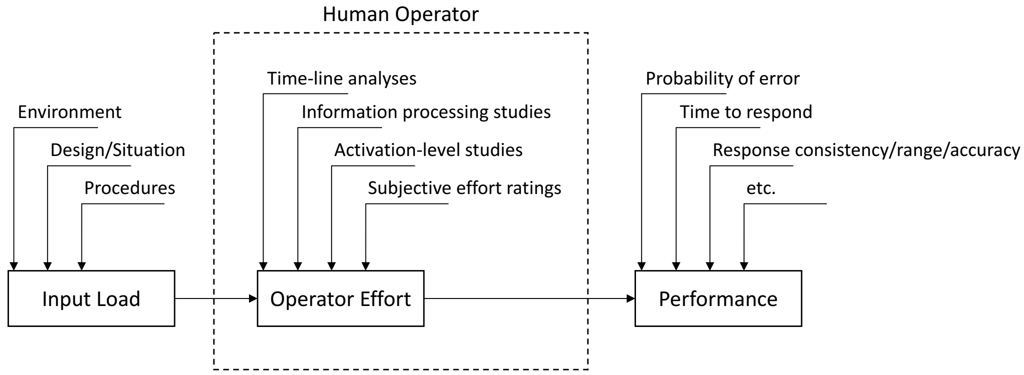


Figure A-4: Human operator workload attributes and performance measures, adapted from Jahns (1973) and Johannsen (1976).

ratings are techniques based on questionnaires to establish information on operator effort. Hart and Staveland (1988) developed one of the major accepted and well-known subjective effort ratings, the National Aeronautics and Space Administration (NASA)-Task Load Index (TLX).

A-2-3 Situation Awareness

Maintaining a sufficient level of Situation Awareness (SA) of the overall mission and the individual UAVs is essential for achieving successful multi-UAV supervisory control (Chen, Barnes, & Harper-Sciarini, 2011). Endsley (1988) defines SA as follows:

"Situation awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future."

From the definition, Endsley (1995) defines three levels of situation awareness: perception of data and elements in the environment, comprehension of the current situation and projection of future states and events, Level 1, 2 and 3 SA, respectively. To obtain SA, information from the environment needs to be obtained, this process is illustrated by Figure A-5. From the real world, information flows into the system. However, the system generally does not observe all information available in the real world. Then, the system provides information to the interface. Analogously, the interface generally does not display all information available to the system. Finally, the information available from the interface and the information directly acquirable from the environment might be incomplete or inaccurate after transmission to the human operator due to perceptual, attention, and working memory constraints.

Endsley (1995) presents a number of system characteristics that are hypothesized to improve operator SA, which are summarized below:

1. The degree of availability of relevant environmental features to the operator, presented directly or through an interface, affects the ability of a person to obtain situation awareness.

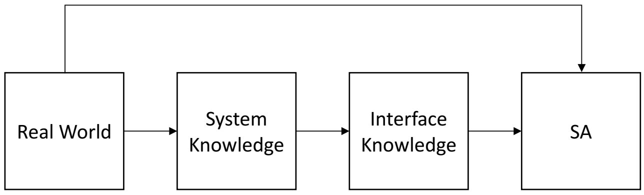


Figure A-5: Situation awareness inputs, adapted from Endsley (1990).

2. Situation awareness is affected by the way in which information is presented. Hypothesized features that improve situation awareness: presentation of integrated and goal oriented information, salience of important cues, parallel information processing support, no unneeded information and salience reduction of noncritical information, presentation of global information irrespective of current goal and goal specific detailed information and future event and state projection support.

3. Automation has a negative effect on SA if used for human decision making and active system control, but has a positive effect if used for peripheral tasks.

SA and operator workload are not independent concepts. Rather, operator workload can influence SA. Not only can SA decrease when the operator is under high workload due to lack of sufficient cognitive resources (Andre & Wickens, 1995), but SA can also decrease when the operator experiences low workload due to boredom and complacency (Rodgers, Mogford, & Strauch, 2000).

A-2-4 Effects of Automation

Often, automation is introduced to aid the operator in performing assigned tasks, reducing workload and preventing information overload (Cummings, Bruni, et al., 2007; Cummings et al., 2010; Prinet et al., 2012; Ruff et al., 2002). Research shows that despite the positive intentions, automation solutions can also degrade human performance instead of improve it (Wickens, Mavor, & James, 1997; Wickens, Mavor, Parasuraman, & Mcgee, 1998), leading to an increase in workload (Wiener, 1988) and a decrease in SA (Sarter & Woods, 1992, 1994a, 1994b). Also, if too much automation is introduced, negative consequences, such as over-reliance on the automation and loss of skills to perform the task manually in case of automation failure can be expected (Chen et al., 2011).

A-3 Ecological Interface Design

This section discusses the Ecological Interface Design (EID) framework, which is used as a basis for the interface design efforts in this research. This section starts by discussing the Skill, Rule, Knowledge (SRK) taxonomy in Section A-3-1. Section A-3-2 elaborates on the general EID framework and current applications, then Section A-3-3 discusses the cognitive work analysis.

A-3-1 Skill, Rule, Knowledge Taxonomy

Rasmussen (1983) developed a qualitative human performance model, which can be used to guide the overall design of a human-machine interface structure. Rasmussen (1983) identifies three different levels of human performance by categorizing human behavior: skill-, rule-, and knowledge-based performance. These performance levels and how they relate to one another are depicted in a simplified form in Figure A-6. Skill-based behavior manifests itself as a level of control without conscious attention from the human as a very smooth, automated and integrated form of control. Rule-based behavior is driven by a stored rule or procedure which is applied for control in a familiar situation. These rules or procedures may be obtained from previous experience, might be communicated in advance as instructions, or might result from conscious problem solving and planning activities. Finally, knowledge-based behavior is used in unfamiliar situations, where the human explicitly formulates a goal and develops a useful plan to achieve that goal. For this, the human has a mental model that represents the internal structure of the controlled system.

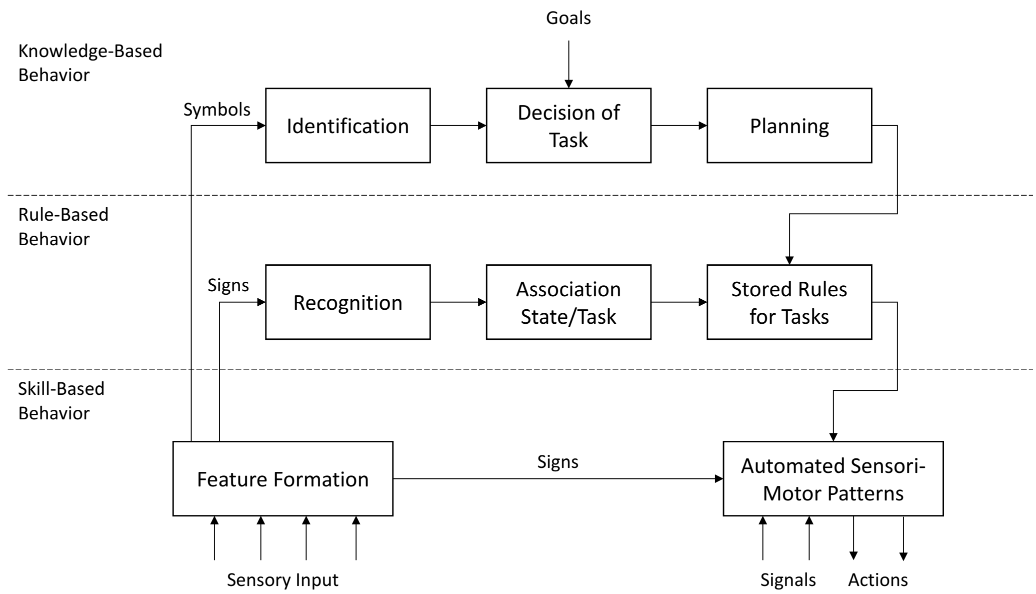


Figure A-6: Simplified representation of the three levels of performance of skilled human operators, adapted from Rasmussen (1983).

According to Rasmussen (1983), each of the three aforementioned human performance levels leads to a different way in which available information is processed: signals, signs or symbols, respectively. This difference is generally not caused by a different representation of information, but rather as a difference in information interpretation. Signals are sensory data and are processed as continuous variables, signs indicate the state of the environment and serve to activate stored rules or procedures and finally, symbols are used for causal functional reasoning and are defined by and refer to the internal system representation of the human.

A-3-2 Ecological Interface Design Framework

Vicente and Rasmussen (1992) propose a theoretical framework for the development of interfaces for complex human-machine systems. The Ecological Interface Design (EID) framework is based on the SRK taxonomy developed by Rasmussen (1983). The goal of the framework is to support the design of human-machine interfaces that do not force cognitive control to a higher level than required for the task at hand, and to provide sufficient support for cognitive control at the skill-, rule- and knowledge-based levels. The interface design problem can be structured into two basic questions. First, given a complex work domain, how should the domain complexity be described? To answer this question, a domain representation framework that describes work domain constraints is required. Second, how should an interface to a complex human-machine system communicate the information. This requires a model of human information processing to be able to achieve information representations that are in line with human cognitive and perceptual properties. This structure of the interface design problem is visualized in Figure A-7.

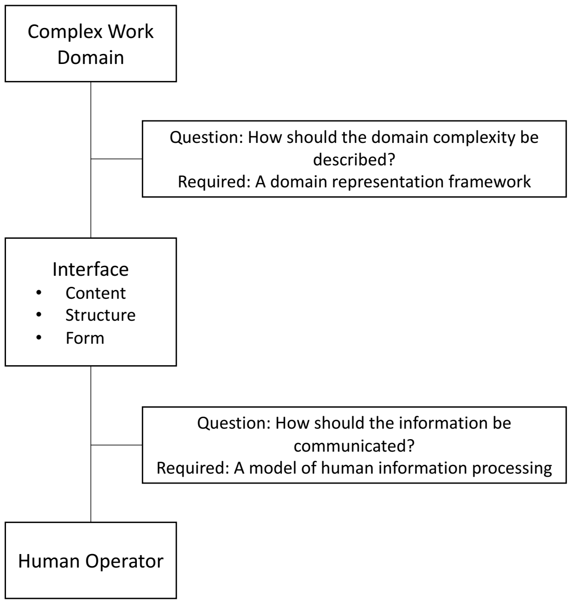


Figure A-7: Structure of the interface design problem for complex human-machine systems, adapted from Vicente and Rasmussen (1992).

To answer the second question, it is proposed to use the SRK taxonomy as a framework for describing human information processing. To answer the first question, the CWA (Vicente, 1999) framework is proposed as a means to obtain information on work domain constraints. This framework will be discussed in more detail in Section A-3-3.

EID aims to facilitate in the design of single interface designs that simultaneously support all of the SRK-based control levels. For each of the three levels of cognitive control, the EID framework postulates three general principles, which are summarized below (Vicente & Rasmussen, 1992):

1. Skill-based behavior: allow for direct user interaction on the interface, and structure the presented information isomorphic to the part-whole structure of movements.

2. Rule-based behavior: map work domain constraints one-to-one to cues or signs in the interface.

3. Knowledge-based behavior: offer the operator an externalized mental model by representing the work domain as an abstraction hierarchy.

Over the years, Ecological Interface Design (EID) has been applied to the design of human-machine interfaces in numerous complex work domains, such as process control (Reising & Sanderson, 2002), health care (Mcewen, Flach, & Elder, 2014), command and control (Hall, Shattuck, & Bennett, 2012), and aviation (Amelink, Mulder, van Paassen, & Flach, 2005; Dinadis & Vicente, 1999; Ellerbroek, Visser, Van Dam, Mulder, & van Paassen, 2011).

A-3-3 Cognitive Work Analysis

Vicente (1999) defines Cognitive Work Analysis (CWA) as a formative framework for the identification of requirements – both technological and organizational – that need to be satisfied for effective work support, targeted at the unique demands of complex sociotechnical systems. The framework consists of five concepts with corresponding modeling tools focused on the identification of work domain constraints. These five concepts, illustrated in Figure A-8, aim to represent different layers of constraints through performing work domain, control tasks, strategies, social-organizational and worker competencies analyses. All five concepts are discussed individually in more detail below.

Also depicted in Figure A-8 is the gradual transition from ecological to cognitive considerations. In the CWA, first the environment is analyzed, since associated constraints must be dealt with to achieve reliable and effective performance. Then, the analyses will shift focus towards satisfying cognitive constraints.

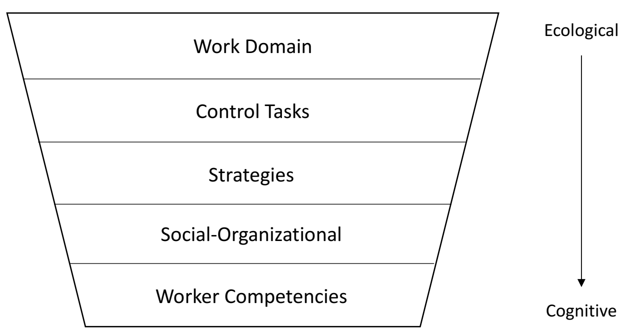


Figure A-8: Five concepts of the CWA framework and transition from ecological to cognitive considerations, adapted from Vicente (1999).

Work Domain Analysis

The Work Domain Analysis (WDA) is probably the most important concept in CWA, since it offers a description of the structure of the controlled system independent of operator, automation or interface. This is essential in the development of interfaces that are robust to unfamiliar events, which are, by definition, not considered during interface design. The modeling tool used for this phase is the abstraction-decomposition space as developed by Rasmussen (1985), the structure of which is illustrated in Figure A-9. It offers hierarchical structuring along two dimensions: the part-whole dimension, which relates parts and subsystems to the system as a whole, and the means-end dimension, which offers levels of abstraction for the purposes of the system. The part-whole dimension of the abstraction hierarchy is a natural hierarchical representation, and hence is not further discussed here. The means-end dimension, however, is more complicated and will be elaborated upon in the following.

Decomposition / Abstraction	Total System	Subsystem	Function Unit	Subassembly	Component
Functional Purpose					
Abstract Function					
Generalized Function					
Physical Function					
Physical Form					

Figure A-9: Abstraction-decomposition space modeling tool, adapted from Rasmussen (1985).

Rasmussen (1985) subdivides the means-end dimension in a number of levels:

1. Functional purpose: high level goals of the system.
2. Abstract function: mass and energy balances governing the underlying processes.
3. Generalized function: generic functions that control the mass and energy balances.
4. Physical function: functions provided by system components.
5. Physical form: spatial location and appearance of system components.

Bottom-up, the abstraction hierarchy provides a physical basis for capabilities, resources and causes of malfunction, whereas top-down, it provides reasons for proper function, requirements and a purpose basis (Rasmussen, 1985). According to the EID framework, the hierarchical work domain representation offered by the abstraction hierarchy should be implemented in the interface design to offer the operator an externalized mental model of the system functioning, which is essential in providing support for unanticipated events.

Control Task Analysis

A Control Task Analysis (CTA), as discussed by (Vicente, 1999), complements a work domain analysis. Whereas a work domain analysis is focused on providing a functional description of the work domain and aid in identifying work domain constraints, a control task analysis is concerned with describing known tasks associated with a particular application and identify constraints and requirements associated with those. As such, the control task analysis should focus on what needs to be done, not how or by whom.

The modeling tool for this phase is the decision ladder, developed by (Rasmussen, 1976), which can be considered a template for decision processes. Important characteristic is the separation of knowledge states (circles) and information processing activities (squares). Where the knowledge states are defined as the results of the information processing activities. This is an important distinction because it has implications for the possible shortcuts that can be made between them.

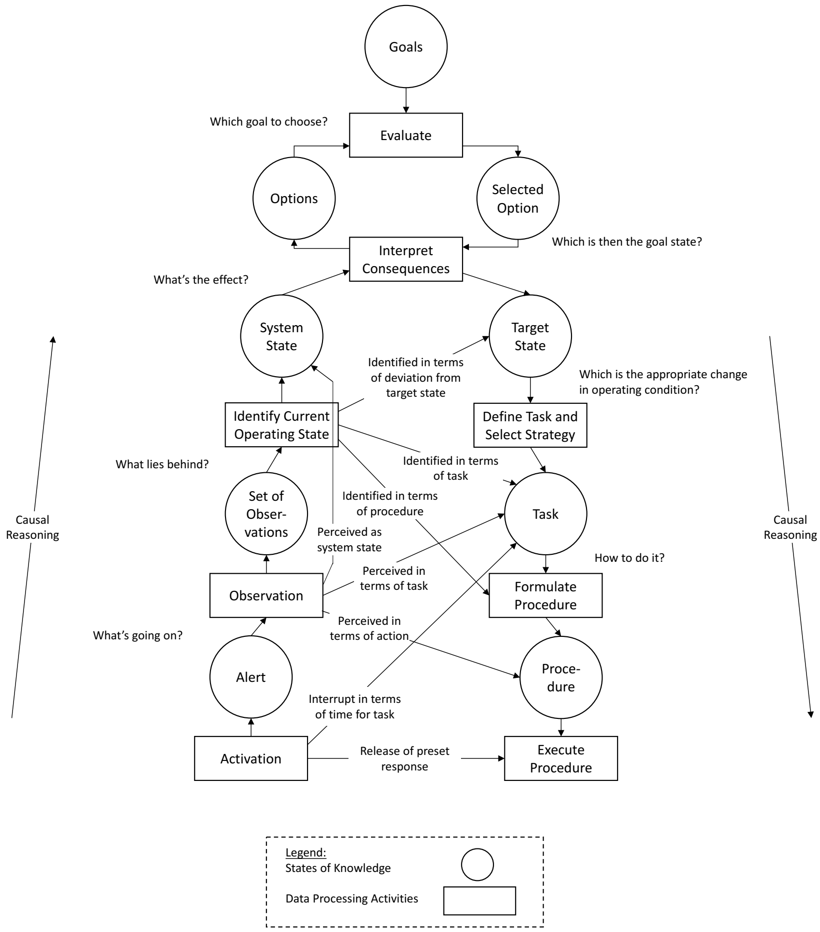


Figure A-10: Decision ladder with some possible shortcuts indicated, adapted from Rasmussen (1976).

Two types of shortcuts, which are often used by expert operators, can be mapped onto the decision ladder: shunts and leaps. Shunts link an information processing activity to a state of knowledge, hence representing skipping parts of the decision process. Leaps relate two states of knowledge together, representing a direct association between the two. Another important remark that needs to be considered is that cognitive activity does not necessarily start in the bottom left of the ladder. Likewise, it does not need to end at the bottom right. Different tasks can lead to different start and end points. Final remark that needs to be made is on the relationship between the decision ladder and the abstraction-decomposition space. These tools are complementary, since each helps in the identification of unique constraints. Basically, the abstraction-decomposition space is a representation of the work domain, which provides information to the control tasks that need to be performed, represented by the decision ladder, the outputs of which act on the work domain.

Strategies Analysis

Next step in the CWA is a strategies analysis (Vicente, 1999). After the previous discussion on the control task analysis, which focuses on what tasks need to be performed, the strategies analysis considers different strategies for how tasks can be executed, as depicted in Figure A-11. These strategies should not be considered as action sequences that switch between different strategies, but rather as different categories of strategies. Since the first approach would not allow for enough flexibility in the designed system to effectively deal with disturbances encountered in open systems. Because operators use strategy switching to keep workload at an acceptable level, it is important to allow for enough flexibility in the system design for the operator to do so, while still offering enough detail to effectively support each of the strategies.

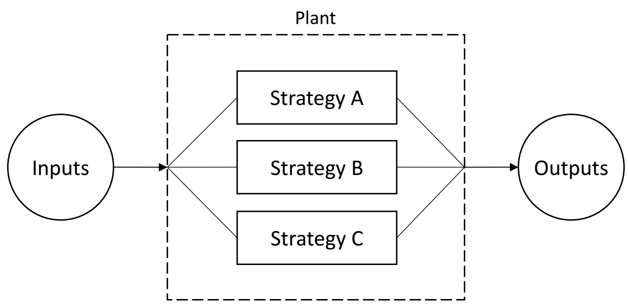


Figure A-11: Strategies analysis, adapted from Vicente (1999).

Rasmussen (1980) proposes the use of information flow maps as a modeling tool for this phase, but unfortunately this tool is not as mature as the previously discussed tools and has not been described in detail as a generic tool. Information flow maps should not be interpreted as a sequential representation of cognitive steps, but rather as a process representation, and should be used as such to describe identified strategies.

Social Organization and Cooperation Analysis

In the social organization and cooperation analysis (Vicente, 1999), task assignment is analyzed. In the previous discussions, techniques to analyze what tasks need to be completed and how are considered. This phase focuses on who should perform the tasks. This could be multiple operators in different roles with different responsibilities, but also the allocation of tasks between operator and computer. To do this, the different actors can be mapped onto the abstraction hierarchy, decision ladders or information flow maps that are the result of previous analyses. The results of this phase could be useful in the creation of teams of operators and associated hierarchy and communication requirements, but also could be used to create interfaces tailored to the specific information requirements of each actor.

Worker Competencies Analysis

The final phase of the CWA framework considers a worker competencies analysis (Vicente, 1999), which is an analysis of the competencies operators need to have to effectively function. This analysis is based on the SRK taxonomy developed by Rasmussen (1983), which is discussed in detail in Section A-3-1. This taxonomy is used as a framework to identify and support the operator competencies for skill-, rule- and knowledge-based behavior. As already mentioned in Section A-3-1, cognitive behavior at all of the three aforementioned levels has to be supported by the system to allow for effective operator performance. All while not forcing cognitive control to a higher level than required.

A-4 Hierarchical Task Analysis

Although work domain-centered approaches, such as presented in the aforementioned discussion offer tools for analyzing the functional structure of a work domain, the actions an actor can or should take to accomplish the control objective are not explicitly considered. This objective has traditionally been the focus of task analysis techniques. Miller and Vicente (1998) propose an integrated approach, where task- and work domain analysis techniques are combined to yield complementary results. It is shown that since both approaches offer different perspectives on the knowledge requirements for human-centered system design, a unification of the approaches offers a more complete set of knowledge for good interface design. The work domain analysis can provide the information requirements related to dealing with unanticipated events and the task analysis can provide information requirements to deal with predictable tasks.

Jamieson, Miller, Ho, and Vicente (2007) provide an industrial demonstration of an ecological interface designed based on an integration of task- and work domain based analyses. The task analysis is focused on an analysis of hierarchical means-ends relationships, relating to how a set of subtasks allow for achieving a higher level goal, and sequential relationships, which consider task related temporal requirements. In their analysis, Jamieson et al. (2007) propose the use of the Hierarchical Task Analysis (HTA) to complement the EID CWA methods. This tool will be further discussed next.

The HTA is widely used for interface design and error analysis in for example the power generation and command and control domains (Ainsworth & Marshall, 1998; Kirwan & Ainsworth, 1992; Shepherd, 2001). The procedure of an HTA consists of the decomposition of tasks into subtasks to any desired level of detail. The aim is to be able to relate what operators do and why they do it, and to identify the consequences in case this is not done correctly (Annett, 2003). The results of the analysis are captured in a table or diagram. All suboperations that are identified in the analysis must be mutually exclusive and exhaustive, which means the superordinate operation is completely defined (Annett, 2003). The diagrammatic representation is well suited to clearly communicate the functional structure of the task, whereas the tabular format is more suited towards recording supplementary information.

An example of an HTA diagram is depicted in Figure A-12, where the top-level goal is indicated at the top and suboperations listed below. The connecting lines between tasks and subtasks illustrate the action means-ends, which illustrate what actions need to be performed in order to achieve the higher level tasks. Also, temporal requirements are indicated using the following symbols: > indicates a sequence, / represents an either/or decision, + indicates dual or parallel operations, and : represents multiple operations in which timing and order are not critical (Annett, 2003).

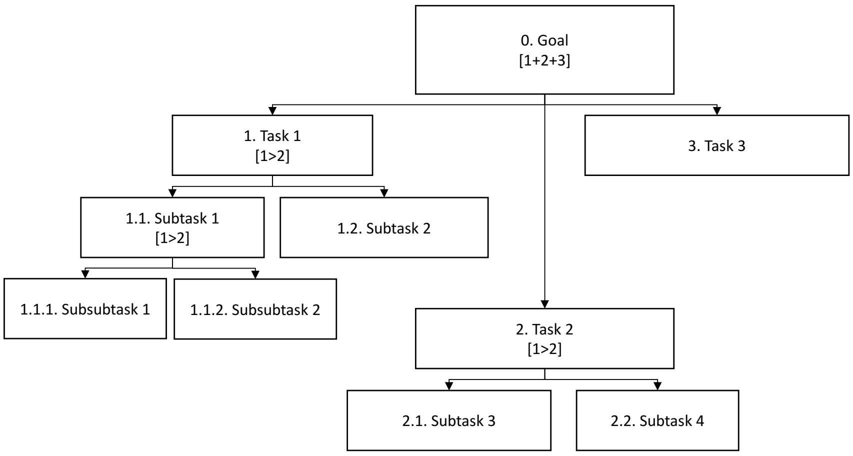


Figure A-12: Hierarchical task analysis diagram

A-5 Previous Research on Multi-UAV EID

As previously mentioned, much research in the field of multi-UAV supervisory control is focused on increasing levels of automation and human-automation interaction (Cummings et al., 2010; Prinet et al., 2012; Ruff et al., 2002), not on interface design. As such, only two recent studies on multi-UAV Ecological Interface Design (EID) have been identified, which will be discussed in this section.

Fuchs et al. (2014) developed a computer based human-machine interface for a simplified search mission consisting of four UAVs. Starting from a pre-existing interface, which contains low level status information, information about remaining range and weather conditions is added and their propagation through the joint mission plan is visualized. The interface was evaluated in an evaluation study, where ten participants performed five different missions, all starting from a predefined flight plan. Disturbances were introduced through battery anomalies and changing wind conditions for both a single and multiple UAVs. To solve the scenarios, participants had control over the number, altitude and position of all waypoints. The results show that the coloring of waypoints and lines between waypoints based on the projected battery state of charge is a very useful interface feature, since it was used for problem identification. However, since it was decided to show the flight plans of all UAVs concurrently, without clear distinctive features allowing matching of waypoints to UAVs, operators were forced to rely on unintended ways of UAV identification. Also, low level information was not found useful, since participants identified and resolved problems at high levels of abstraction, not considering the root cause of the problem.

van Lochem et al. (2015) developed a touch screen, tablet based human-machine interface to support building and maintaining multi-UAV communication relay networks. The interface presents information on communication range, sensor coverage and low level information. An evaluation study with ten participants was conducted to evaluate the interface. Six scenarios were designed with the control objective to obtain maximum sensor ground coverage at locations of interest, without losing the communication link between UAV and ground station. Each UAV was either tasked with providing sensor coverage at locations of interest or functioning as a communication relay. The results show that, due to missing preview information, participants had difficulties in assessing the consequences of their actions, leading to difficulties in maintaining relay networks and the prevention of separation conflicts. Also, the comparison of UAV state information between the vehicles was considered difficult, since low level information was only displayed for a single vehicle at a time. Finally, concerns regarding the distinction between UAVs, interface clutter, and scalability remained.

A-6 Conclusions

Supervisory multi-UAV control still faces many challenges related to achieving acceptable operator workload and SA. Although automation is needed, too much can lead to degraded human performance and a reduction in SA. The EID framework has shown its value in supporting effective human-machine interfaces in multiple domains (process control, health care, command and control, and aviation). In the framework, focus is put on constraint based analyses, leading to interfaces that support the human operator in dealing with unexpected and unforeseen circumstances, such as disturbances or failures. Also, hierarchical task analysis has demonstrated its value in being a complementary analysis next to the EID framework for obtaining information requirements to deal with predictable tasks.

In this chapter, the vehicle routing problem was discussed and theory on supervisory control, EID, cognitive work analysis, and hierarchical task analysis have been investigated. Contributions in the form of lessons learned, as well as modeling tools and frameworks have been identified and discussed. Also, previous research in the field of multi-UAV ecological interfaces has been reviewed, from which it can be concluded that extensive research is still required to achieve effective and satisfactory human control performance. Challenges regarding identification of problematic UAVs from a fleet, interface clutter and scalability should be important considerations in any such activity.

The content in this chapter has been graded as part of the preliminary thesis report under AE4020.

Appendix B

Work and Task Analysis

In this chapter, an analysis, by means of a Cognitive Work Analysis (CWA) and Hierarchical Task Analysis (HTA), is presented that yields information on constraints and operator tasks that need to be considered in the design of a multi-UAV ecological interface for a generic payload delivery mission. This chapter is structured as follows: first, Section B-1 discusses the scope of the analysis, then, Section B-2 presents the Work Domain Analysis (WDA), Section B-3 discusses the Control Task Analysis (CTA), Section B-4 elaborates on the strategies analysis, Section B-5 presents the social organization and cooperation analysis, Section B-6 discusses the worker competencies analysis, and, finally, Section B-7 presents the Hierarchical Task Analysis (HTA). At the end of this chapter, conclusions are presented in Section B-8.

B-1 Scope

The first step in the analysis is to define the task and assumptions that govern the scope of the CWA and HTA. The mission to be executed by the UAVs is the delivery of payload to a number of pre-defined delivery locations. All vehicles are deployed from a joint home base, which is the same location to which they should return at the end of their flight. Since there are multiple delivery locations, multiple UAVs will be deployed simultaneously to expedite mission execution. The use of multiple UAVs will also make the mission execution more robust, since the vehicles can take over each others tasks in case any anomalies arise. An example mission is depicted in Figure F-1.

The following assumptions are made with respect to the mission, vehicles and environment:

- Homogeneous UAV fleet
- Homogeneous payload
- Single UAV can carry multiple payload items
- UAVs fly at constant airspeed

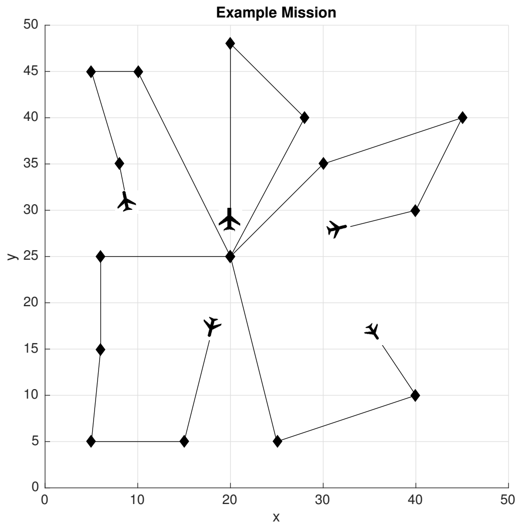


Figure B-1: Example multi-UAV payload delivery mission, with black diamonds indicating delivery locations, aircraft symbols indicating UAV positions and solid lines indicating guidance reference.

- Vehicles are vertically separated from each other and the ground
- Constant and uniform wind
- Single home base
- UAVs have a pre-defined flightplan

Since every UAV has a pre-defined flightplan, the task of the operator is to manage disturbances and anomalies, as introduced by for example wind or a lack of energy. This is done by modifying UAV flightplans to achieve feasibility, while ensuring overall mission success.

B-2 Work Domain Analysis

An Abstraction Decomposition Space (ADS) is created for the multi-UAV payload delivery work domain, which is depicted in Figure B-2. From this, constraints associated with the work domain can be identified and analyzed. Each level of abstraction will be elaborately discussed from the bottom up in the following.

Functional Purpose

At the functional purpose level, the high level purpose of the system is represented, which is depicted with means-ends links in Figure B-3. The overall purpose is to achieve the mission goals of serving customers with minimal cost in a safe manner. These purposes are achieved through Command, Control, Communication (C3), capacity management and time management on the abstract function level.

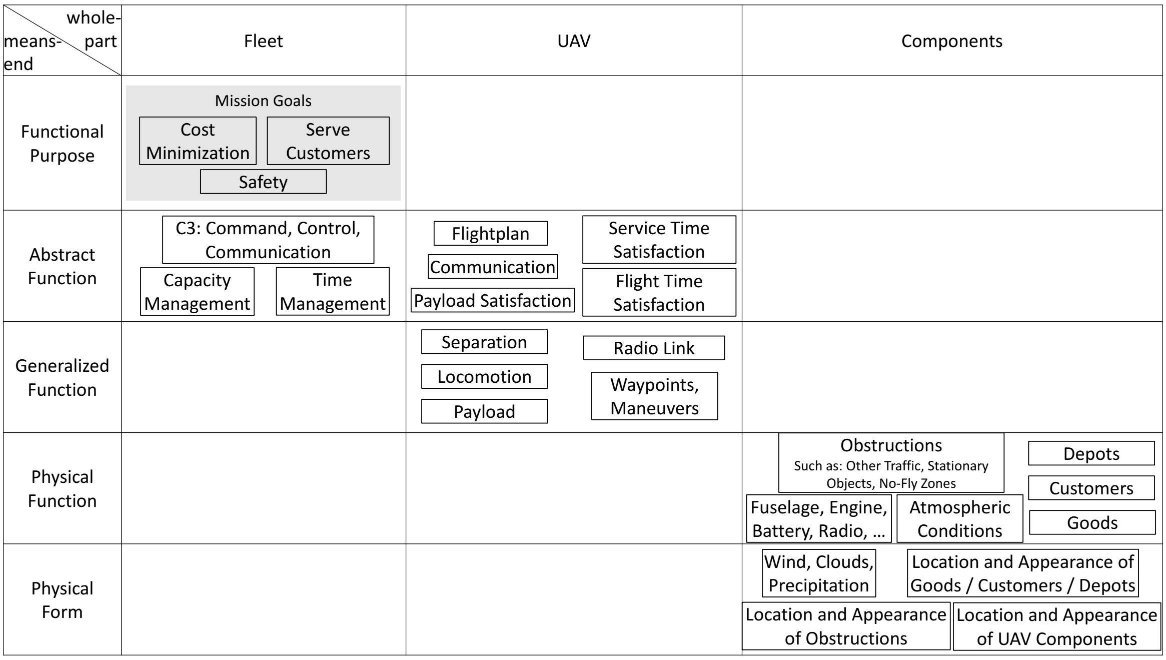


Figure B-2: Abstraction Decomposition Space (ADS) representing the structure of the multi-UAV work domain.

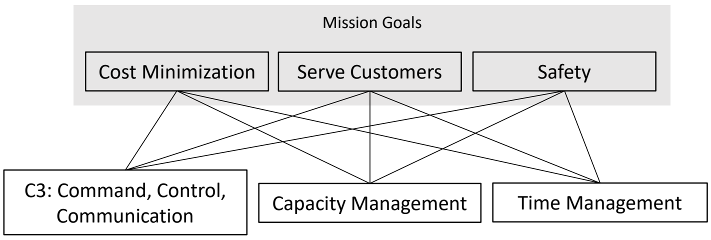


Figure B-3: Functional purpose and abstract function levels with means-ends links.

Abstract Function

The abstract function level, depicted with means-ends links in Figure B-4, represents the way in which the system purposes are achieved. The flightplan is achieved through locomotion and is dictated by waypoints and maneuvers, separation has an influence on the flightplan, since separation is required for safe operation. Payload satisfaction is dictated by the availability of payload to be transported and the waypoints that represent customer locations. Service time and flight time satisfaction is achieved through locomotion, waypoints and maneuvers and separation, which combined yield the UAVs flight path. Finally, communication is facilitated by the radio link.

The items discussed are parts of the items presented as abstract functions on the fleet level, as depicted in Figure B-5. C3 is achieved on the UAV level through communication and a flightplan. Capcity management consists of a flightplan and payload satisfaction. Finally, time management is achieved through service time and flight time satisfaction.

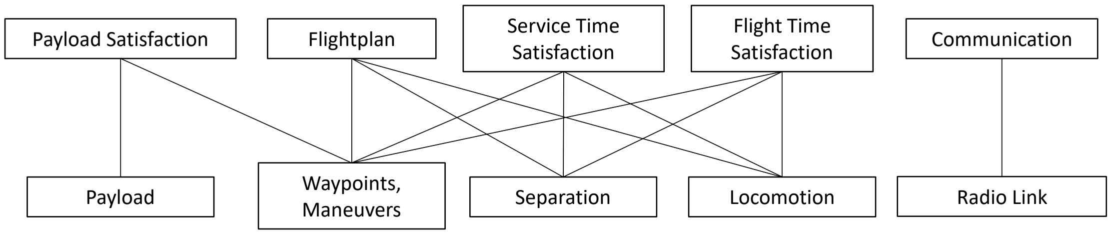


Figure B-4: Abstract function and generalized function levels with means-ends links.

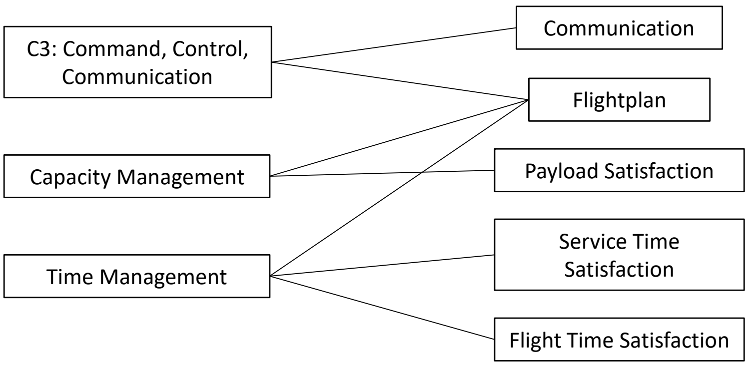


Figure B-5: Abstract function level at both fleet and UAV level with decomposition links.

Generalized Function

At the generalized function level, as represented with means-ends links in Figure B-6, functionality, independent of physical implementation are depicted. Obstructions dictate separation, which is also influenced by the atmospheric conditions. The radio link is facilitated by the radio component and signal propagation is influenced by atmospheric conditions. Locomotion, or the act of moving form place to place, is achieved through the UAV components and is limited by obstructions and atmospheric conditions. Waypoints and maneuvers are an abstraction of depot and customer locations and are influenced by the UAV components and atmospheric conditions. Finally, payload consists of the goods that need to be transported.

Physical Form and Physical Function

The physical form level, as depicted in detail in Figure B-7, is represented by descriptions of location and appearance at component level. Included are location and appearance of goods, customers and depots, terrain and obstacles, UAV components and wind. These items are a means to obtaining the functions represented at the physical function level, as depicted with structural means-ends links in Figure B-7 as well. The physical function level considers goods, customers and depots, obstructions (such as: traffic, stationary obstacles and no-fly zones), UAV components (such as: fuselage, engine, battery and radio), and atmospheric conditions,

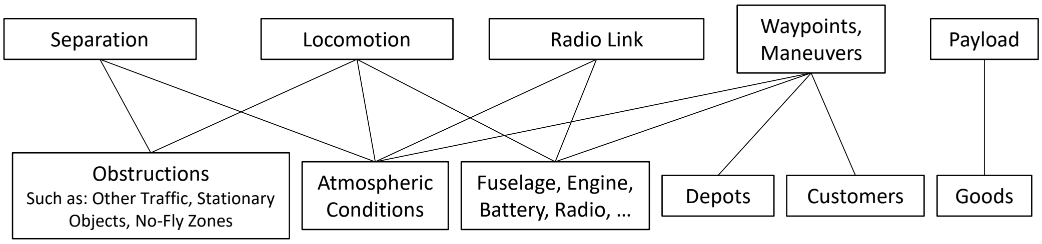


Figure B-6: Generalized function and physical function levels with means-ends links.

which are linked one-to-one with the physical form descriptions through the means-ends links.

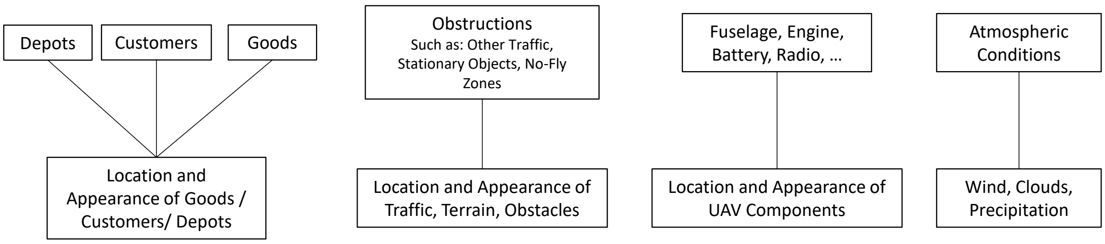


Figure B-7: Physical function and physical form levels with means-ends links.

B-3 Control Task Analysis

As previously discussed, the focus of this research is on supporting mission planning and disturbance and failure management. Hence, the control task analysis focuses on this phase of the payload delivery mission. The decision ladder of the aforementioned control task is depicted in Figure B-8.

Activation, on the bottom left of the decision ladder, represents the continuous process of monitoring mission status. After being alerted by off-nominal performance, observations regarding the depots, customers and vehicles in the fleet take place. These observations can be interpreted directly as a task. For example, when observing a flightplan passes a UAV through a no-fly zone, this can be directly perceived as the necessity for a change in waypoints to address this issue, leading to a new desired flightplan. Also, the observations can be interpreted directly as a procedure, for example when an unassigned customer location is desired to be included in the route of one of the vehicles. From the set of observations, the system state is identified, which relates to the flight plans, flight time, service time and payload capacity. Again, there is a shortcut, which represents interpretation as a task, for example when there is insufficient payload on board, which leads to the need to remove customer locations from the flight plan.

So far, the information processing activities correspond very well to skill- and rule-based behavior. The upper structure of the decision ladder represents more knowledge-based behavior, where a goal is explicitly formulated and a useful plan to achieve that goal is developed. This could, for example, correspond to situations in which there is insufficient energy available to achieve the planned mission. In such cases, decisions on reducing planned tasks and reassignment of those tasks need to be evaluated, taking into consideration the overall mission goals. Once these decisions are made, this leads to a target state, for which a task can be formulated, which in turn can be executed.

For the development of an interface that does not force cognitive control to a higher level than required, the identified shortcuts need to be carefully considered in the interface design. Especially experts often make use of these shortcuts. To also allow for creative problem solving, the top part of the decision ladder offers important cues. An interface should allow for exploring different options and evaluating associated consequences with respect to the overall mission goals.

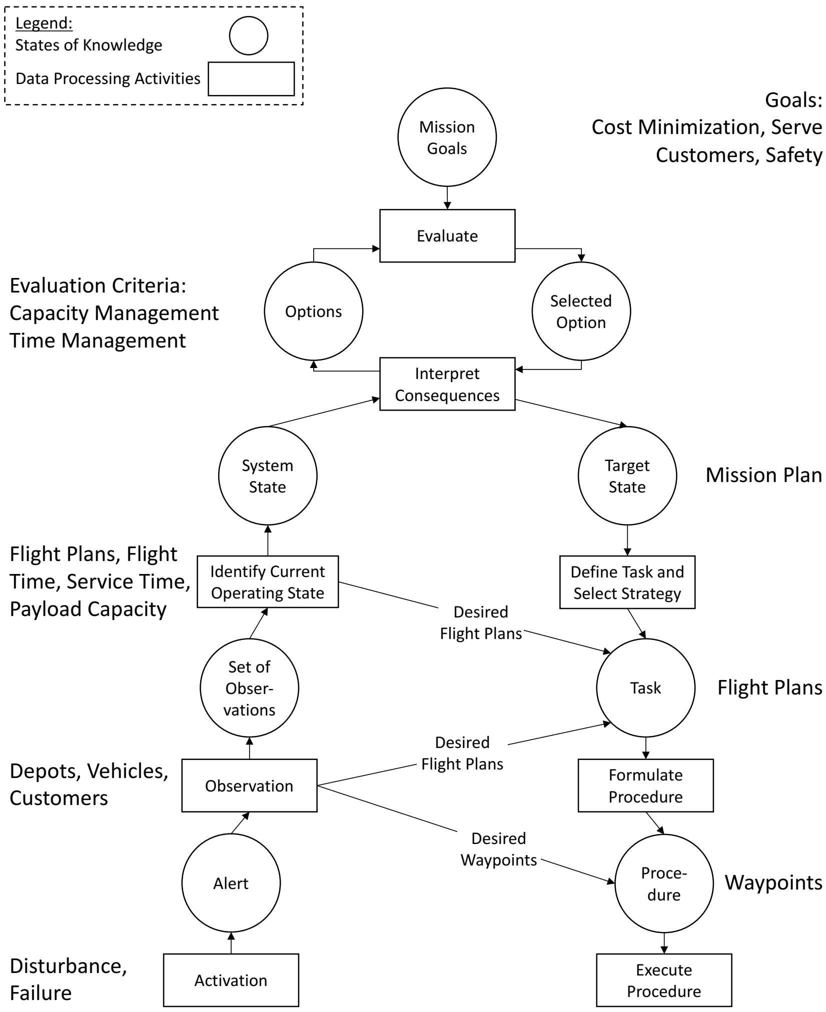


Figure B-8: Decision ladder for mission planning and disturbance and failure management.

B-4 Strategies Analysis

The core of the control problem is to transition from an infeasible mission plan, caused by disturbances and/or failures, to a feasible mission plan. Feasibility is dictated by the work domain constraints previously identified. Four different strategies to achieve the control objective are considered. Path stretching can be used to solve mission plan infeasibilities related to, for example, directing vehicles around areas such as no-fly zones. To facilitate solving mission plan infeasibilities caused by, for example, insufficient energy to complete the assigned tasks, or related to insufficient payload on-board, mission tasks can be reprioritized, reassigned, or removed. These strategies are not mutually exclusive, most likely a combination has to be used to solve a given mission plan infeasibility. A summarized representation of the different control strategies is depicted in Figure B-9.

To allow for creative problem solving, the to be developed ecological interface should support the identified strategies for solving mission plan infeasibilities. These infeasibilities can be the result of disturbances or failures introduced as deviations from the original mission plan during mission execution.

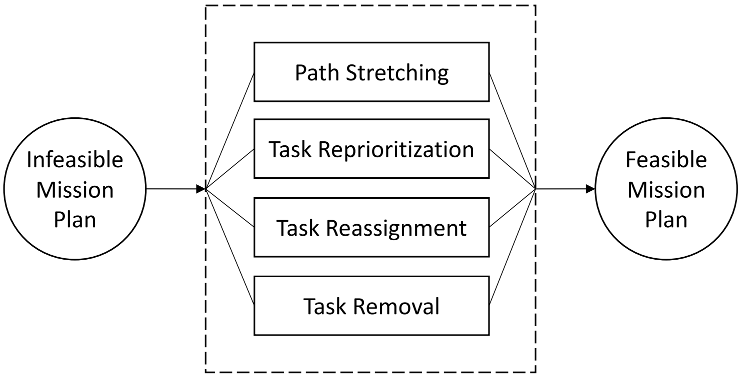


Figure B-9: Identified strategies for transition of infeasible mission plan to feasible mission plan.

B-5 Social Organization and Cooperation Analysis

After previous discussions on what tasks should be performed and how, it is now time to assess who should perform these tasks. In the present context of single-operator multi-UAV control, two actors can be identified: the human operator and automation. Hence, responsibility for the execution of the defined tasks has to be assigned to the operator or automation. This division of tasks is important, since if too many tasks are assigned to the human, this can have detrimental effects on workload and SA. Also, the quality of automation lies in the execution of repetitive, predictable tasks, whereas the human is better at finding solutions to unforeseen and unpredictable problems.

To aid in the division of tasks, the ADS that was previously presented is reproduced here, where tasks assigned to automation are grayed out. From the ADS, depicted in Figure B-10, it can be seen that the work domain of the human operator is focused on high level mission management through managing the UAVs in the fleet. That is not to say that the lower levels will not influence higher level processes, but rather that the human operator is not involved in managing individual UAV components. For example, each vehicle is assumed equipped with an on-board autopilot that steers the vehicle along the dictated guidance trajectory. These forms of automation will allow the human to focus on managing the mission and not become overloaded with having to attend low level vehicle functions.

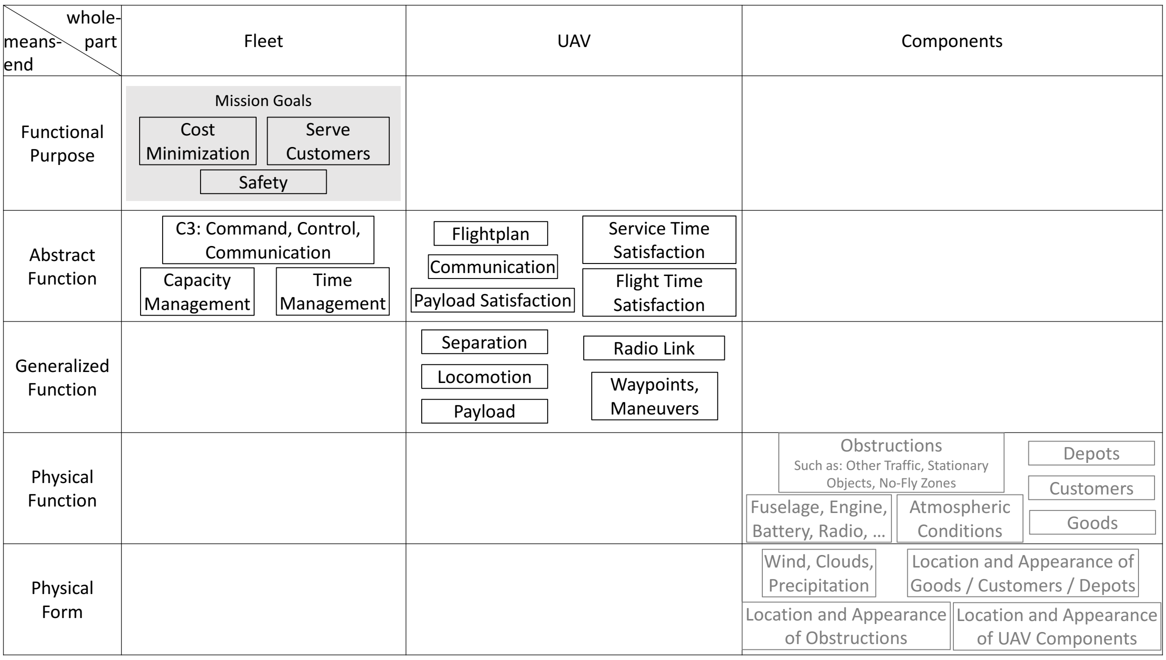


Figure B-10: Abstraction Decomposition Space (ADS) with human-automation task division.

B-6 Worker Competencies Analysis

In this section, worker competencies are assessed based on the previously discussed SRK-taxonomy. Figure B-11 provides an overview of information processing steps, resulting knowledge states, and corresponding skill-, rule-, and knowledge-based behavior for achieving single-operator multi-UAV mission management for a payload delivery mission. It is desired for the to be developed interface to support each level of the identified skill-, rule-, and knowledge-based behavior.

Information Processing Step	Resultant Knowledge State	Skill-Based Behavior	Rule-Based Behavior	Knowledge-Based Behavior
Observe future flightplan for each UAV	Whether any UAVs have infeasible flightplans	Monitor UAV flight plans	Perceive explicit indication that UAVs have infeasible flightplans	Reason, based on proposed flightplans, that UAVs may have infeasible flightplans
Predict time-based states for UAVs with infeasible flightplans	Whether UAVs will run out of energy, penetrate no-fly zones, have insufficient payload, etc.	Perceive energy levels, no-fly zones, payload levels, etc.	Use heuristics to estimate whether UAVs will run out of energy, penetrate no-fly zones, have insufficient payload, etc.	Calculate, based on current state and flightplan, if UAV has sufficient resources to complete flightplan
Determine the criticality of infeasible flightplans	Which UAV flightplan needs to be addressed in what order	Perceive which UAV is in most critical state	Use heuristics to estimate when UAV resources are depleted	Calculate, based on consumption of resources when UAV resources are depleted
Choose to modify UAV flightplan to address infeasibility	Which UAV flightplans need to be modified to yield mission feasibility	Directly perceive that one or more UAVs must be redirected	Apply doctrine: (e.g. If insufficient resources, MUST change UAV flightplan)	Reason from knowledge of proposed flightplans, current state, and expected future use of resources that flightplans must be altered
Select method for accomplishing feasible flightplan	Operator awareness of new flightplan	Respond automatically to insufficient resources by directly manipulating a representation of UAV flightplan(s)	Classify insufficient resources within a set of generalized scenarios and select appropriate stereotypical control rule	Develop new, optimized flightplan(s) based on weighted criteria including urgency, priority, efficiency, safety, etc.
Convey flightplan modifications to UAV for execution	UAV awareness of new flightplan	Direct, simultaneous interaction with communication equipment through control interface through input of updated flightplan information	Apply stereotypical control rules to select method/sequencing for conveying proposed flightplans	Reason using knowledge of UAV systems, priorities, urgency, etc., the best means and order for contacting each UAV to convey proposed flightplan(s)

Figure B-11: Worker competencies analysis, listing information processing steps, knowledge states, and corresponding skill-, rule-, and knowledge-based behavior, adapted from Kilgore and St-Cyr (2006).

B-7 Hierarchical Task Analysis

Where the previous sections have presented constraint-based analyses, this section will focus on the tasks that need to be performed to achieve the overall goal of the mission along with associated temporal relationships, by means of a Hierarchical Task Analysis (HTA). The purpose of the analysis is to provide knowledge that can be used as input to the interface design process relating to the support of tasks that need to be performed. Since this analysis is part of a design process, it is aimed for the analysis to be interface independent and, hence, restricted to a purely functional representation. Since the focus of the control task is put on operator mission management and disturbance and fault management, which are high level tasks, this analysis is restricted to high level goals, tasks and subtasks and the decomposition will not go into the detail of management of low level functions.

For representing the functional structure of the tasks, a diagrammatic format of the HTA is best suited and, hence, is used for the present analysis. The results of the analysis are depicted in Figure B-12. From the overall goal of the mission at the top of the diagram, tasks are formulated that need to be performed to successfully achieve this goal. For these tasks, several subtasks are identified that, together, accomplish the higher level tasks. Also, temporal information is included by means of the numbers between the brackets, that indicate if tasks are to be executed concurrently or consecutively. Especially the tasks at the bottom of the decomposition need to be carefully considered in the interface design, since these tend to yield the most information and are, hence, the most important (Annett, 2003).

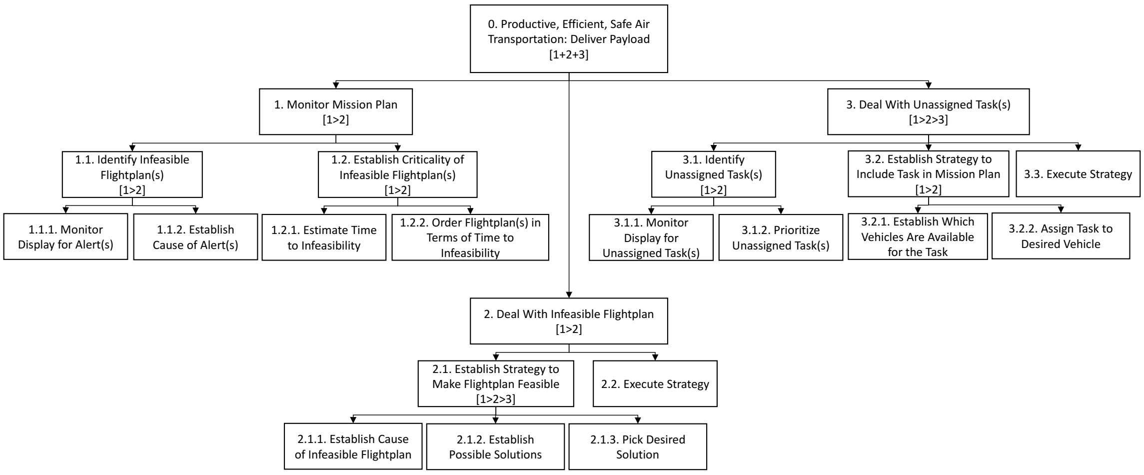


Figure B-12: Hierarchical task analysis for mission management and disturbance and failure management.

B-8 Conclusions

This chapter presented an analysis of work domain constraints, what tasks need to be performed, how those tasks can be performed, who should perform those tasks and in what order and what competencies are required to perform those tasks. Each step in the analysis offers cues for information requirements that will be considered in the design of an ecological interface for supporting mission management and disturbance and failure management in a single-operator multi-UAV payload delivery scenario.

The identified information requirements, although very useful in the analysis of the work to be performed, do not offer explicit cues on how this information should be visualized. This step is subject to further analysis and is the topic of the next chapter.

The content in this chapter has been graded as part of the preliminary thesis report under AE4020.

Appendix C

Preliminary Ecological Interface

C-1 Map View

The WDA presents locomotion, separation, radio link and waypoints and maneuvers at the generalized function level. Of these properties, locomotion, and associated constraints, have the biggest impact on effective control. Since, for managing disturbances and failures on UAV level, flightplans need to be effectively converted from being infeasible to feasible. To yield a feasible flightplan, waypoints need to be placed within locomotion constraints.

Locomotion is dependent on atmospheric conditions and battery status and performance. Since a detailed discussion on battery properties and associated electrical modeling is out of the scope of the current research, the concept of total flight time is introduced. Total flight time represents the flight time that is available to a single UAV on a full charge. As the State of Charge (SOC) of the battery decreases linearly with time, the flight time used increases proportionally. Equation C-1 defines the relationship between total, used, flightplan and available flight time and Equation C-2 defines the same relationship in terms of the battery SOC.

$$t_{total} = t_{used}(t) + t_{flightplan}(t) + t_{available} \tag{C-1}$$

$$t_{total} = (1 - SOC(t)) \cdot t_{total} + t_{flightplan}(t) + t_{available} \tag{C-2}$$

Consider the waypoint structure, as represented in Figure C-1, as the basis for a discussion on how geometric flightplan properties, battery SOC and atmospheric conditions relate to UAV locomotion. Three waypoints are depicted, where the path from waypoint 1 to waypoint 3 is the flightplan and waypoint 2 represents a probe point used for calculating an endurance envelope. In this scenario, Equation C-3 describes the relation between waypoint-to-waypoint flight time and flight time available. As the flight time is dictated by distance and ground speed, Equation C-4 represents the same relation with corresponding substitutions made.

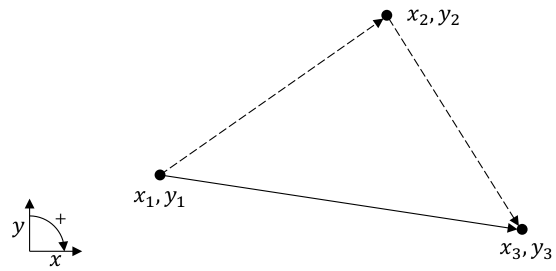


Figure C-1: Waypoint structure as used for endurance envelope derivation.

$$t_{1-2} + t_{2-3} - t_{1-3} < t_{available} \tag{C-3}$$

$$\frac{d_{1-2}}{V_{g_{1-2}}} + \frac{d_{2-3}}{V_{g_{2-3}}} - \frac{d_{1-3}}{V_{g_{1-3}}} < SOC(t) \cdot t_{total} - t_{flightplan} \tag{C-4}$$

Equation C-5 shows the relationship between the distance between two waypoints and the location of those waypoints.

$$d_{1-2} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{C-5}$$

Calculating V_g requires a more elaborate analysis. Starting point will be the relation between ground speed, airspeed and wind speed as described by Equation C-6.

$$\overrightarrow{V_g} = \overrightarrow{V_a} + \overrightarrow{V_w} \tag{C-6}$$

It is assumed that the wind is constant and uniform and as such it will not influence aircraft dynamics, but only increase or decrease the ground speed and hence, result in a longer or shorter flight time. A description of the relation between desired ground track, airspeed, wind speed, wind direction, and resulting ground speed is needed for substitution in Equation C-4. Figure C-2 depicts the geometric relationships between the three speed vectors.

From Figure C-2 and using the law of cosines, Equation C-7 is obtained:

$$V_g^2 = V_a^2 + V_w^2 - 2V_aV_w cos(\beta_g) \tag{C-7}$$

Also, from the figure, the following relation for β_g can be obtained:

$$\beta_g = \pi - \beta_w - \beta_a \tag{C-8}$$

Using the law of sines, a relation for β_w is obtained:

$$\frac{V_a}{sin(\beta_a)} = \frac{V_w}{sin(\beta_w)} \tag{C-9}$$

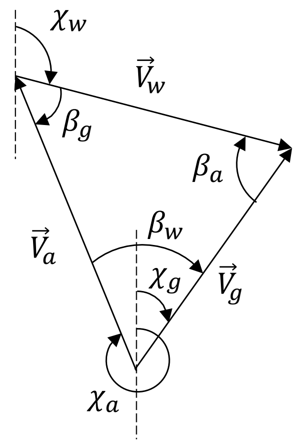


Figure C-2: Geometric relationships between ground speed, wind speed, and airspeed vectors.

$$\beta_w = arcsin\left(\frac{V_w}{V_a}sin(\beta_a)\right) \tag{C-10}$$

Substituting in Equation C-8, yields:

$$\beta_g = \pi - arcsin\left(\frac{V_w}{V_a}sin(\beta_a)\right) - \beta_a \tag{C-11}$$

Expressing β_a in terms of χ_w and χ_g, yields:

$$\beta_a = \chi_w - \chi_g \tag{C-12}$$

Substituting in Equation C-7, yields:

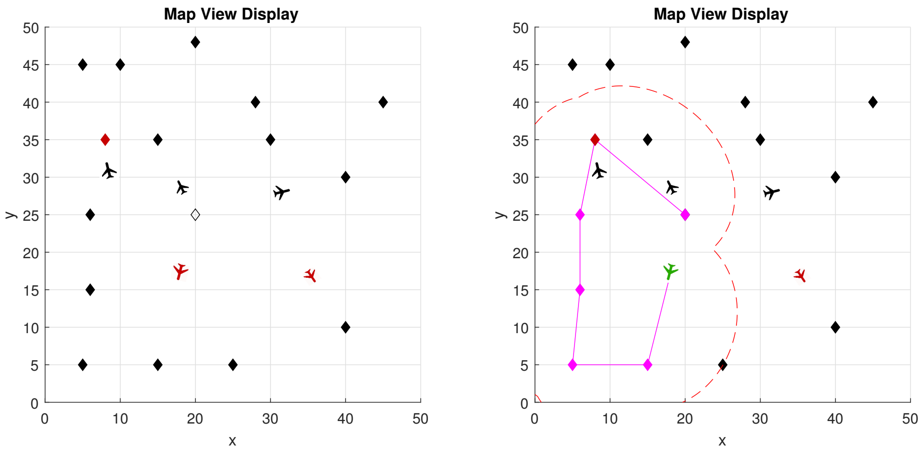
$$V_g^2 = V_a^2 + V_w^2 - 2V_aV_wcos(\pi - arcsin\left(\frac{V_w}{V_a}sin(\chi_w - \chi_g)\right) - (\chi_w - \chi_g)) \tag{C-13}$$

Where χ_g can be expressed for two waypoints as:

$$\chi_{g_{1-2}} = \frac{\pi}{2} - arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{C-14}$$

Substituting Equation C-5 and Equation C-13 in Equation C-4 and leaving x_2 and y_2 as variables yields a relation that can be used to obtain the endurance envelope around a current flightplan. That is, it represents where the UAV can locomote while adhering to battery constraints and subject to wind. For a simple two-waypoint scenario with varying wind conditions, some endurance envelopes are depicted in Figure C-4.

The previous discussion on endurance envelopes is very relevant for vehicle level control, since it describes UAV locomotion subject to wind and battery constraints. These endurance envelopes can be effectively mapped onto a map view, as depicted in Figure C-3. The display shows all UAVs in the fleet as well as all the waypoints and the home location and uses color to direct operator attention to infeasibilities. After selecting a UAV, extra information related to the selected vehicle is displayed, namely its waypoints that make up the flightplan and the associated endurance envelope. In this mode, the flightplan can be modified by creating, moving, or removing waypoints. The endurance envelope supports the operator in the creation of feasible UAV flightplans.



(a) Map view display visualizing all UAVs (airplane symbols), waypoints (filled diamonds) and the home location (diamond).

(b) Map view display showing, after selecting a UAV, the associated flightplan and endurance envelope.

Figure C-3: Map view display for UAV level control (zero wind).

C-2 Mission View

The WDA presents capacity management, time management and C3 at the fleet level and flightplan, communication, payload satisfaction, service time satisfaction and flight time satisfaction at the UAV level. The operator is tasked with managing the mission plan, which is the combination of UAV flightplans, and ensuring the mission plan is feasible, which is achieved through satisfying the payload, service time and flight time constraints. Hence, visualization of the mission plan, along with energy and payload constraints is needed for effective fleet level control.

A mission view, as depicted in Figure C-5, is proposed. By plotting the available flight time versus the remaining flight time in the current flightplan for all UAVs, an overview of mission plan feasibility and mission progress is achieved. The gray area in the display is where UAVs with infeasible flightplans show up, in this area, the flight time available is negative and hence, return to the home location is impossible. When selecting a UAV, the associated flightplan is indicated, as well as any unassigned waypoints that lie within reach. Each waypoint marker of the active flightplan includes an indicator showing the amount of energy, or flight time, that can be obtained by removing the waypoint. This cue is expected to be useful in managing infeasible flightplans as the map view display will not offer any specific support towards turning an infeasible flightplan into a feasible one. Also, the payload level for the selected

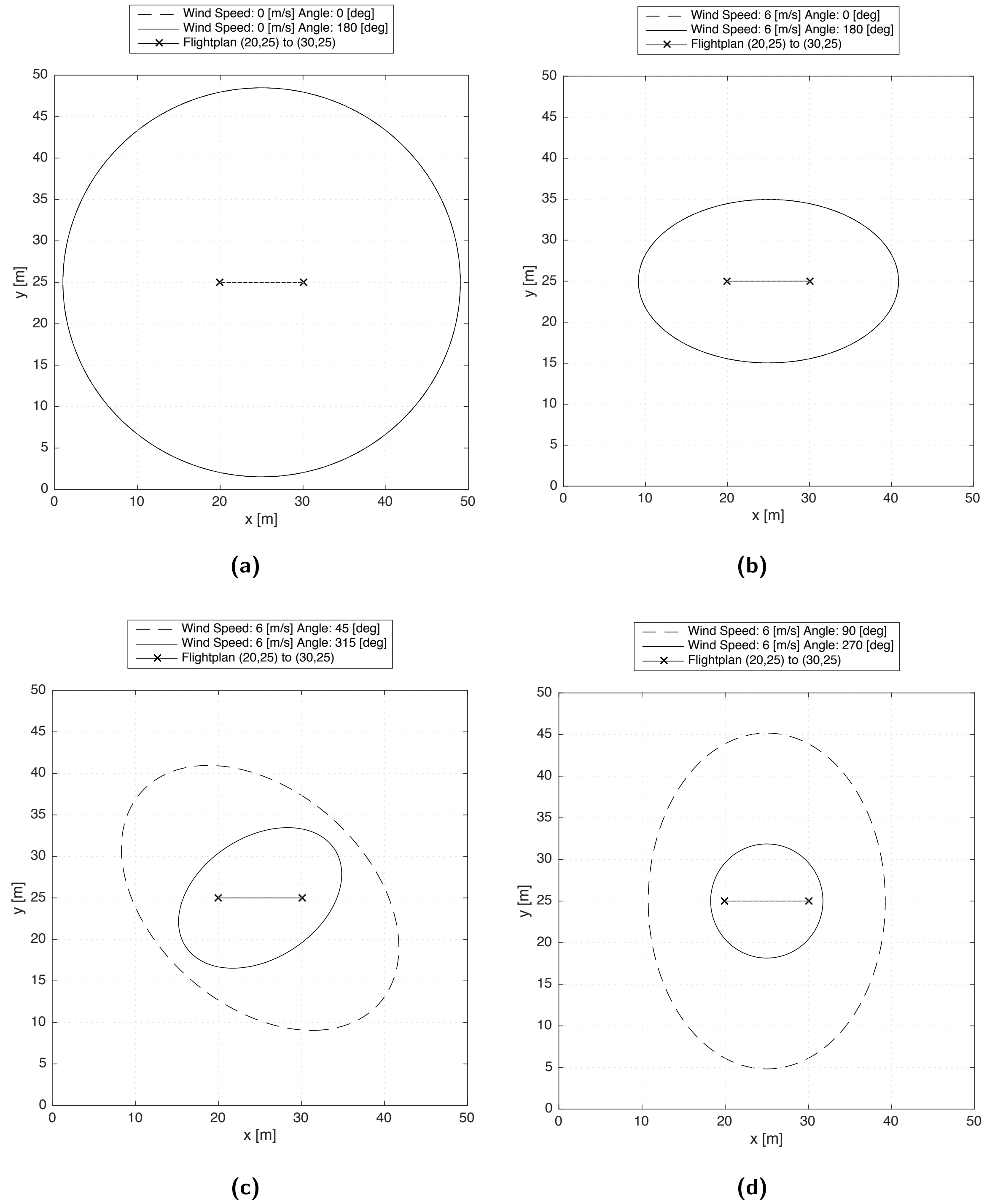
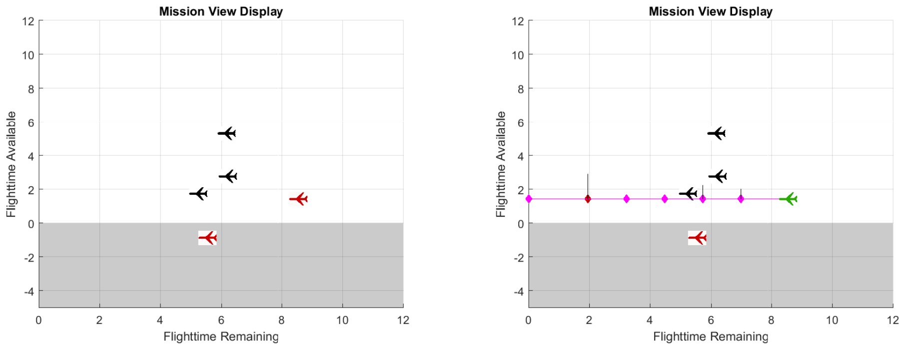


Figure C-4: Endurance envelopes for varying wind conditions (airspeed: 8 [m/s]).

UAV is taken into account by marking any waypoints in the flightplan for which insufficient payload is available. Color is used to direct operator attention to flightplan infeasibilities. Both waypoints and UAV symbols are colored red to indicate infeasibilities, green for selected items and magenta is used to indicate the current flightplan.

The map view and mission view provide similar cues to the operator, since both mark UAVs and waypoints that are infeasible. Both present a high level mission overview as well as a lower level UAV flightplan representation. And, both show work domain affordances, where the map view displays them on the generalized function level and the mission view on abstract function level, respectively.



(a) Mission view display visualizing all UAVs, flightplan feasibility, energy state and overall mission completion.

(b) Mission view display showing, after selecting a UAV, the associated flightplan and alternative flightplan options with payload requirements.

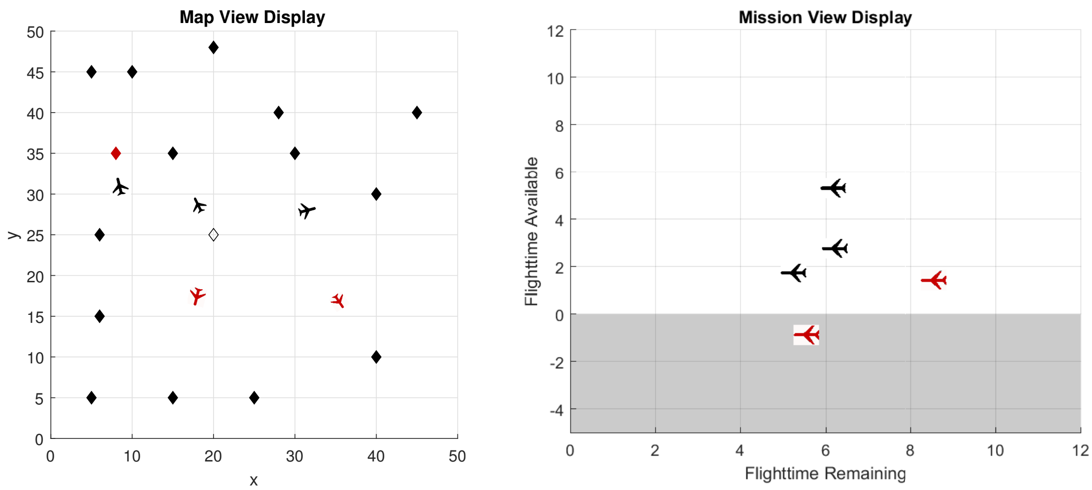
Figure C-5: Mission view display for fleet level control.

C-3 Example Case

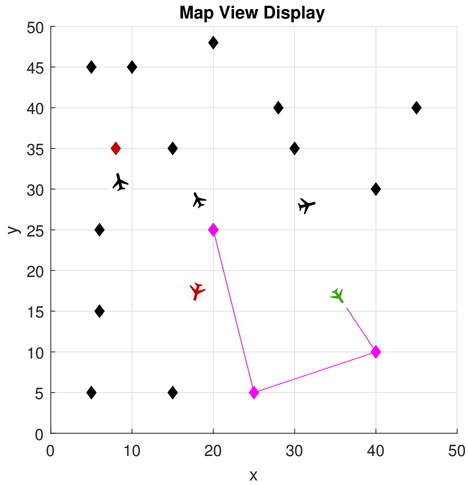
Now that the interface concepts have been defined, Figure C-6 shows an example multi-UAV payload delivery mission case to illustrate how the displays can be used in a representative scenario. The scenario consists of five UAVs that each have a unique flightplan associated with them. Figure C-6a and Figure C-6b display the start of the mission when all waypoints, which correspond to payload drop-off locations, are assigned to one of the flightplans. One of the UAVs has an infeasible flightplan caused by insufficient energy and another has an infeasible flightplan due to insufficient payload, both are indicated by a red vehicle icon and a red waypoint icon.

The role of the human operator is to manage the mission and address any infeasibilities and at the same time ensure overall mission success, which is defined as delivering payload to all delivery locations. To investigate the first infeasible flightplan, the corresponding UAV can be selected, the result of which is visualized in Figure C-6c and Figure C-6d. From the visualizations it can be observed that it is possible to remove a waypoint to make the flightplan feasible again, as displayed in Figure C-6e and Figure C-6f. Deselecting the UAV results in the mission overview as depicted in Figure C-6g and Figure C-6h. Since a waypoint was removed from a flightplan, not all waypoints are included in the mission plan and hence, the overall mission is not successful. The waypoint that was removed from the flightplan and is not visited by any UAV is indicated in red. The energy display shows the available options for including the unassigned waypoint into one of the UAV flightplans. Important note is that only options that satisfy current constraints are indicated, different solutions are of course possible but would require shifting around waypoints between UAVs. Hence, the goal of the display is not to indicate the most optimal solution, but to offer the operator an overview of currently available options. Selecting the available option in the energy display results in the selection of the associated UAV, as is illustrated by Figure C-6i and Figure C-6j. Assigning the waypoint to the flightplan results in an updated flightplan as depicted by Figure C-6k and Figure C-6l. Finally, deselecting the active UAV yields the mission overview as illustrated by Figure C-6m and Figure C-6n.

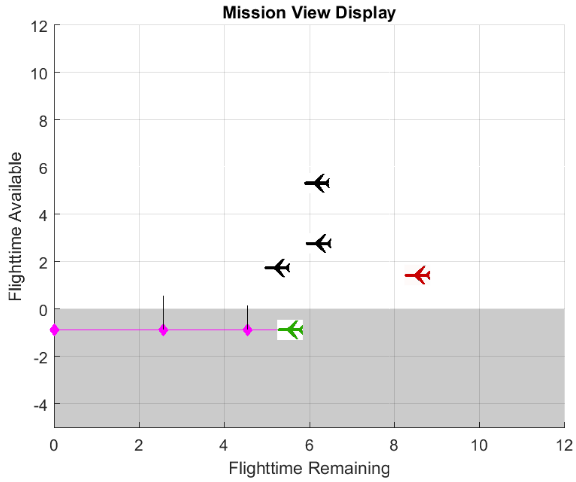
The second infeasibility can now be dealt with. Selecting the UAV as indicated by Figure C-6o and Figure C-6p shows the flightplan is infeasible because of insufficient payload at the last waypoint before the home location, as indicated by the red waypoint icon. Although only the last waypoint is colored red, any waypoint in the flightplan can be removed to solve this infeasiblity. Again, the goal is not to display the best waypoint to be removed but to indicate the payload constraints. Clearly, the last waypoint violates the payload constraints since that is where insufficient payload would be available. If the indicated waypoint is removed from the flightplan, that results in a feasible flightplan for the associated UAV as illustrated by Figure C-6q and Figure C-6r. Deselecting the vehicle results in the displays as depicted by Figure C-6s and Figure C-6t. Although the vehicle is not colored red any more, the waypoint still is. By removing the waypoint from the flightplan, this location is now not visited by any UAV under the current mission plan. Hence, options for assigning this waypoint are indicated in the displays. Selecting the option that requires the least amount of energy selects the associated UAV as illustrated by Figure C-6u and Figure C-6v. As can be seen from the mission view display, the other available option is still indicated, which means the waypoint could also be included at a later point in the flightplan, although this will require more energy. Assigning the waypoint to the flightplan yields the displays as depicted by Figure C-6w and Figure C-6x. Deselecting the UAV shows the current mission overview as illustrated by Figure C-6y and Figure C-6z. As can be seen from the displays, all flightplans are feasible and all waypoints are assigned, yielding a successful mission.



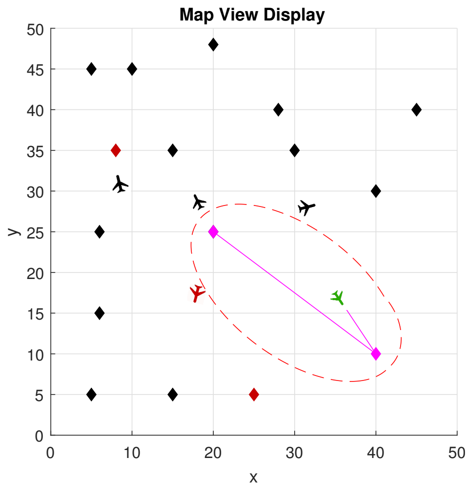
(a) **(b)**


Map View Display
x
y

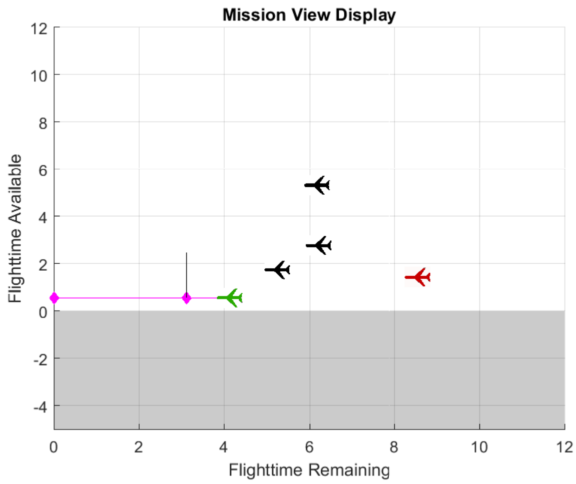
(c)


Mission View Display
Flighttime Remaining
Flighttime Available

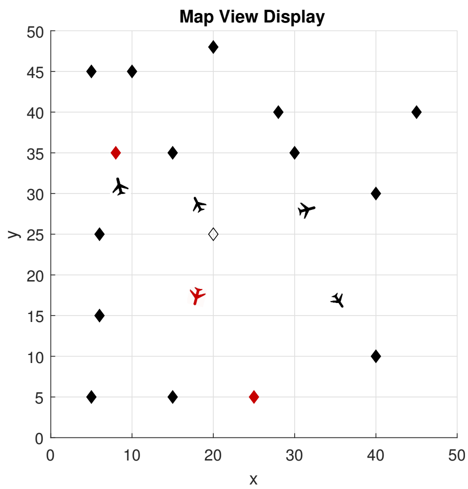
(d)


Map View Display
x
y

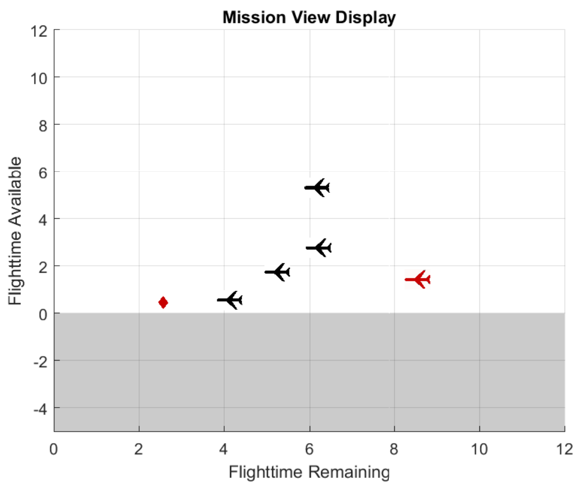
(e)


Mission View Display
Flighttime Remaining
Flighttime Available

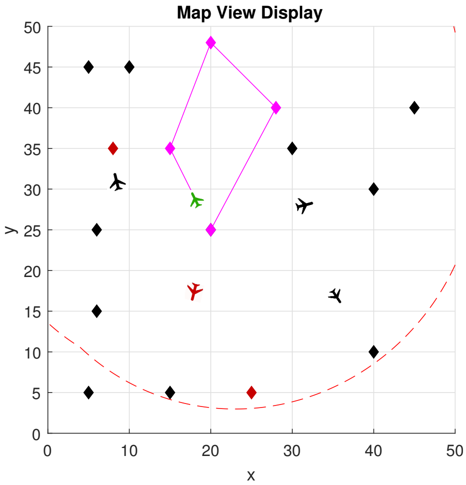
(f)


Map View Display
x
y

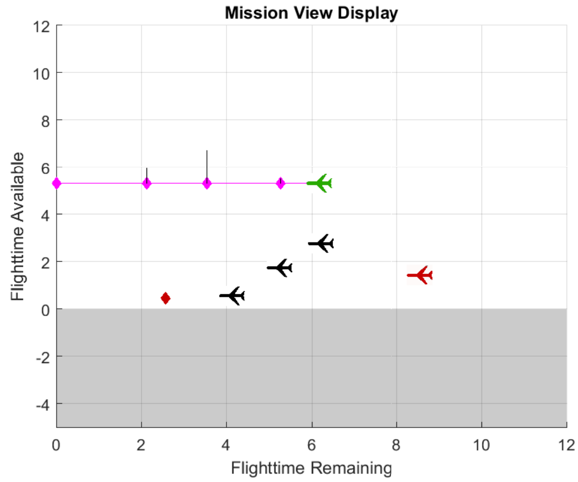
(g)


Mission View Display
Flighttime Remaining
Flighttime Available

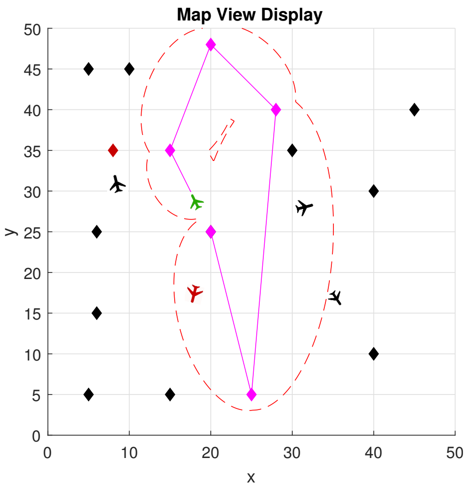
(h)


Map View Display
x
y

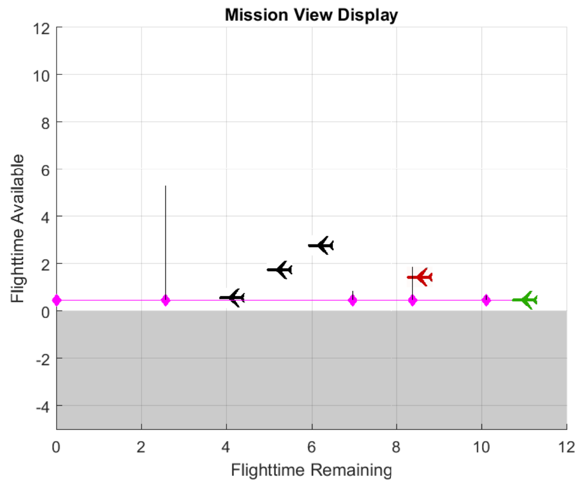
(i)


Mission View Display
Flighttime Available
Flighttime Remaining

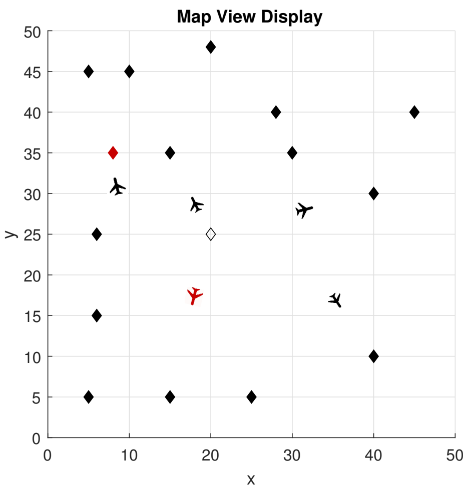
(j)


Map View Display
x
y

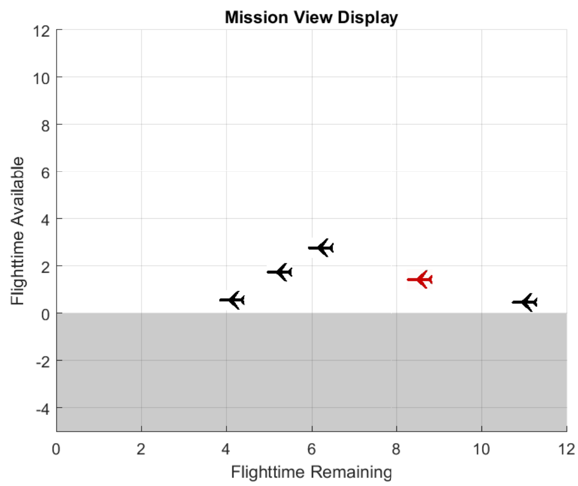
(k)


Mission View Display
Flighttime Available
Flighttime Remaining

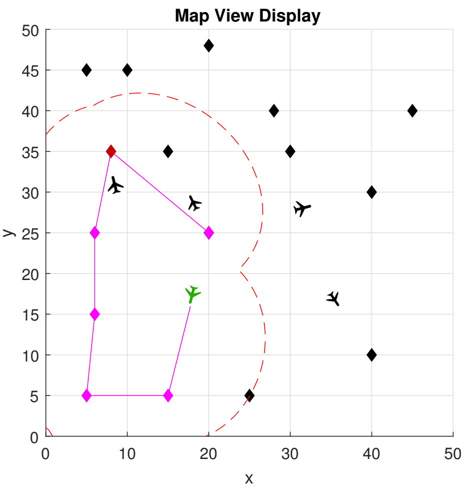
(l)


Map View Display
x
y

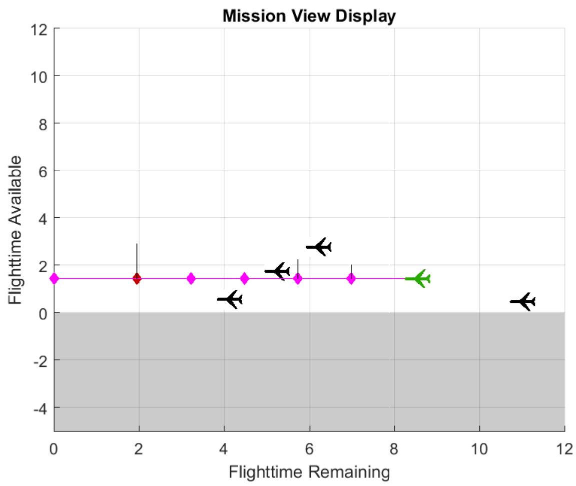
(m)


Mission View Display
Flighttime Available
Flighttime Remaining

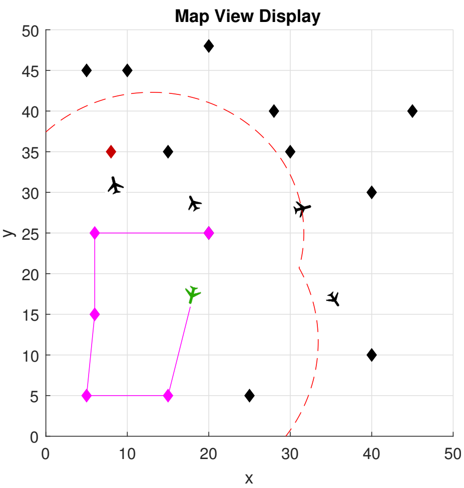
(n)


Map View Display
y
x

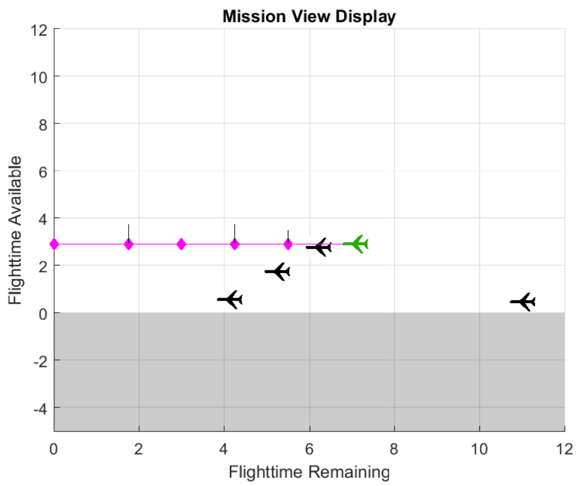
(o)


Mission View Display
Flighttime Available
Flighttime Remaining

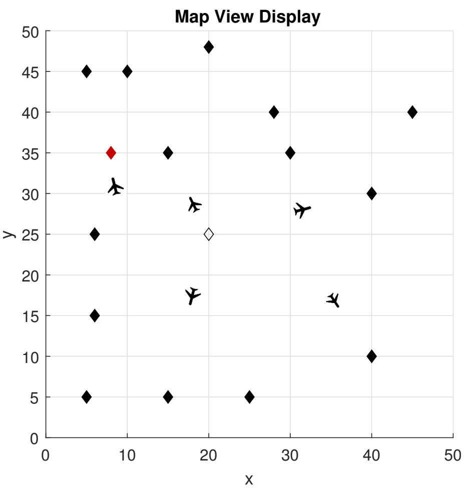
(p)


Map View Display
y
x

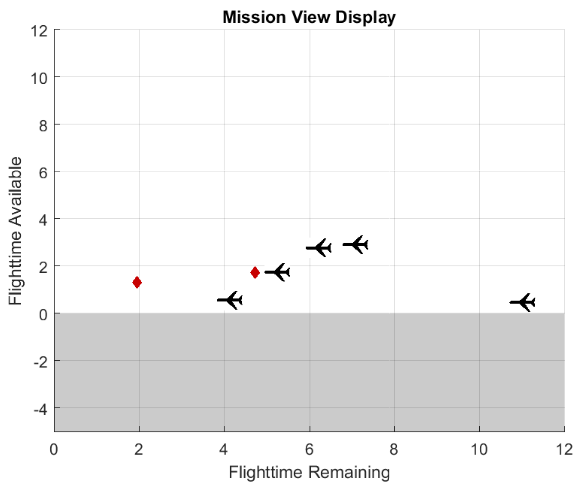
(q)


Mission View Display
Flighttime Available
Flighttime Remaining

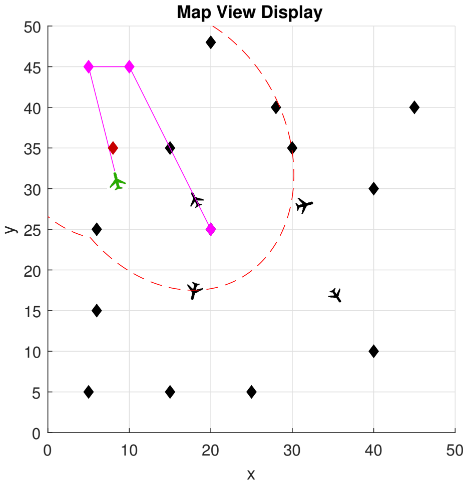
(r)


Map View Display
y
x

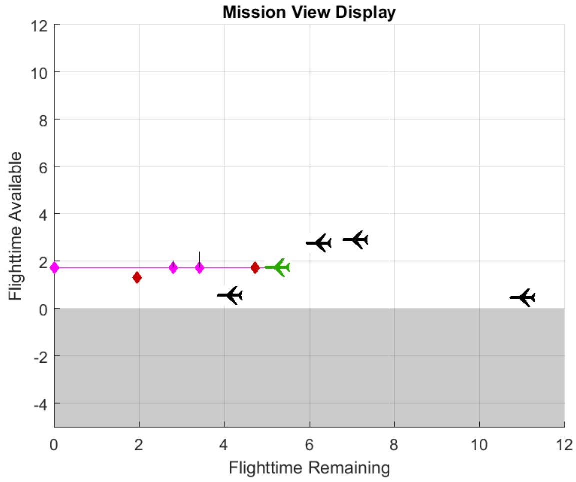
(s)


Mission View Display
Flighttime Available
Flighttime Remaining

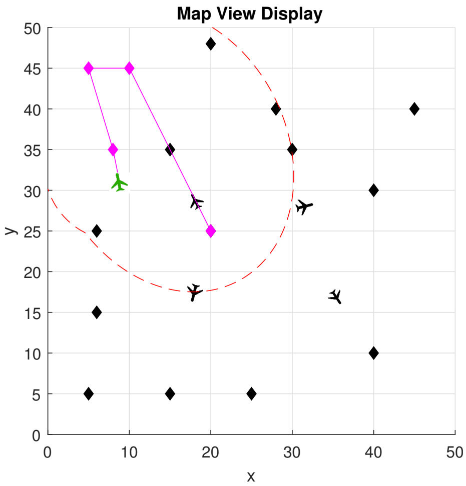
(t)


Map View Display
y
x
0
5
10
15
20
25
30
35
40
45
50
0
10
20
30
40
50

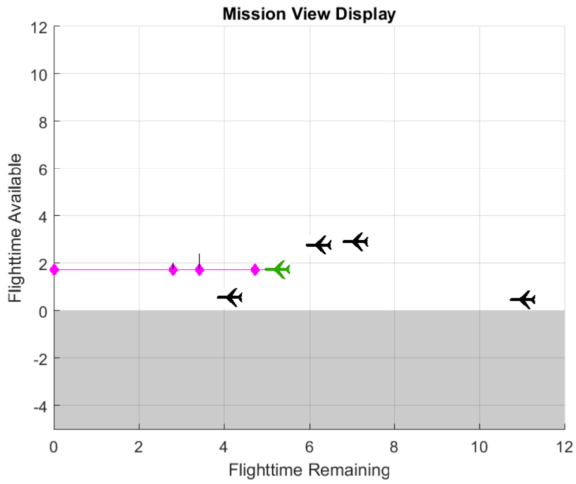
(u)


Mission View Display
Flighttime Available
Flighttime Remaining
12
10
8
6
4
2
0
-2
-4
0
2
4
6
8
10
12

(v)


Map View Display
y
x
0
5
10
15
20
25
30
35
40
45
50
0
10
20
30
40
50

(w)


Mission View Display
Flighttime Available
Flighttime Remaining
12
10
8
6
4
2
0
-2
-4
0
2
4
6
8
10
12

(x)

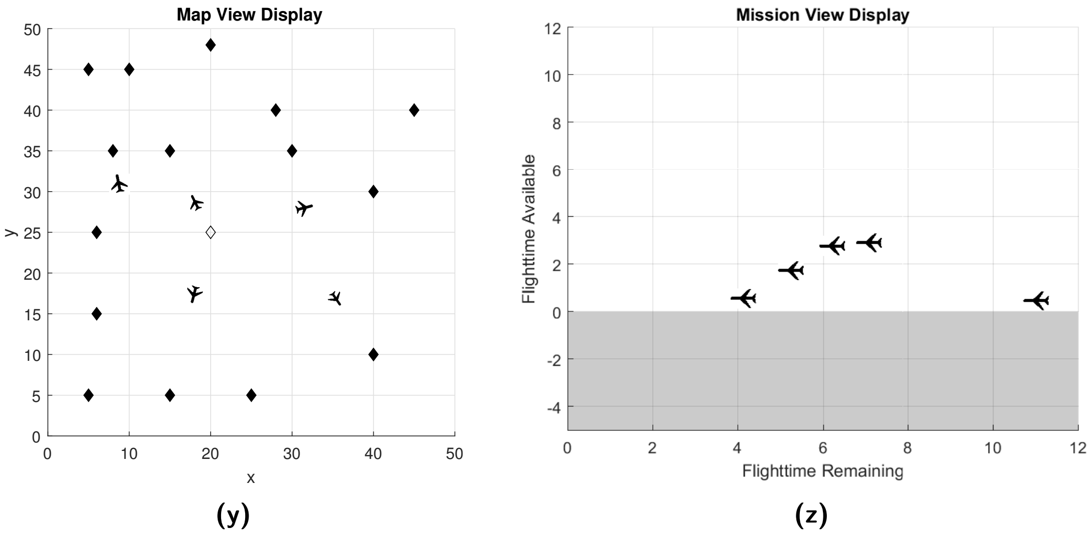


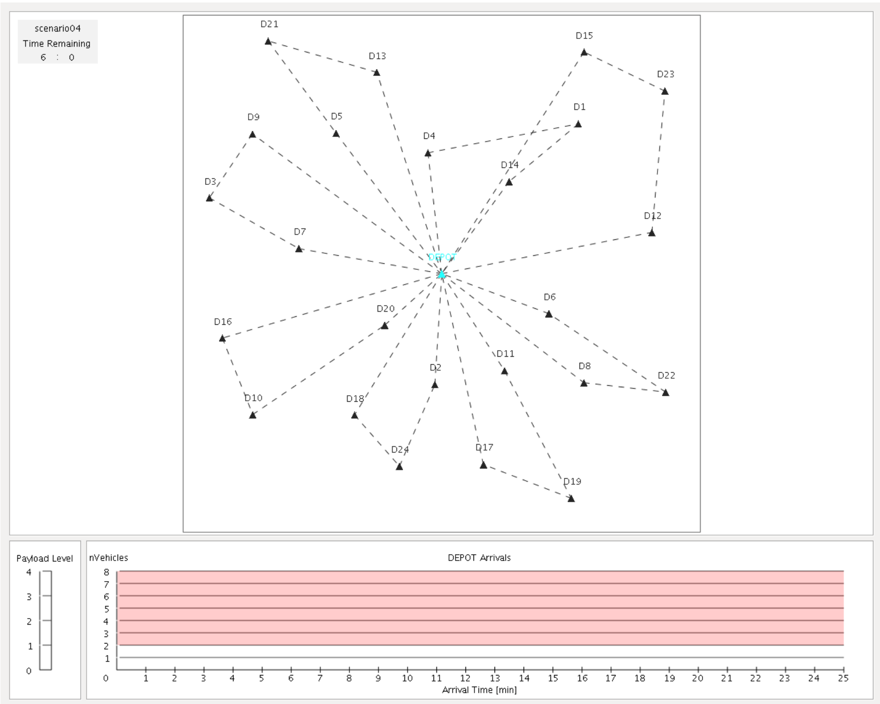
Figure C-6: Map display and energy display prototypes corresponding to an example multi-UAV payload delivery mission case.

Appendix D

Final Ecological Interface

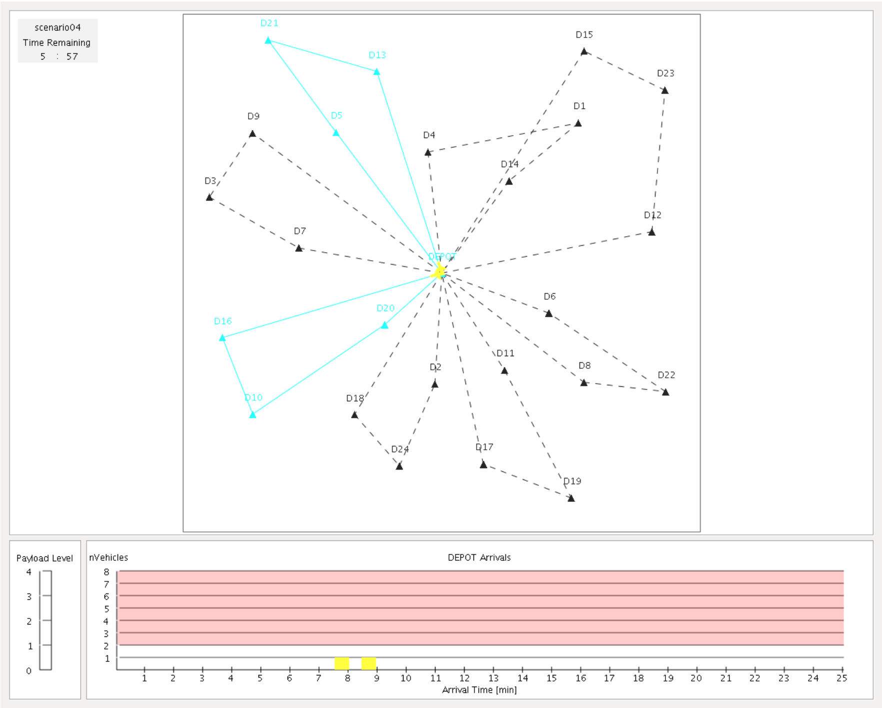
D-1 Map, Payload and Timeline Views

This section presents a number of illustrative screenshots of the final ecological interface for a representative scenario.

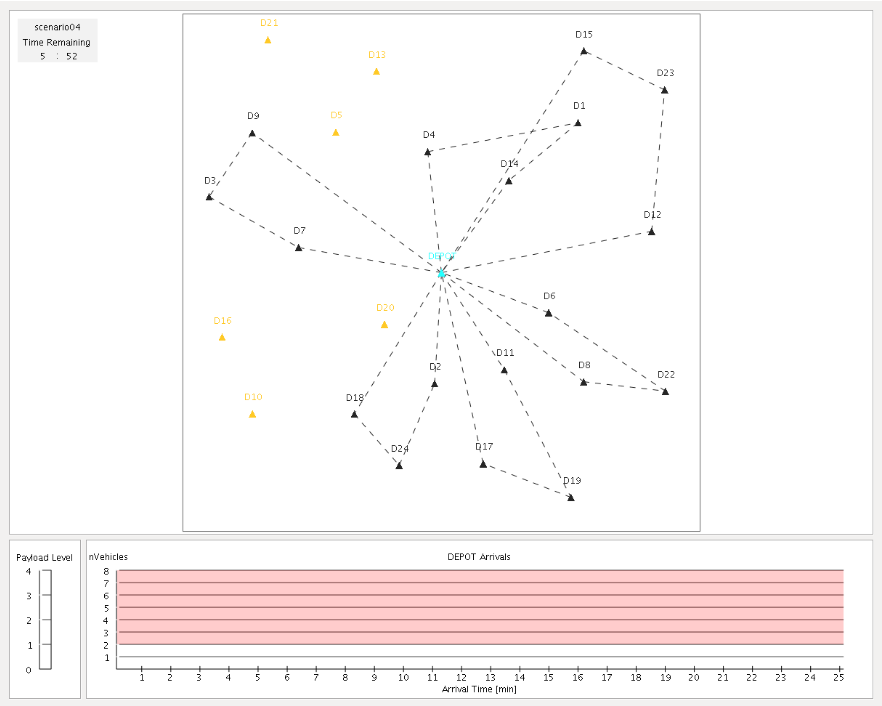


(a) final-display-d1

Figure D-1: Final Interface

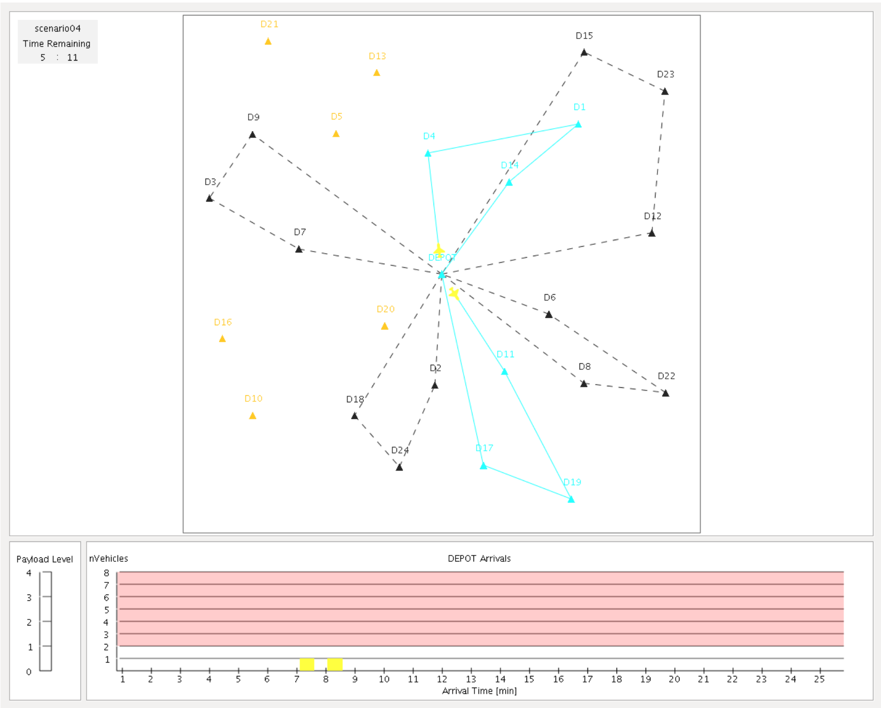


(a) final-display-d2

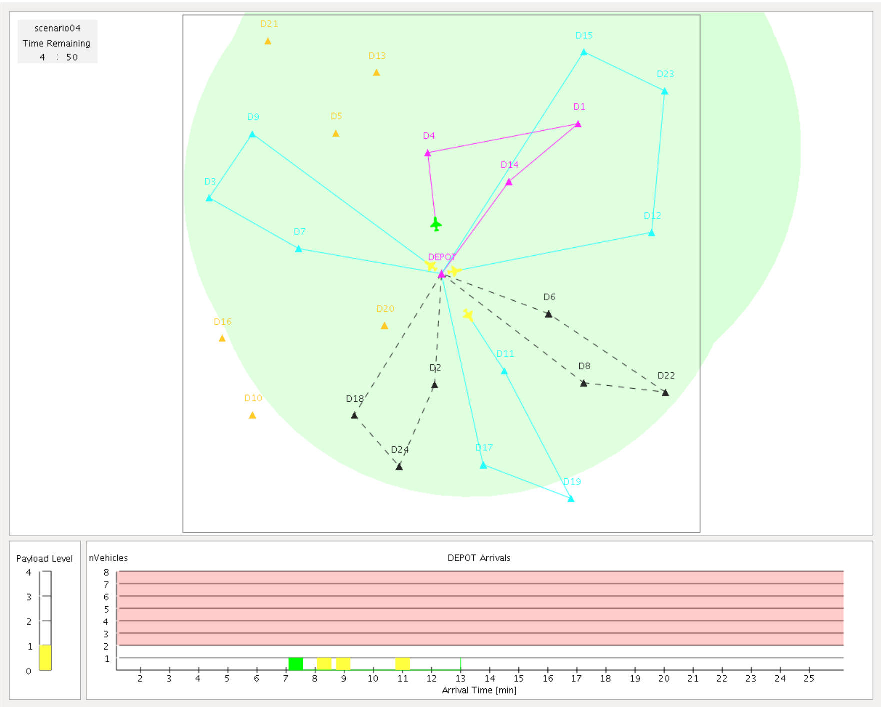


(b) final-display-d3

Figure D-2: Final Interface (continued)

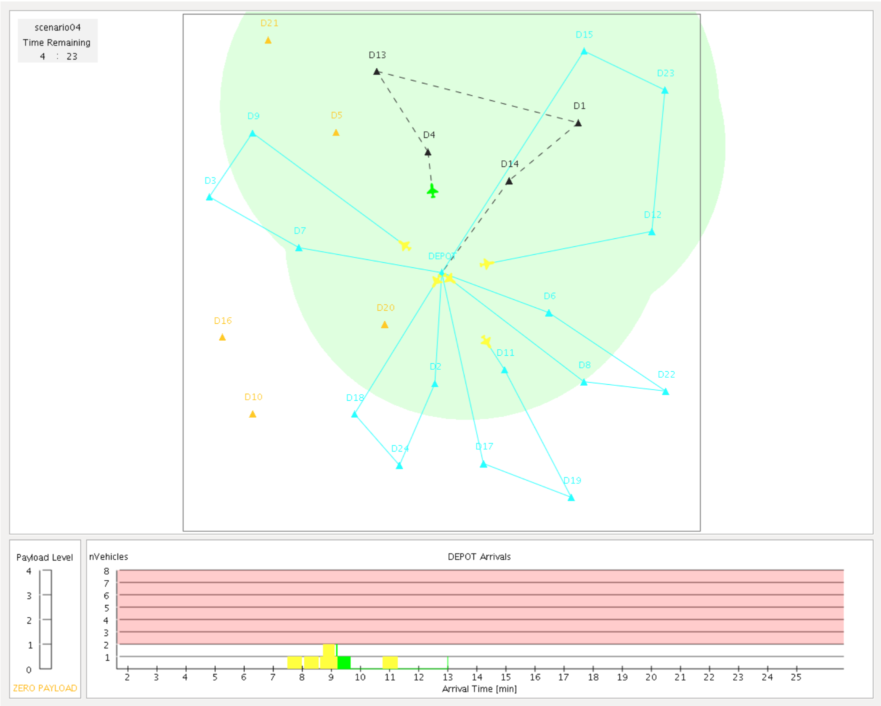


(c) final-display-d4

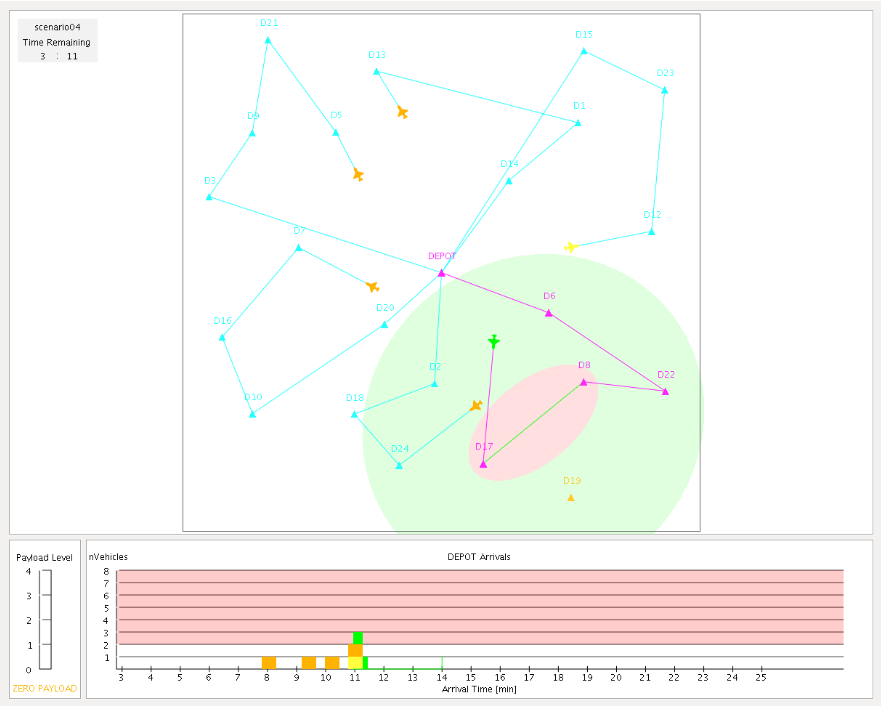


(d) final-display-d5

Figure D-2: Final Interface (continued)

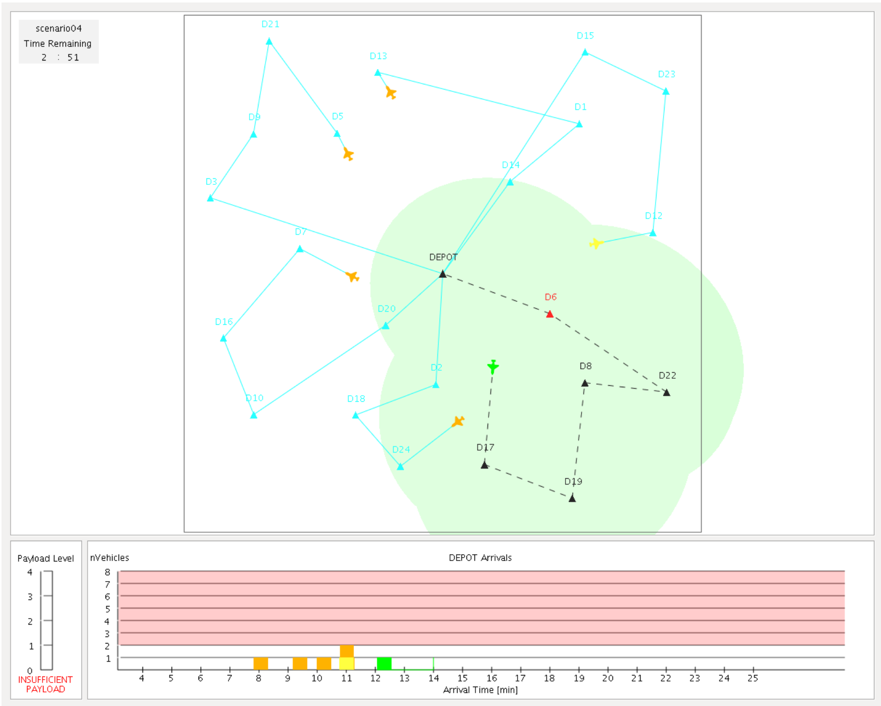


(e) final-display-d6

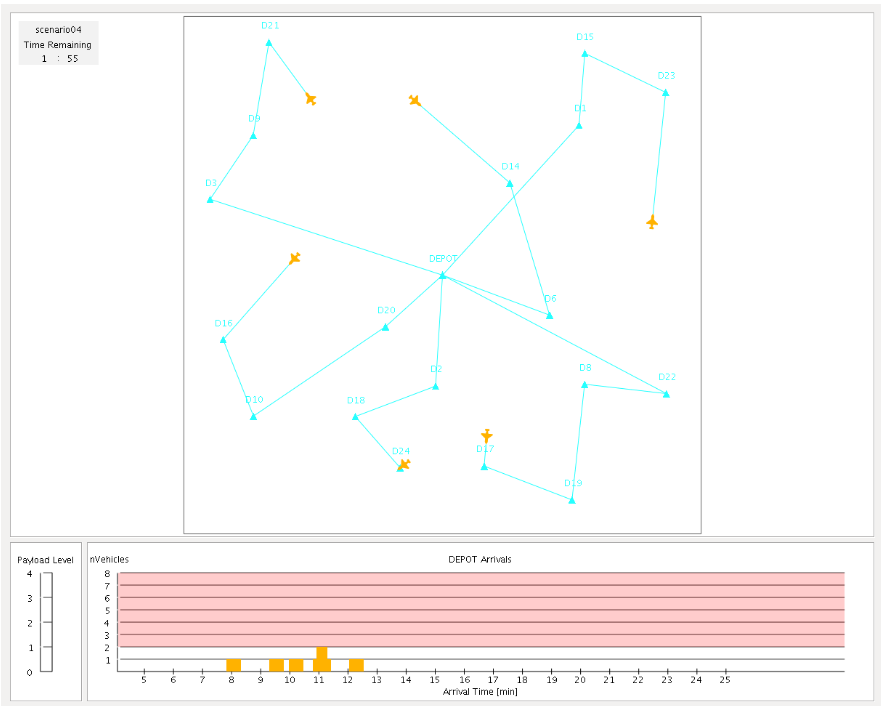


(f) final-display-d7

Figure D-2: Final Interface (continued)



(g) final-display-d8



(h) final-display-d9

Figure D-2: Final Interface (continued)

D-2 Flight Time Constraint Visualization

The flight time constraint is visualized as an ellipse around the guidance reference. For zero wind conditions, this ellipse can be easily calculated from geometry and available flight time, as briefly demonstrated below. From the parametric representation of an ellipse, with the semi-major axis a, the semi-minor axis b, linear eccentricity c and center (c_1, c_2) as defined in Figure D-3:

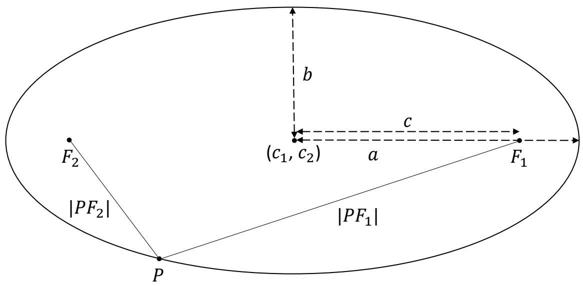


Figure D-3: Ellipse definition

$$(c_1 + a \cdot cos(\theta), c_2 + b \cdot sin(\theta)), 0 \leq \theta < 2\pi \tag{D-1}$$

where,

$$b^2 = a^2 - c^2 \tag{D-2}$$

$$|PF_1| + |PF_2| = 2a \tag{D-3}$$

By letting c be half the distance between waypoints F_1 and F_2 and letting a be half the sum of the flight time from F_1 to F_2 and any remaining available flight time, the flight time constraint envelope can be drawn. It is first rotated to the proper orientation using:

$$\begin{aligned} x' &= x \cdot cos(\theta) - y \cdot sin(\theta) \\ y' &= x \cdot sin(\theta) + y \cdot cos(\theta) \end{aligned} \tag{D-4}$$

Then it is moved to its proper location by shifting it to (c_1, c_2).

Appendix E

Experiment Design

E-1 Experiment Conditions

The experiment conditions are summarized in Table E-1. The number of vehicles and number of customers in the scenarios is determined by the number of failures and number of payload for each vehicle, which are the independent variables. The number of vehicles and customers is calculated by using Equation E-1 and Equation E-2, respectively. Also, Table E-2 lists the resulting values for each experimental condition. The control variables are listed in Table E-3.

Table E-4 is a tabulated form of Equation E-2, where the number of failures are indicated as a function of numbers of customers (vertically) and numbers of payload (horizontally). The marked cells are the experiment conditions. These conditions have been chosen in such a way that for each condition the payload capacity of the fleet ($nPayload \cdot nVehicles$) is exactly equal to the number of customers.

$$nVehicles = nFailurs \cdot nPayload \tag{E-1}$$

$$nCustomers = nPayload \cdot nFailurs \cdot (nPayload - 1) \tag{E-2}$$

Table E-1: Experiment Conditions

	Payload 4	Payload 5	Payload 6	Payload 7
Single UAV Failure	F1P4	F1P5	F1P6	F1P7
Double UAV Failure	F2P4	F2P5	F2P6	F2P7

[1]Rounded to nearest integer.

Table E-2: Customers and Vehicles per Condition

	nCustomers	nVehicles
F1P4	12	4
F1P5	20	5
F1P6	30	6
F1P7	42	7
F2P4	24	8
F2P5	40	10
F2P6	60	12
F2P7	84	14

Table E-3: Control Variables

Variable	Value
Max Flight Time (s)	750
Airspeed (m/s)	13
Service Time (s)	30
Scenario Duration (s)	360
Failure Times (s)	5
Payload Margin (-)	1
Sector Size (m^2)	5000 x 5000
Depot Capacity (-)	30% of nVehicles [1]

Table E-4: Number of failures for various numbers of customers and numbers of payload.

	2	3	4	5	6	7	8	9	1e+01
1e+01	5.000	1.667	0.833	0.500	0.333	0.238	0.179	0.139	0.111
1e+01	5.500	1.833	0.917	0.550	0.367	0.262	0.196	0.153	0.122
1e+01	6.000	2.000	1.000	0.600	0.400	0.286	0.214	0.167	0.133
1e+01	6.500	2.167	1.083	0.650	0.433	0.310	0.232	0.181	0.144
1e+01	7.000	2.333	1.167	0.700	0.467	0.333	0.250	0.194	0.156
2e+01	7.500	2.500	1.250	0.750	0.500	0.357	0.268	0.208	0.167
2e+01	8.000	2.667	1.333	0.800	0.533	0.381	0.286	0.222	0.178
2e+01	8.500	2.833	1.417	0.850	0.567	0.405	0.304	0.236	0.189
2e+01	9.000	3.000	1.500	0.900	0.600	0.429	0.321	0.250	0.200
2e+01	9.500	3.167	1.583	0.950	0.633	0.452	0.339	0.264	0.211
2e+01	10.000	3.333	1.667	1.000	0.667	0.476	0.357	0.278	0.222
2e+01	10.500	3.500	1.750	1.050	0.700	0.500	0.375	0.292	0.233
2e+01	11.000	3.667	1.833	1.100	0.733	0.524	0.393	0.306	0.244
2e+01	11.500	3.833	1.917	1.150	0.767	0.548	0.411	0.319	0.256
2e+01	12.000	4.000	2.000	1.200	0.800	0.571	0.429	0.333	0.267
2e+01	12.500	4.167	2.083	1.250	0.833	0.595	0.446	0.347	0.278
3e+01	13.000	4.333	2.167	1.300	0.867	0.619	0.464	0.361	0.289
3e+01	13.500	4.500	2.250	1.350	0.900	0.643	0.482	0.375	0.300
3e+01	14.000	4.667	2.333	1.400	0.933	0.667	0.500	0.389	0.311
3e+01	14.500	4.833	2.417	1.450	0.967	0.690	0.518	0.403	0.322
3e+01	15.000	5.000	2.500	1.500	1.000	0.714	0.536	0.417	0.333
3e+01	15.500	5.167	2.583	1.550	1.033	0.738	0.554	0.431	0.344
3e+01	16.000	5.333	2.667	1.600	1.067	0.762	0.571	0.444	0.356
3e+01	16.500	5.500	2.750	1.650	1.100	0.786	0.589	0.458	0.367
3e+01	17.000	5.667	2.833	1.700	1.133	0.810	0.607	0.472	0.378
4e+01	17.500	5.833	2.917	1.750	1.167	0.833	0.625	0.486	0.389
4e+01	18.000	6.000	3.000	1.800	1.200	0.857	0.643	0.500	0.400
4e+01	18.500	6.167	3.083	1.850	1.233	0.881	0.661	0.514	0.411
4e+01	19.000	6.333	3.167	1.900	1.267	0.905	0.679	0.528	0.422
4e+01	19.500	6.500	3.250	1.950	1.300	0.929	0.696	0.542	0.433
4e+01	20.000	6.667	3.333	2.000	1.333	0.952	0.714	0.556	0.444
4e+01	20.500	6.833	3.417	2.050	1.367	0.976	0.732	0.569	0.456
4e+01	21.000	7.000	3.500	2.100	1.400	1.000	0.750	0.583	0.467
4e+01	21.500	7.167	3.583	2.150	1.433	1.024	0.768	0.597	0.478
4e+01	22.000	7.333	3.667	2.200	1.467	1.048	0.786	0.611	0.489
4e+01	22.500	7.500	3.750	2.250	1.500	1.071	0.804	0.625	0.500
5e+01	23.000	7.667	3.833	2.300	1.533	1.095	0.821	0.639	0.511
5e+01	23.500	7.833	3.917	2.350	1.567	1.119	0.839	0.653	0.522
5e+01	24.000	8.000	4.000	2.400	1.600	1.143	0.857	0.667	0.533
5e+01	24.500	8.167	4.083	2.450	1.633	1.167	0.875	0.681	0.544
5e+01	25.000	8.333	4.167	2.500	1.667	1.190	0.893	0.694	0.556
5e+01	25.500	8.500	4.250	2.550	1.700	1.214	0.911	0.708	0.567
5e+01	26.000	8.667	4.333	2.600	1.733	1.238	0.929	0.722	0.578
5e+01	26.500	8.833	4.417	2.650	1.767	1.262	0.946	0.736	0.589
5e+01	27.000	9.000	4.500	2.700	1.800	1.286	0.964	0.750	0.600
6e+01	27.500	9.167	4.583	2.750	1.833	1.310	0.982	0.764	0.611
6e+01	28.000	9.333	4.667	2.800	1.867	1.333	1.000	0.778	0.622
6e+01	28.500	9.500	4.750	2.850	1.900	1.357	1.018	0.792	0.633
6e+01	29.000	9.667	4.833	2.900	1.933	1.381	1.036	0.806	0.644
6e+01	29.500	9.833	4.917	2.950	1.967	1.405	1.054	0.819	0.656
6e+01	30.000	10.000	5.000	3.000	2.000	1.429	1.071	0.833	0.667
6e+01	30.500	10.167	5.083	3.050	2.033	1.452	1.089	0.847	0.678
6e+01	31.000	10.333	5.167	3.100	2.067	1.476	1.107	0.861	0.689
6e+01	31.500	10.500	5.250	3.150	2.100	1.500	1.125	0.875	0.700
6e+01	32.000	10.667	5.333	3.200	2.133	1.524	1.143	0.889	0.711
6e+01	32.500	10.833	5.417	3.250	2.167	1.548	1.161	0.903	0.722
7e+01	33.000	11.000	5.500	3.300	2.200	1.571	1.179	0.917	0.733
7e+01	33.500	11.167	5.583	3.350	2.233	1.595	1.196	0.931	0.744
7e+01	34.000	11.333	5.667	3.400	2.267	1.619	1.214	0.944	0.756
7e+01	34.500	11.500	5.750	3.450	2.300	1.643	1.232	0.958	0.767
7e+01	35.000	11.667	5.833	3.500	2.333	1.667	1.250	0.972	0.778
7e+01	35.500	11.833	5.917	3.550	2.367	1.690	1.268	0.986	0.789
7e+01	36.000	12.000	6.000	3.600	2.400	1.714	1.286	1.000	0.800
7e+01	36.500	12.167	6.083	3.650	2.433	1.738	1.304	1.014	0.811
7e+01	37.000	12.333	6.167	3.700	2.467	1.762	1.321	1.028	0.822
8e+01	37.500	12.500	6.250	3.750	2.500	1.786	1.339	1.042	0.833
8e+01	38.000	12.667	6.333	3.800	2.533	1.810	1.357	1.056	0.844
8e+01	38.500	12.833	6.417	3.850	2.567	1.833	1.375	1.069	0.856
8e+01	39.000	13.000	6.500	3.900	2.600	1.857	1.393	1.083	0.867
8e+01	39.500	13.167	6.583	3.950	2.633	1.881	1.411	1.097	0.878
8e+01	40.000	13.333	6.667	4.000	2.667	1.905	1.429	1.111	0.889
8e+01	40.500	13.500	6.750	4.050	2.700	1.929	1.446	1.125	0.900
8e+01	41.000	13.667	6.833	4.100	2.733	1.952	1.464	1.139	0.911
8e+01	41.500	13.833	6.917	4.150	2.767	1.976	1.482	1.153	0.922
8e+01	42.000	14.000	7.000	4.200	2.800	2.000	1.500	1.167	0.933
8e+01	42.500	14.167	7.083	4.250	2.833	2.024	1.518	1.181	0.944
9e+01	43.000	14.333	7.167	4.300	2.867	2.048	1.536	1.194	0.956
9e+01	43.500	14.500	7.250	4.350	2.900	2.071	1.554	1.208	0.967
9e+01	44.000	14.667	7.333	4.400	2.933	2.095	1.571	1.222	0.978
9e+01	44.500	14.833	7.417	4.450	2.967	2.119	1.589	1.236	0.989
9e+01	45.000	15.000	7.500	4.500	3.000	2.143	1.607	1.250	1.000
9e+01	45.500	15.167	7.583	4.550	3.033	2.167	1.625	1.264	1.011
9e+01	46.000	15.333	7.667	4.600	3.067	2.190	1.643	1.278	1.022
9e+01	46.500	15.500	7.750	4.650	3.100	2.214	1.661	1.292	1.033
9e+01	47.000	15.667	7.833	4.700	3.133	2.238	1.679	1.306	1.044
1e+02	47.500	15.833	7.917	4.750	3.167	2.262	1.696	1.319	1.056
1e+02	48.000	16.000	8.000	4.800	3.200	2.286	1.714	1.333	1.067
1e+02	48.500	16.167	8.083	4.850	3.233	2.310	1.732	1.347	1.078
1e+02	49.000	16.333	8.167	4.900	3.267	2.333	1.750	1.361	1.089
1e+02	49.500	16.500	8.250	4.950	3.300	2.357	1.768	1.375	1.100
1e+02	50.000	16.667	8.333	5.000	3.333	2.381	1.786	1.389	1.111

E-2 Experiment Matrix

Table E-5: Experiment matrix for the conducted experiment. Each participant first goes through 9 training runs, followed by 2x8 experiment runs, with one break halfway the experiment.

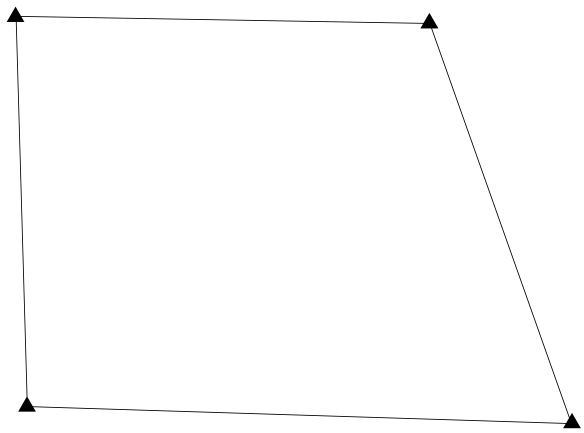
P1	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P4-1	F1P5-1	F2P7-1	F1P6-1	F2P6-1	BREAK	F1P7-1	F2P5-1	F2P4-1	F1P4-2	F1P5-2	F2P7-2	F1P6-2	F2P6-2	F1P7-2	F2P5-2	F2P4-2
P2	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P5-1	F1P6-1	F1P4-1	F1P7-1	F2P7-1	BREAK	F2P4-1	F2P6-1	F2P5-1	F1P5-2	F1P6-2	F1P4-2	F1P7-2	F2P7-2	F2P4-2	F2P6-2	F2P5-2
P3	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P6-1	F1P7-1	F1P5-1	F2P4-1	F1P4-1	BREAK	F2P5-1	F2P7-1	F2P6-1	F1P6-2	F1P7-2	F1P5-2	F2P4-2	F1P4-2	F2P5-2	F2P7-2	F2P6-2
P4	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P7-1	F2P4-1	F1P6-1	F2P5-1	F1P5-1	BREAK	F2P6-1	F1P4-1	F2P7-1	F1P7-2	F2P4-2	F1P6-2	F2P5-2	F1P5-2	F2P6-2	F1P4-2	F2P7-2
P5	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P4-1	F2P5-1	F1P7-1	F2P6-1	F1P6-1	BREAK	F2P7-1	P1P5-1	F1P4-1	F2P4-2	F2P5-2	F1P7-2	F2P6-2	F1P6-2	F2P7-2	P1P5-2	F1P4-2
P6	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P5-1	F2P6-1	F2P4-1	F2P7-1	F1P7-1	BREAK	F1P4-1	F1P6-1	F1P5-1	F2P5-2	F2P6-2	F2P4-2	F2P7-2	F1P7-2	F1P4-2	F1P6-2	F1P5-2
P7	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P6-1	F2P7-1	F2P5-1	F1P4-1	F2P4-1	BREAK	F1P5-1	F1P7-1	F1P6-1	F2P6-2	F2P7-2	F2P5-2	F1P4-2	F2P4-2	F1P5-2	F1P7-2	F1P6-2
P8	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P7-1	F1P4-1	F2P6-1	F1P5-1	F2P5-1	BREAK	F1P6-1	F2P4-1	F1P7-1	F2P7-2	F1P4-2	F2P6-2	F1P5-2	F2P5-2	F1P6-2	F2P4-2	F1P7-2
P9	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P4-1	F1P5-1	F2P7-1	F1P6-1	F2P6-1	BREAK	F1P7-1	F2P5-1	F2P4-1	F1P4-2	F1P5-2	F2P7-2	F1P6-2	F2P6-2	F1P7-2	F2P5-2	F2P4-2
P10	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P5-1	F1P6-1	F1P4-1	F1P7-1	F2P7-1	BREAK	F2P4-1	F2P6-1	F2P5-1	F1P5-2	F1P6-2	F1P4-2	F1P7-2	F2P7-2	F2P4-2	F2P6-2	F2P5-2
P11	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P6-1	F1P7-1	F1P5-1	F2P4-1	F1P4-1	BREAK	F2P5-1	F2P7-1	F2P6-1	F1P6-2	F1P7-2	F1P5-2	F2P4-2	F1P4-2	F2P5-2	F2P7-2	F2P6-2
P12	T1	T2	T3	T4	T5	T6	T7	T8	T9	F1P7-1	F2P4-1	F1P6-1	F2P5-1	F1P5-1	BREAK	F2P6-1	F1P4-1	F2P7-1	F1P7-2	F2P4-2	F1P6-2	F2P5-2	F1P5-2	F2P6-2	F1P4-2	F2P7-2
P13	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P4-1	F2P5-1	F1P7-1	F2P6-1	F1P6-1	BREAK	F2P7-1	P1P5-1	F1P4-1	F2P4-2	F2P5-2	F1P7-2	F2P6-2	F1P6-2	F2P7-2	P1P5-2	F1P4-2
P14	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P5-1	F2P6-1	F2P4-1	F2P7-1	F1P7-1	BREAK	F1P4-1	F1P6-1	F1P5-1	F2P5-2	F2P6-2	F2P4-2	F2P7-2	F1P7-2	F1P4-2	F1P6-2	F1P5-2
P15	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P6-1	F2P7-1	F2P5-1	F1P4-1	F2P4-1	BREAK	F1P5-1	F1P7-1	F1P6-1	F2P6-2	F2P7-2	F2P5-2	F1P4-2	F2P4-2	F1P5-2	F1P7-2	F1P6-2
P16	T1	T2	T3	T4	T5	T6	T7	T8	T9	F2P7-1	F1P4-1	F2P6-1	F1P5-1	F2P5-1	BREAK	F1P6-1	F2P4-1	F1P7-1	F2P7-2	F1P4-2	F2P6-2	F1P5-2	F2P5-2	F1P6-2	F2P4-2	F1P7-2

E-3 Training Scenarios

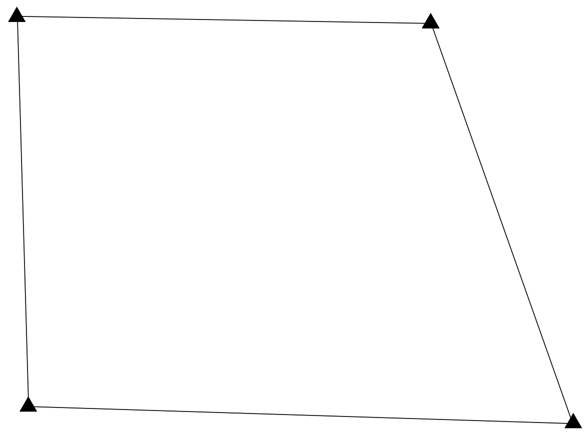
Training scenarios.

Table E-6: Training Runs

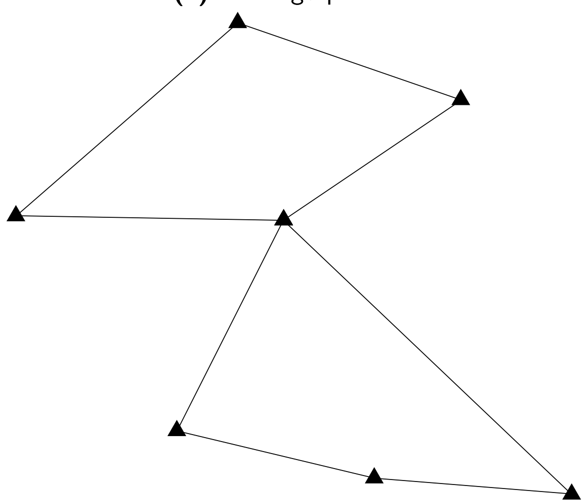
	Category	Scenario			
T1	Interface Familiarization	Flight Time Constraint			
T2	Interface Familiarization	Payload Constraint			
T3	Interface Familiarization	Depot Congestion			
T4	DVRP	nCustomers: 6	maxPayload: 5	nVehicles: 3	nFailures: 1
T5	DVRP	nCustomers: 10	maxPayload: 5	nVehicles: 4	nFailures: 1
T6	DVRP	nCustomers: 15	maxPayload: 5	nVehicles: 4	nFailures: 1
T7	DVRP	nCustomers: 25	maxPayload: 5	nVehicles: 7	nFailures: 1
T8	DVRP	nCustomers: 56	maxPayload: 8	nVehicles: 8	nFailures: 1
T9	DVRP	nCustomers: 72	maxPayload: 9	nVehicles: 9	nFailures: 1



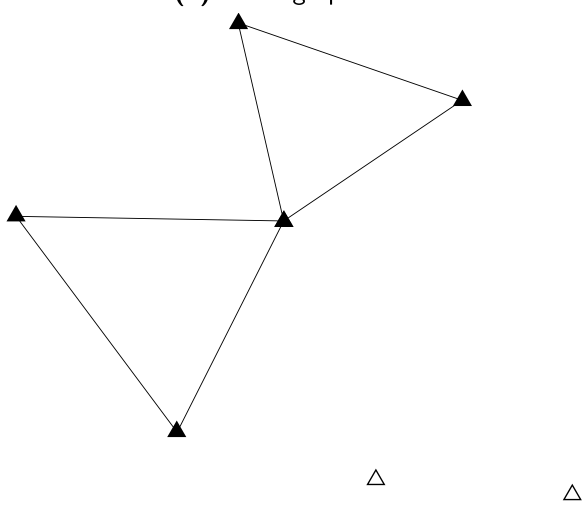
(a) training0-problem



(b) training1-problem

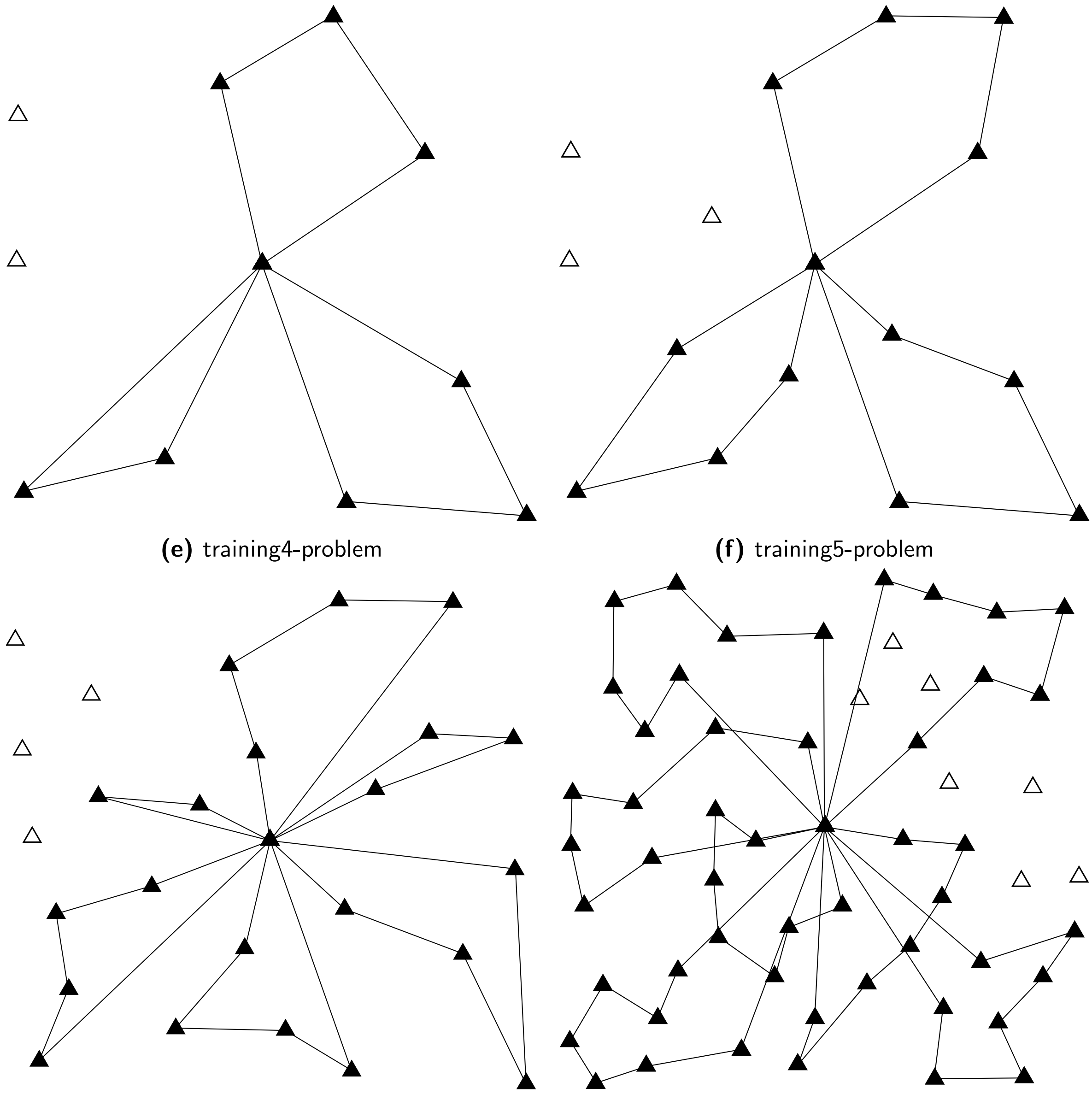


(c) training2-problem



(d) training3-problem

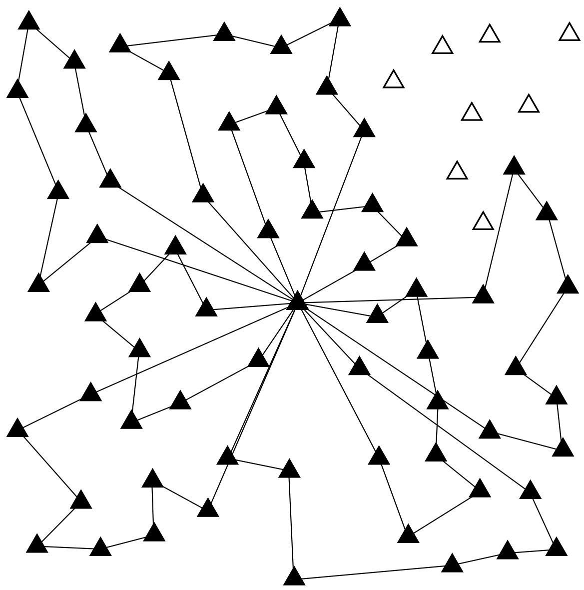
Figure E-1: Training scenarios



(e) training4-problem **(f)** training5-problem

(g) training6-problem **(h)** training7-problem

Figure E-1: Training scenarios (continued)

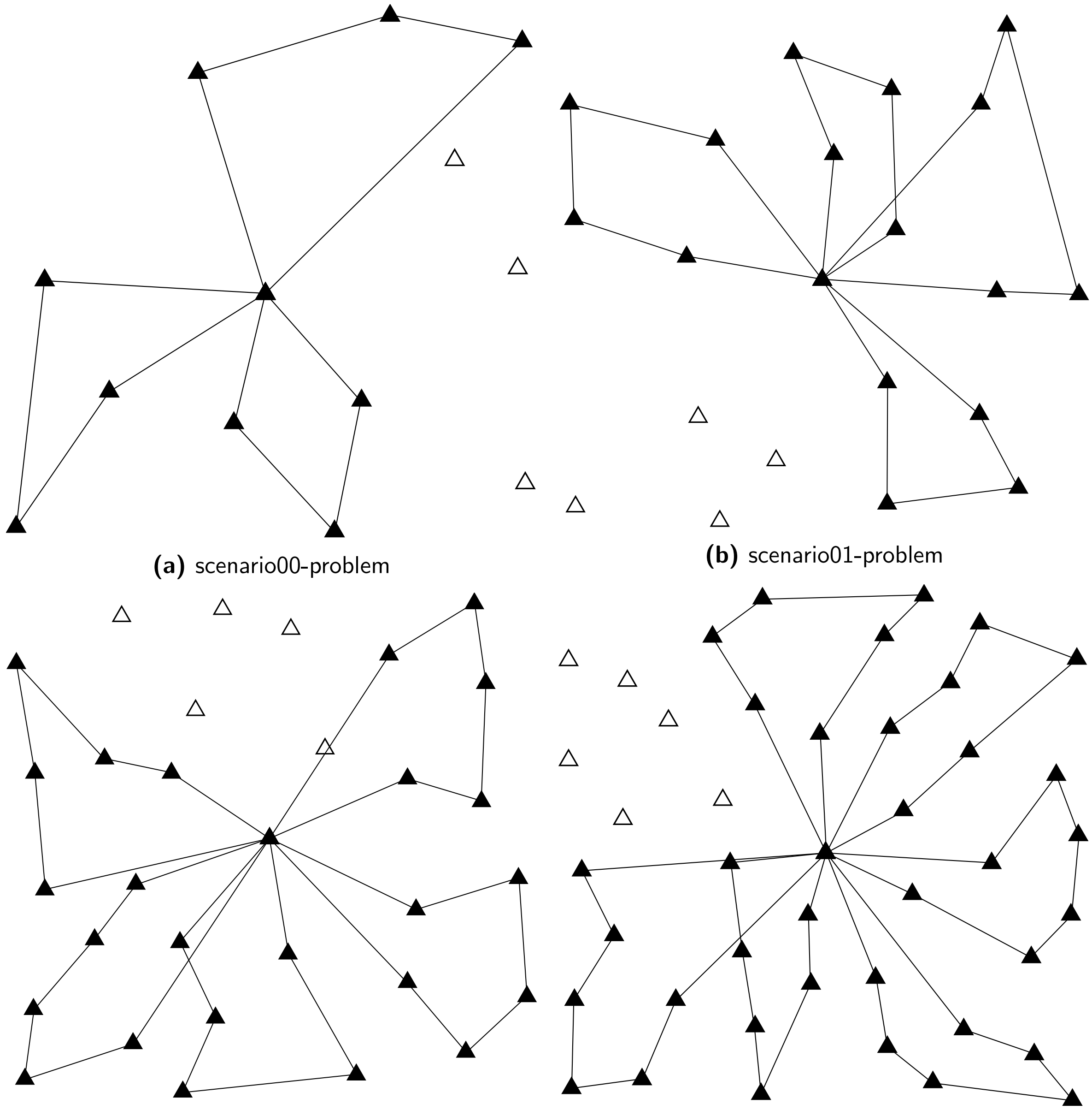


(i) training8-problem

Figure E-1: Training scenarios (continued)

E-4 Experiment Scenarios

Experiment scenarios.



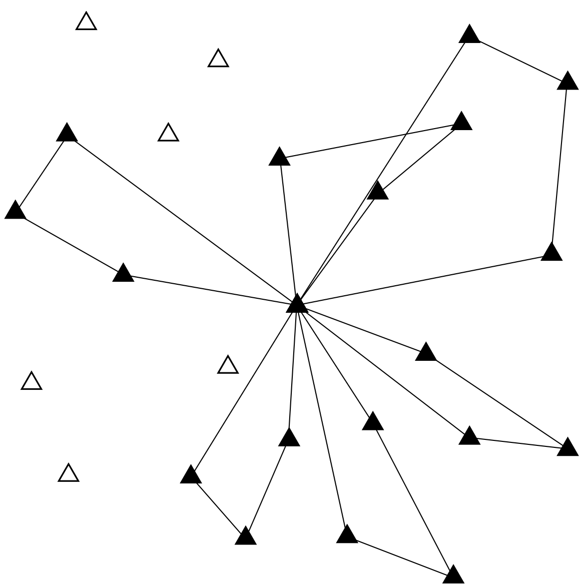
(a) scenario00-problem

(b) scenario01-problem

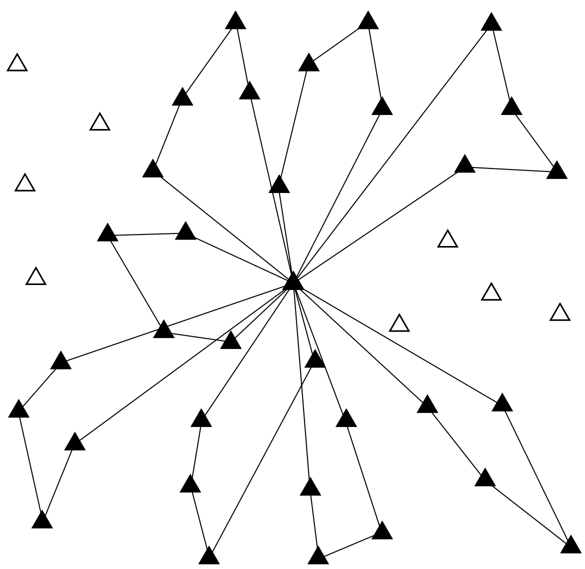
(c) scenario02-problem

(d) scenario03-problem

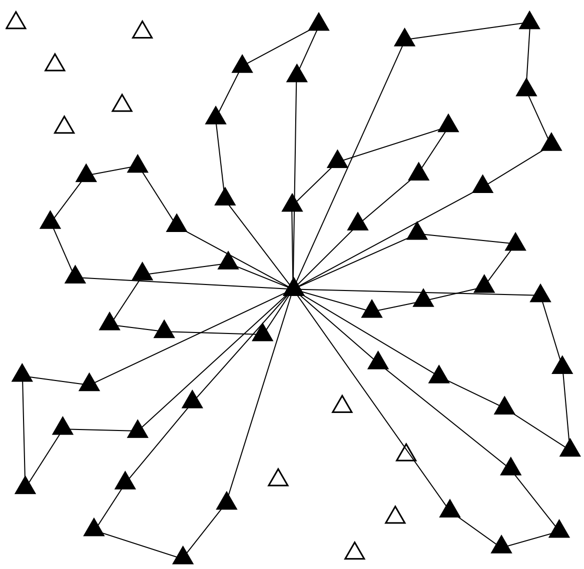
Figure E-2: Experiment scenarios



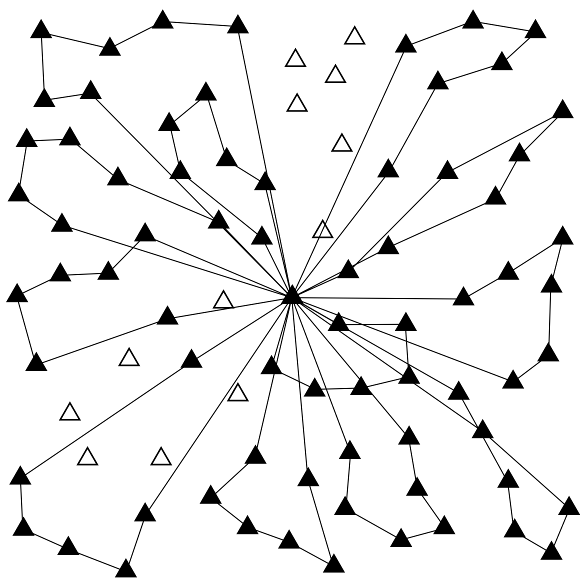
(e) scenario04-problem



(f) scenario05-problem

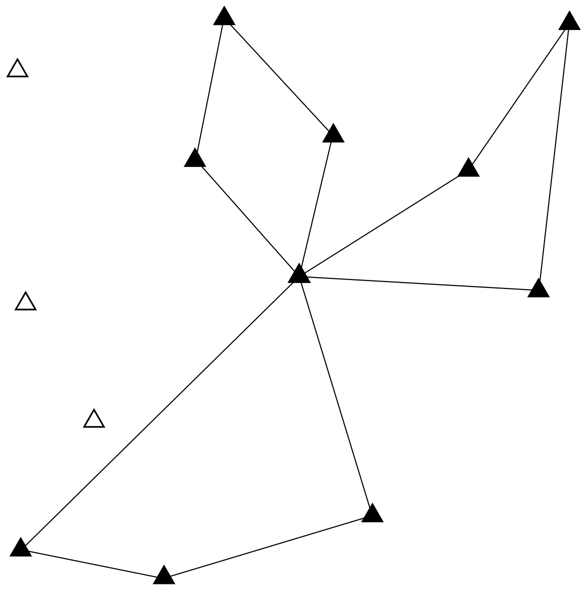


(g) scenario06-problem

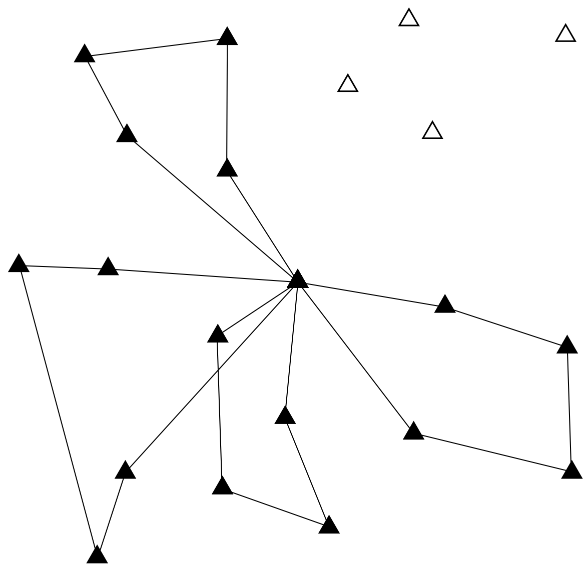


(h) scenario07-problem

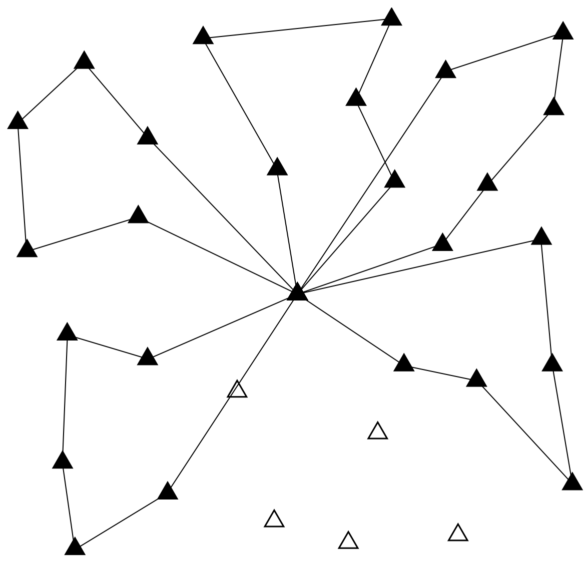
Figure E-2: Experiment scenarios (continued)



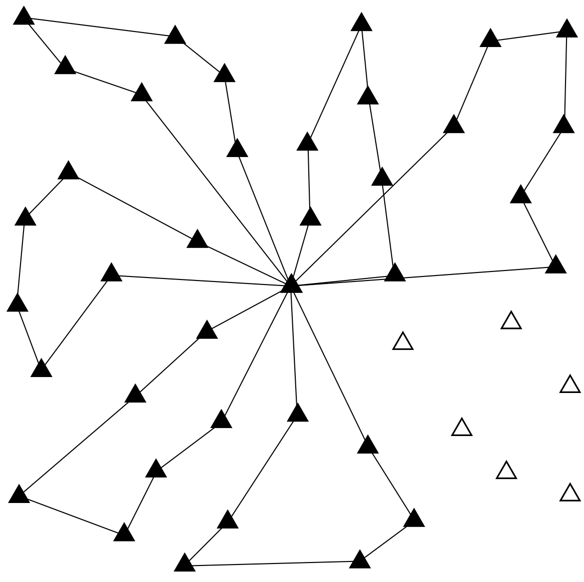
(i) scenario08-problem



(j) scenario09-problem

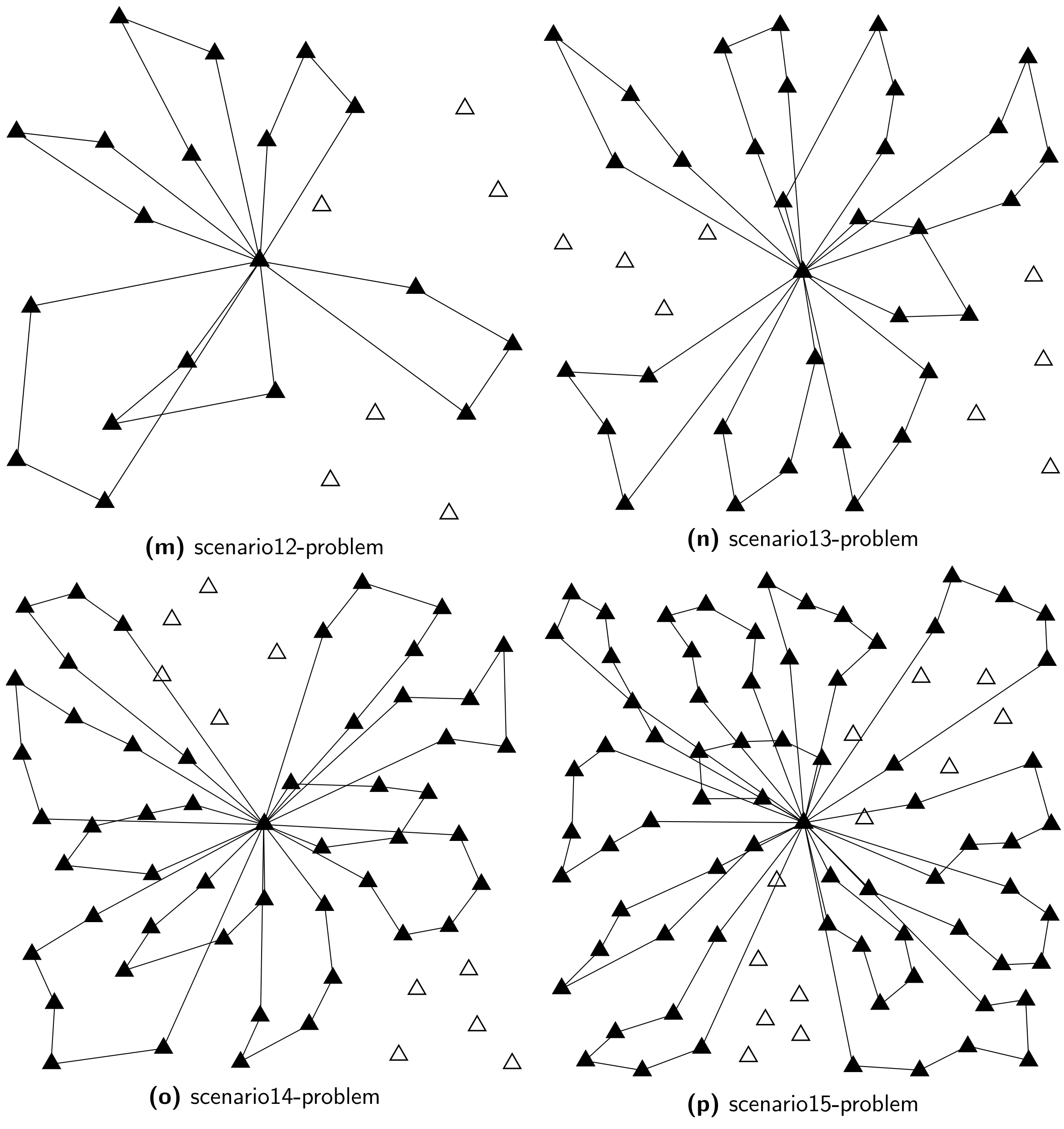


(k) scenario10-problem



(l) scenario11-problem

Figure E-2: Experiment scenarios (continued)



(m) scenario12-problem

(n) scenario13-problem

(o) scenario14-problem

(p) scenario15-problem

Figure E-2: Experiment scenarios (continued)

Appendix F

Experiment Briefing

F-1 Introduction

Thank you for participating in this experiment! The goal of this experiment is to investigate human control performance in multi-UAV (Unmanned Aerial Vehicle) dynamic vehicle routing problems (DVRP). Consider a scenario in which payload needs to be delivered to customer locations using multiple payload carrying UAVs. The vehicles all start and end their flight at the depot. The assignment of all the customer locations to specific vehicles is the essence of the VRP. Figure F-1 shows a visual representation of an example mission.

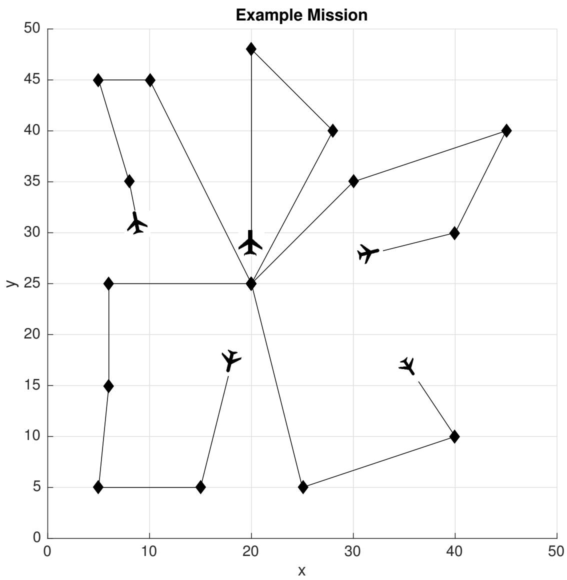


Figure F-1: Example mission of a multi-UAV vehicle routing problem, with the depot at the center (20;25), 5 vehicles and 14 customer locations.

In this experiment, it is your task to mitigate the effects caused by UAV failures during several multi-UAV payload delivery missions.

You will be presented with several VRP scenarios in which vehicle failures will result in unassigned customer locations. These unassigned locations will somehow have to be included in the flightplans of the remaining vehicles, while satisfying all constraints (flight time, payload capacity and depot capacity). Note that all vehicle failures will occur at the same time and at the start of the scenario. You will thus not have to account for extra vehicle failures during the scenario.

Please consider the following goals during the execution of your control task:

1. **Assign all customer locations to the fleet of UAVs, while satisfying flight time, payload capacity and depot capacity constraints.**
2. **Optimize all UAV routes for shortest distance.**

The experiment starts with a number of training scenarios, which allow you to familiarize yourself with the control task and the interface. At the end of each scenario you will fill out the web-survey. You are also asked to fill out the web-survey for a large portion of the training scenarios to familiarize yourself with the usage of this tool.

Note: each scenario is time limited to 6 minutes, which means you only have limited time to identify the failed vehicles and come up with an updated routing.

An explanation of the experiment setup is up next, followed by a review of the training scenarios, which you will use as a guide to follow along. If you have any questions when reading the subsequent chapters or during the training phase of the experiment, please do not hesitate to ask!

F-2 Experiment Setup

The interface you will use consists of three distinct windows: the map view, the payload view and the timeline view. Each of these windows show relevant information for successful execution of your control task. Figure F-2 shows a visual representation of the interface and experiment setup you will use.

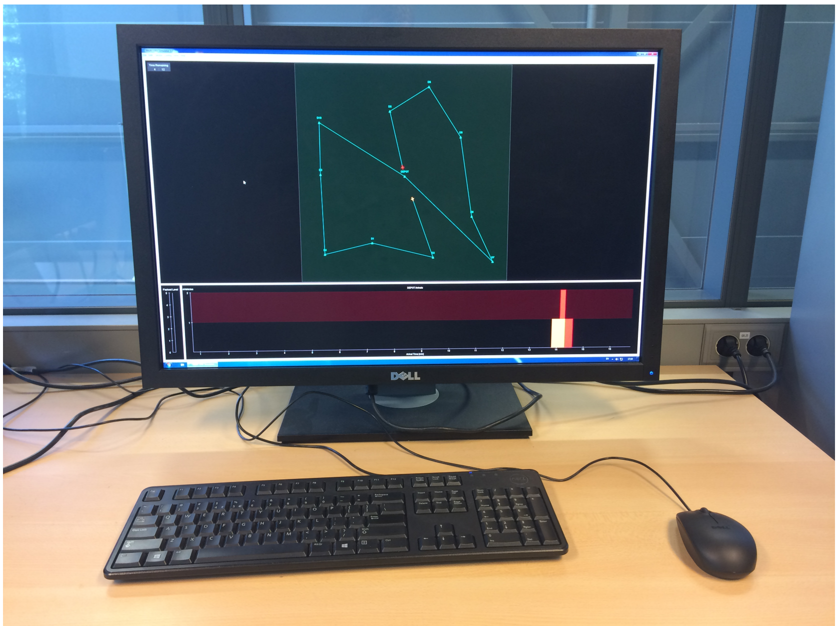


Figure F-2: Experiment setup consisting of keyboard, mouse, display and desktop PC (not shown).

The map view (as depicted in Figure F-3b and Figure F-3e) shows all UAVs, their routes and all customer locations. Also, after selecting a vehicle, the area that is within range given the available flight time is shown. If relevant, superimposed on this view is the area that needs to be avoided when the depot capacity is exceeded. This will help you in determining the required path stretching to introduce sufficient delay to solve the depot conflict.

The payload view (as depicted in Figure F-3a and Figure F-3d) shows the remaining payload for the selected vehicle. It will also show this information when hovering over a UAV without selecting it. When zero payload remaining is reached and when insufficient payload is available for successful mission execution, a corresponding textual warning will appear. Also, the color of the payload level indicator corresponds to the UAV color in the map view display.

The timeline view (as depicted in Figure F-3c and Figure F-3f) shows the arrival times of all vehicles at the depot. After arrival at the depot, UAVs require a 30 second service window. Due to constraints at the depot, the amount of vehicles allowed to arrive at the same time is limited. This constraint is indicated by the red zone in the timeline view. Also, the color of the arrival block corresponds to the color associated with the payload level.

The following control inputs are available for you to interact with the interface:

- *P:* start scenario
- *LMB click:* select UAV / select leg
- *RMB click:* deselect UAV / deselect leg / discard FLTPLN change
- *Enter:* confirm FLTPLN change
- *CTRL + LBM click + leg selected:* add WPT
- *CTRL + LBM click + UAV selected:* remove WPT

The following colors are used in the interface:

- *Grey:* flightplan and customer locations associated with UAV that is to depart the depot in the future.
- *Cyan:* flightplan and customer locations associated with inactive UAV.
- *Magenta:* flightplan and customer locations associated with active UAV.
- *White:* flightplan and customer locations associated with active and modified UAV flightplan.
- *Green:* selected UAV.
- *Amber:* unvisited customer locations.
- *Red:* infeasibility due to insufficient available flight time, payload or depot capacity.
- *Several shades of yellow:* UAV payload level.

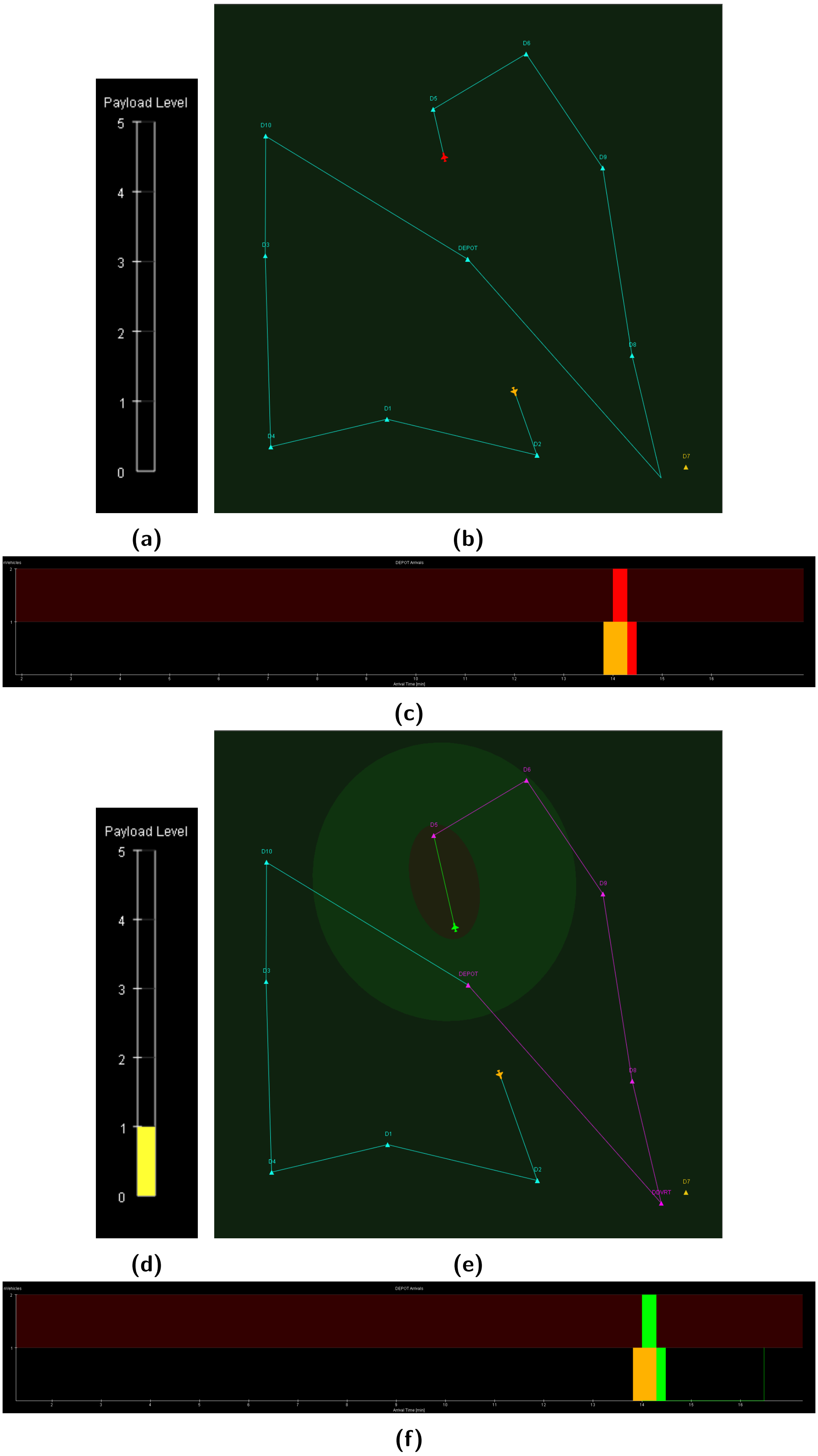


Figure F-3: Interface views for an example inactive and active UAV case.

F-3 Training Scenarios

You will now go through a number of training scenarios to familiarize yourself with solving dynamic vehicle routing problems and with the use of the interface. Remember your goals: assign all customer locations to the fleet of UAVs, while satisfying all constraints and optimize all UAV routes for shortest distance. Please double-click on the “UAV-sim” icon on the desktop, fill out your participant ID, press “Enter”, and maximize the screen.

F-4 Training 1: Flight Time Constraint

Observe the cyan colored depot in the center of the map view, the two amber colored unassigned customer locations at the top and the dashed gray colored flightplan with customer locations associated with a UAV that is to depart the depot in the future in the bottom left.

≫ Click inside the map view and press “P”.

Observe the gray flightplan with associated customer locations now turn cyan, indicating a flying but inactive UAV.

≫ Select (LMB) the UAV.

Observe the payload level in the payload view and the flight time available in both the map and timeline view (green vertical line depicts latest feasible depot arrival time). Now incorporate the customer location that lies within range into the flightplan.

≫ Select (LMB) a specific flightplan leg.

≫ Add (CTRL + LMB) the customer location.

≫ Confirm (Enter) the changed flightplan.

Now also incorporate the other unvisited customer location into the flightplan.

≫ Select (LMB) the UAV.

≫ Select (LMB) a specific flightplan leg.

≫ Add (CTRL + LMB) the customer location.

Observe, the red leg, indicating the UAV will not be able to fly back from the last waypoint to the depot location.

≫ Confirm (Enter) the changed flightplan.

Observe, the same red leg, but now also the red UAV icon and the red arrival block indicating an infeasibility.Proceed to remove the last customer location from the flightplan.

≫ Select (LMB) the UAV.

≫ Remove (CTRL + LBM) the last waypoint.

≫ Confirm (Enter) the changed flightplan.

Lastly,

≫ Select (LMB) the UAV.

≫ Select (LMB) a specific flightplan leg.

≫ Add (CTRL + LMB) a waypoint at any desired location.

≫ Deselect (RMB) the UAV.

Observe the changes made to the flightplan now being discarded.

Please now close the simulator. Double-click on the "UAV-sim" icon on the desktop, fill out your participant ID, press "Enter", and maximize the screen to get set up for the next training scenario.

F-5 Training 2: Payload Constraint

This scenario is similar to the previous one. Observe the single gray UAV flightplan and unassigned customer locations.

≫ Click inside the map view and press "P".

≫ Include all unassigned customer locations in the UAV flightplan.

Observe, after including the last unassigned customer location into the UAV flightplan, the waypoint color change to red. Also note the red UAV icon color and red arrival block indicating an infeasibility.

≫ Hover over the UAV and notice the textual warning in the payload view.

≫ Remove one of the customer locations from the flightplan, to make it feasible again.

≫ Remove some more customer locations from the UAV flightplan.

Notice the payload level indicator come up. Also notice the change of color of the UAV icon, the payload level indicator, and the arrival block to more and more brighter shades of yellow.

Please now close the simulator and re-open it to get set up for the next training scenario.

F-6 Training 3: Depot Congestion

Observe the gray UAV flightplans, the payload capacity and the depot arrival constraint.

≫ Click inside the map view and press "P".

Notice the red UAV icon and the red arrival block.

≫ Select the red arrival block in the timeline view.

≫ Select the last flightplan leg (the one connected to the depot) in the map view.

Observe the red zone around the flightplan leg.

≫ Place a waypoint inside the red zone.

≫ Select the UAV and the leg again and place a waypoint outside the red zone.

Notice how the red zone helps you to introduce sufficient path stretching to solve the conflict. It is also possible to solve depot arrival infeasibilities by rearranging customer locations within a flightplan or by exchanging customer locations between UAVs.

Please now close the simulator and re-open it to get set up for the next training scenario.

F-7 Training 4: DVRP

Now that you are familiar with using the interface, it is time to solve the first DVRP. Observe the gray UAV flightplans, the payload capacity and the depot arrival constraint (constraints will change from scenario to scenario!).

Note that it is possible to claim a (grey colored) customer location from a UAV that has not left the depot yet by adding it to a flightplan of an active vehicle (try this during training!). If you want to exchange waypoints between UAVs that already left the depot, you need to unassign first, then select the other vehicle and assign the customer location there.

≫ Click inside the map view and press "P".

Notice the UAV failure.

≫ Incorporate all unassigned customer locations in the flightplans of the remaining UAVs.

When you are satisfied with the new routing you created,

≫ Close the simulator.

≫ Fill out the web-survey.

≫ Start the simulator.

F-8 Training 5-9: DVRP

Whenever you are ready,

≫ Click inside the map view and press "P".

When you are satisfied with the new routing you created,

≫ Close the simulator.

≫ Fill out the web-survey.

≫ Start the simulator.

You are now ready to start the experiment! Remind yourself of the goals of your control task and the 6 minute time limit on the scenarios. Also, please do not close the simulator before the timer is at zero. Good luck!

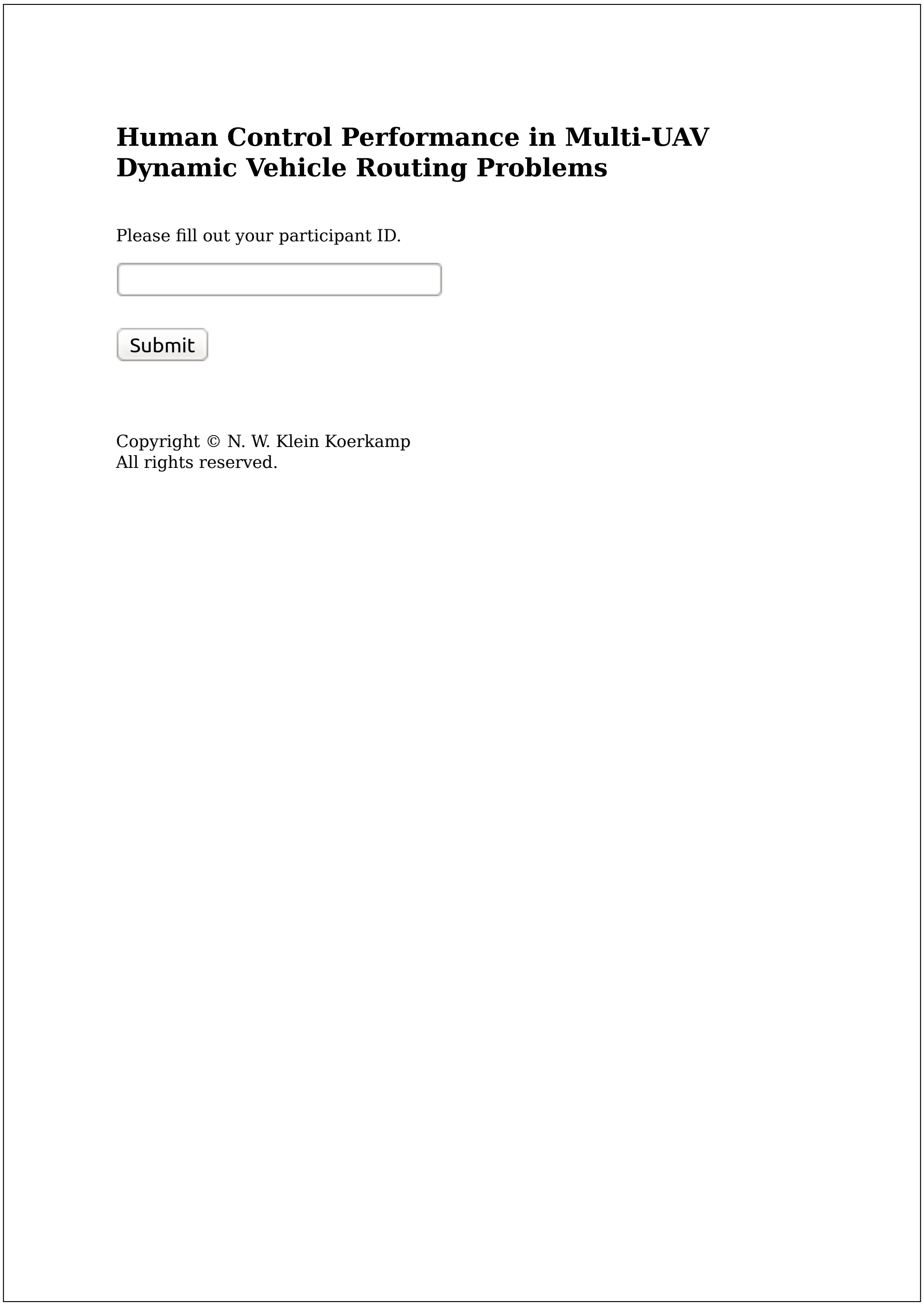
Appendix G

Experiment Survey

A web form was created to present and process the participant surveys during the experiment. The surveys consisted of an intake survey, post scenario survey and a post experiment survey. The web form was presented to the participant in a web browser on a secondary display. By running a local web server with PHP support, the web form is able to automatically capture user input, process all data and write the data to a data file.

The remainder of this chapter presents a detailed representation of the content and layout of the web form.

G-1 Intake Survey



Human Control Performance in Multi-UAV Dynamic Vehicle Routing Problems

Please fill out your participant ID.

Submit

Human Control Performance in Multi-UAV Dynamic Vehicle Routing Problems

Please fill out the following questions.

Age:

Gender:

○ Male ○ Female

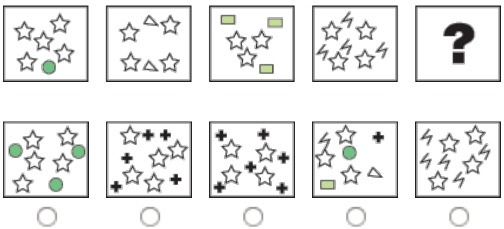
Language:

○ Dutch ○ English

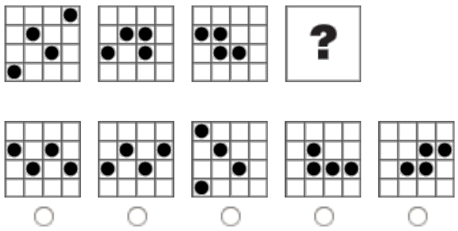
Do you consider yourself a regular video game player?

○ No ○ Yes

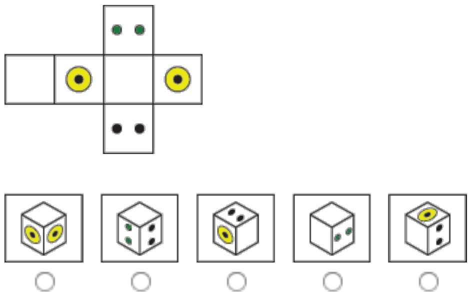
Q1: Which of the bottom figures should logically take the place of the question mark in the upper set?



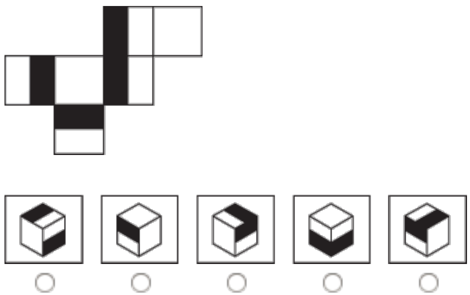
Q2: Which of the bottom figures should logically take the place of the question mark in the upper set?



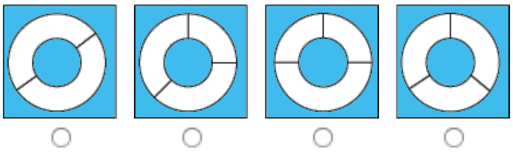
Q3: Which cube results when folding the unfolded cube?



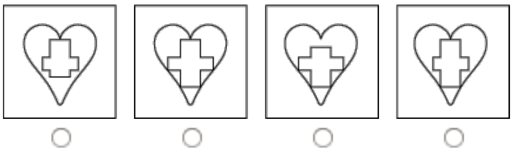
Q4: Which cube results when folding the unfolded cube?



Q5: Which of the bottom figures can be composed of the individual parts?



Q6: Which of the bottom figures can be composed of the individual parts?

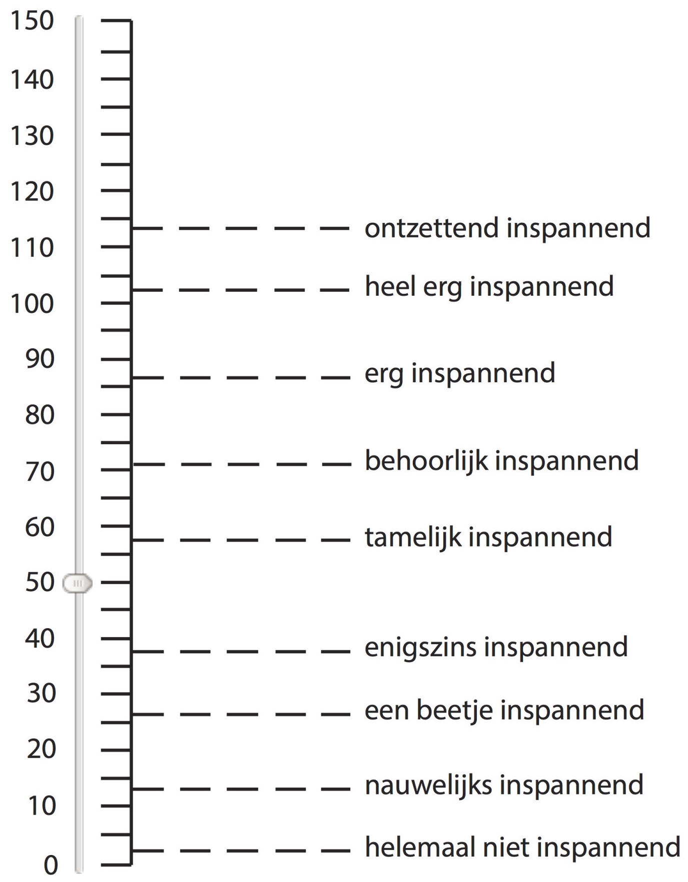


Submit

G-2 Post Scenario Survey

Human Control Performance in Multi-UAV Dynamic Vehicle Routing Problems

Please fill out the Rating Scale Mental Effort.



Please fill out the following questions.

Please give an estimate of the relative amount of time you spent interacting with the 3 display elements (0-100%, total 100%).

Map View

Timeline View

Payload View

How would you describe the way you came up with a solution for the scenario?

- Tactical
- Strategic

Tactical: using local solutions to achieve overall goal.
Strategic: executing a pre-defined plan.

How would you describe the way you came up with a solution for the scenario?

- Satisfice
- Optimize

Satisfice: achieving a solution that achieves the overall goal.
Optimize: achieving the best solution to achieve the overall goal.

Submit

G-3 Post Experiment Survey

Human Control Performance in Multi-UAV Dynamic Vehicle Routing Problems

Please fill out the following questions.

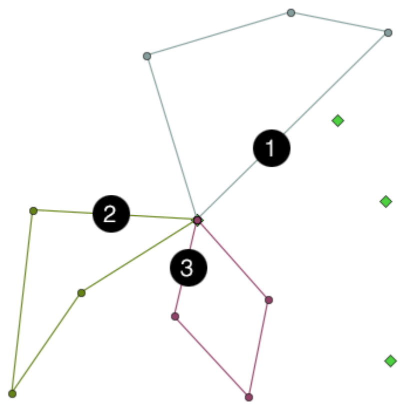
How do you assess the usefulness and the functionality of the map view? Please provide examples in your elaboration.

How do you assess the usefulness and the functionality of the timeline view? Please provide examples in your elaboration.

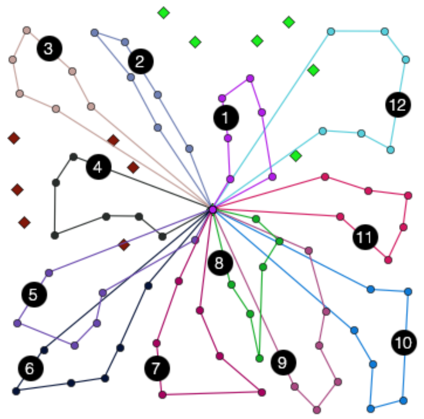
How do you assess the usefulness and the functionality of the payload view? Please provide examples in your elaboration.

How do you assess the usefulness and clarity of the color use in the display?
Please provide examples in your elaboration.

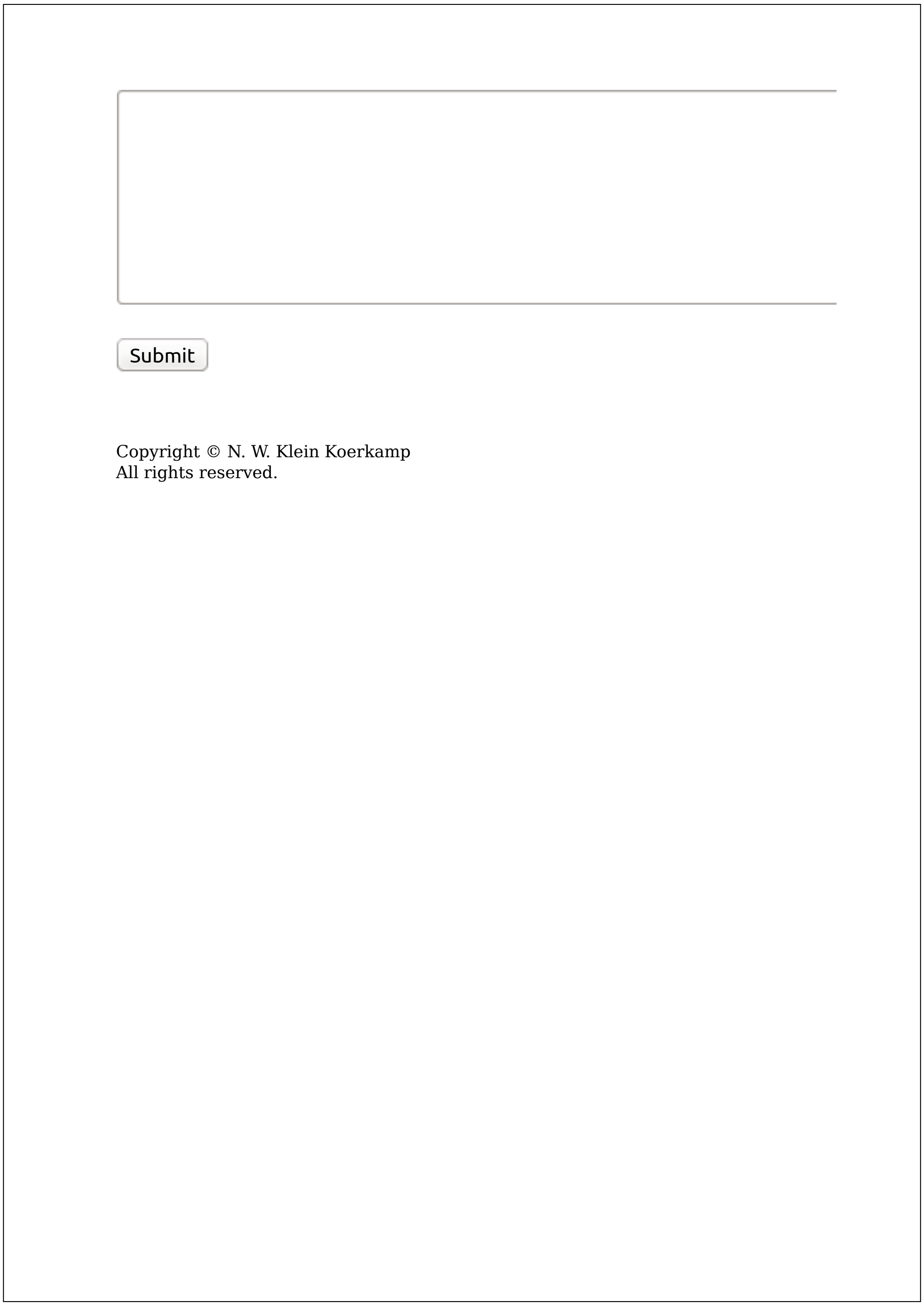
How would you solve the scenario depicted below (high-level description)?
NOTE: each vehicle only has one spare payload item on board.



How would you solve the scenario depicted below (high-level description)?
NOTE: each vehicle only has one spare payload item on board.



Do you have any other comments or suggestions with respect to the interface or the experiment?


Submit

Copyright © N. W. Klein Koerkamp
All rights reserved.

Appendix H

Experiment Results

H-1 Participant Characteristics

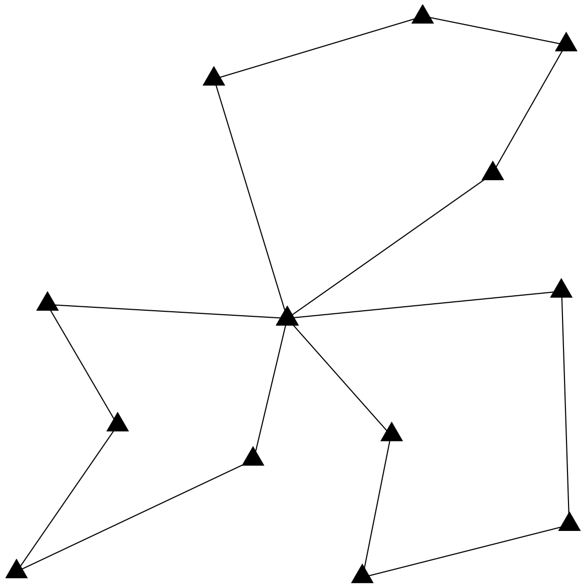
Table H-1 summarizes the characteristics of the participants that participated in the experiment, which were obtained from the intake-survey.

Table H-1: Participant Characteristics

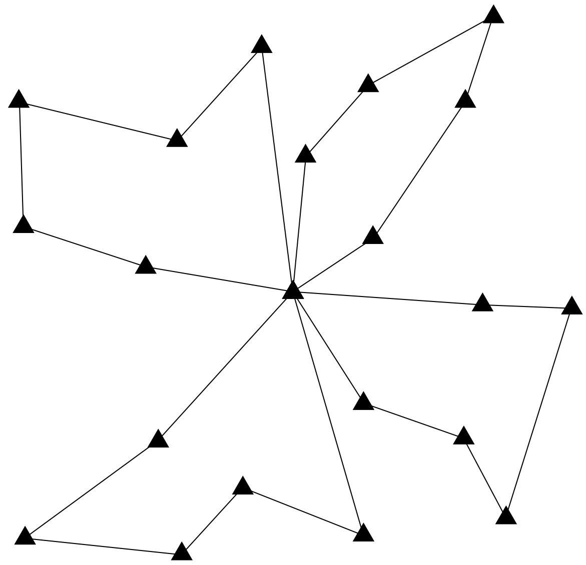
	Age	Gender	Language	Gamer	Test Score	Test Time (s)
P1	26	Male	Dutch	False	6	175
P2	24	Male	Dutch	True	6	80
P3	26	Male	Dutch	True	6	319
P4	24	Male	English	False	5	261
P5	28	Male	English	False	6	153
P6	25	Female	Dutch	False	5	141
P7	29	Male	Dutch	False	5	186
P8	25	Male	Dutch	False	6	191
P9	24	Male	Dutch	True	6	197
P10	24	Male	Dutch	False	4	190
P11	25	Male	Dutch	True	6	281
P12	25	Male	Dutch	True	6	220
P13	23	Male	Dutch	False	6	262
P14	25	Male	Dutch	True	6	215
P15	26	Male	Dutch	False	4	419
P16	26	Male	Dutch	True	6	174

H-2 Optimized Solutions

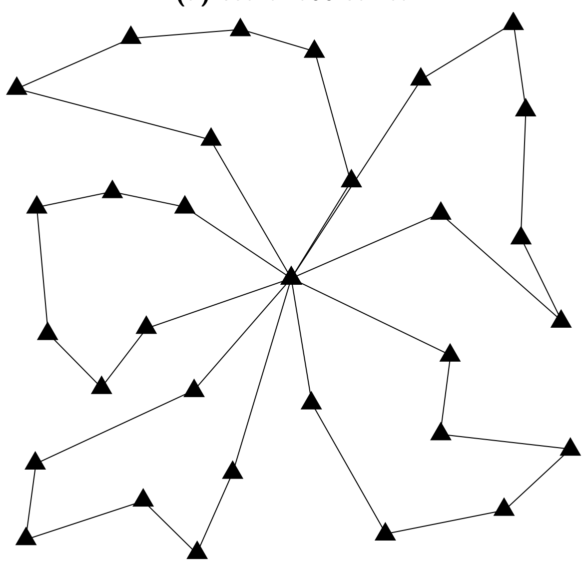
Optimized solutions to experiment scenarios.



(a) scenario00-solved



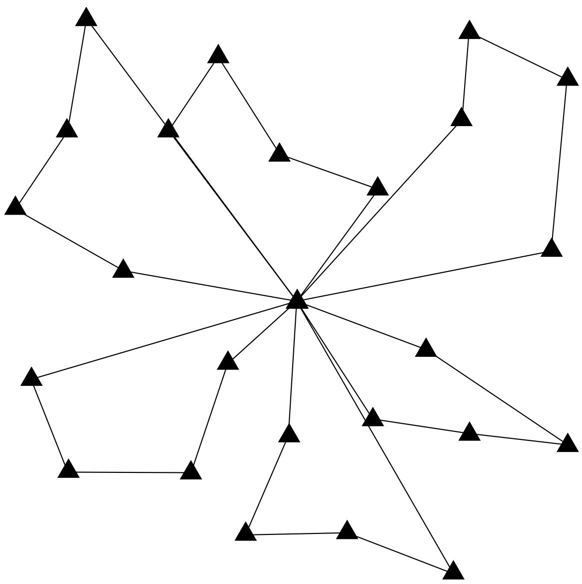
(b) scenario01-solved



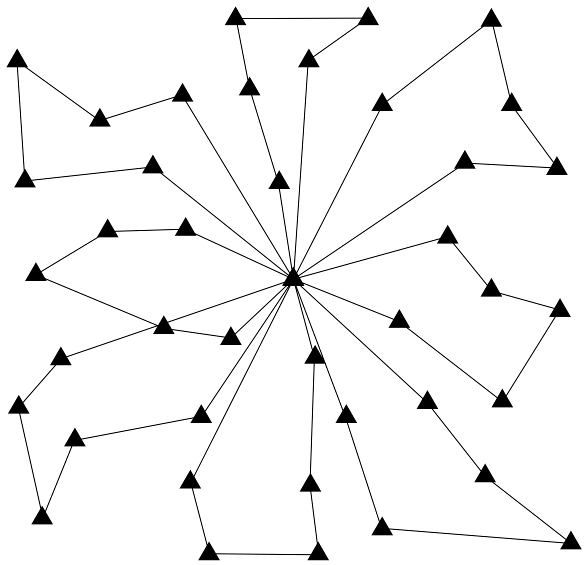
(c) scenario02-solved

(d) scenario03-no-solution

Figure H-1: Optimized solutions to experiment scenarios.

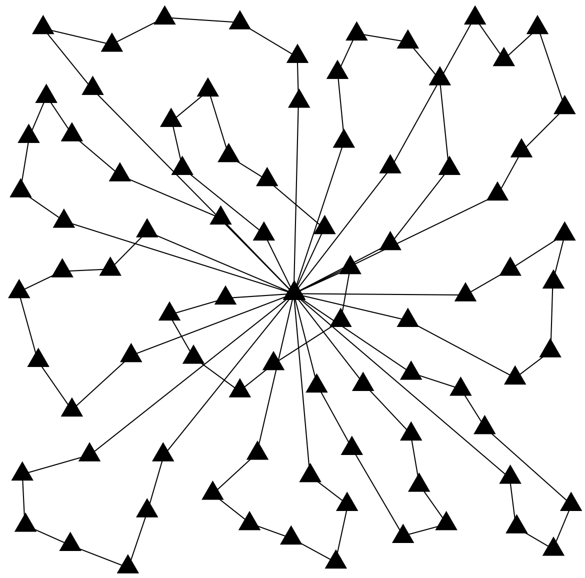


(e) scenario04-solved



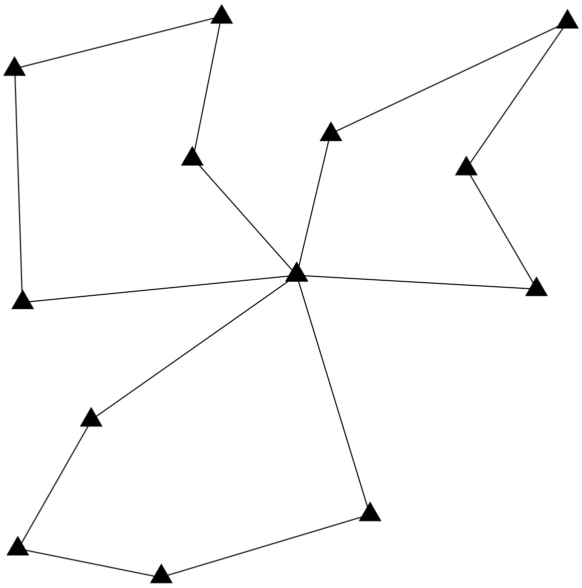
(f) scenario05-solved

(g) scenario06-no-solution

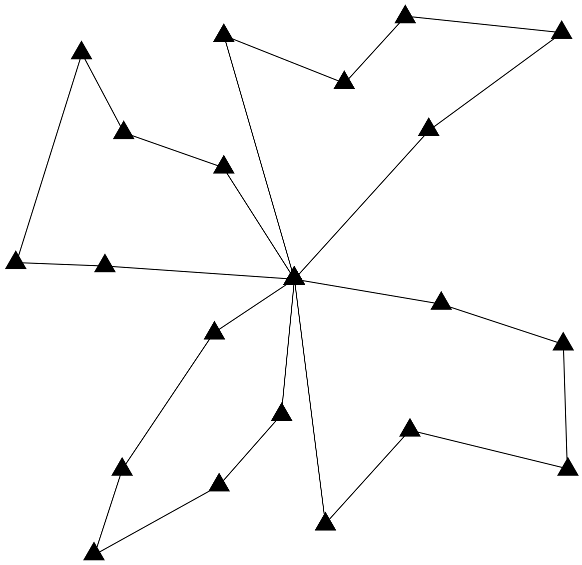


(h) scenario07-solved

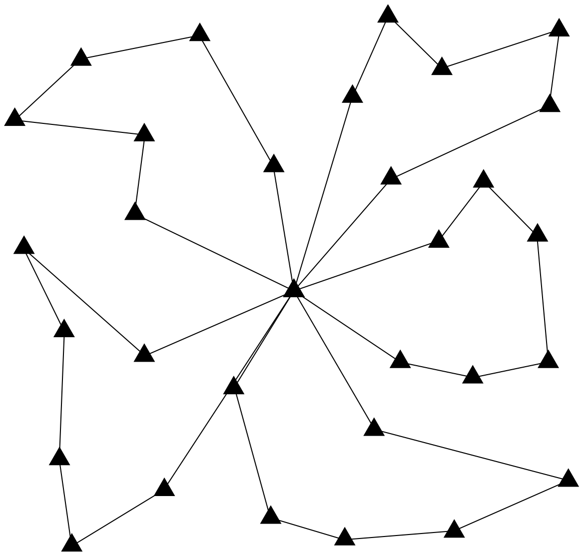
Figure H-1: Optimized solutions to experiment scenarios (continued).



(i) scenario08-solved



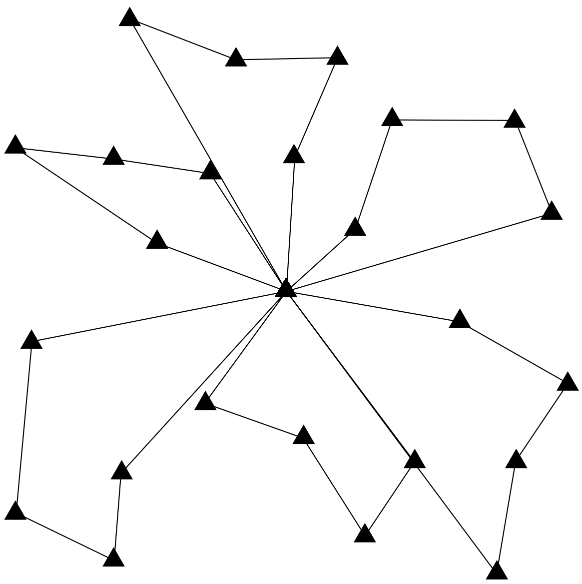
(j) scenario09-solved



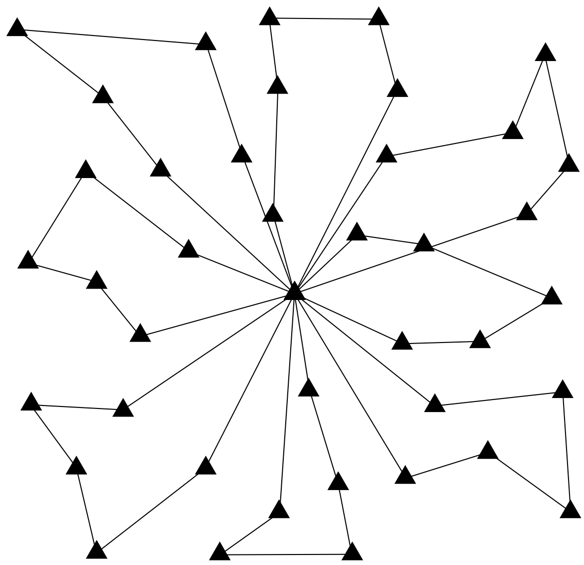
(k) scenario10-solved

(l) scenario11-no-solution

Figure H-1: Optimized solutions to experiment scenarios (continued).

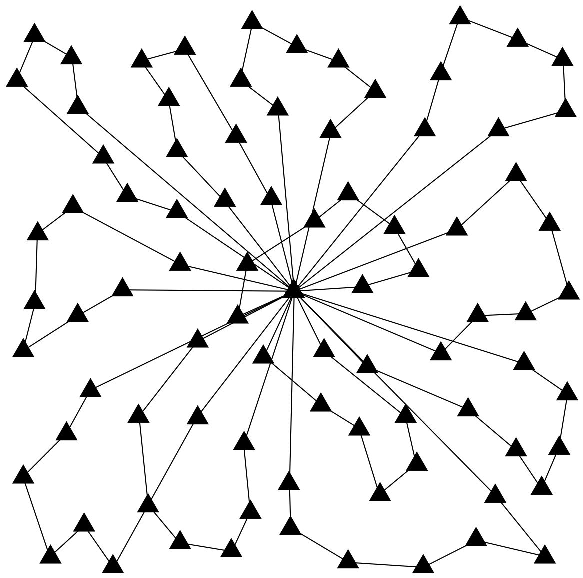


(m) scenario12-solved



(n) scenario13-solved

(o) scenario14-no-solution



(p) scenario15-solved

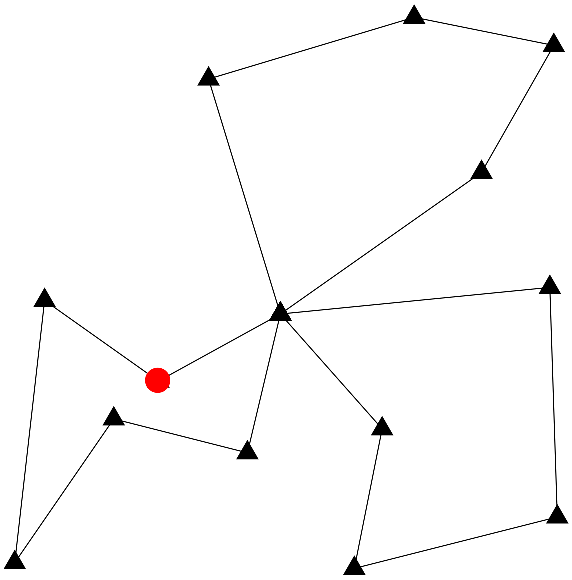
Figure H-1: Optimized solutions to experiment scenarios (continued).

H-3 Participant Solutions

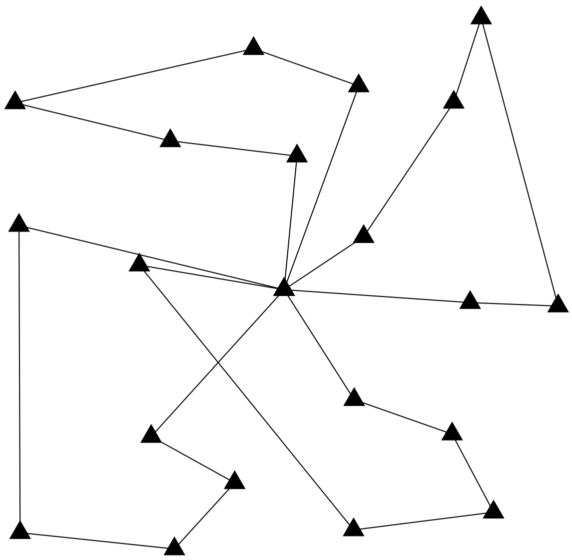
This chapter presents all participant solutions to all experiment scenarios. The red circles represent path stretch waypoints that were used by participants for delay allocation in case of exceeding the depot capacity constraint.

H-3-1 Participant 1

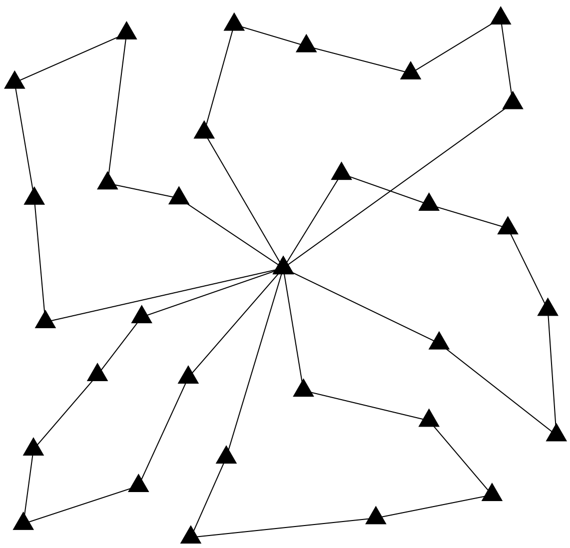
Participant 1 solutions.



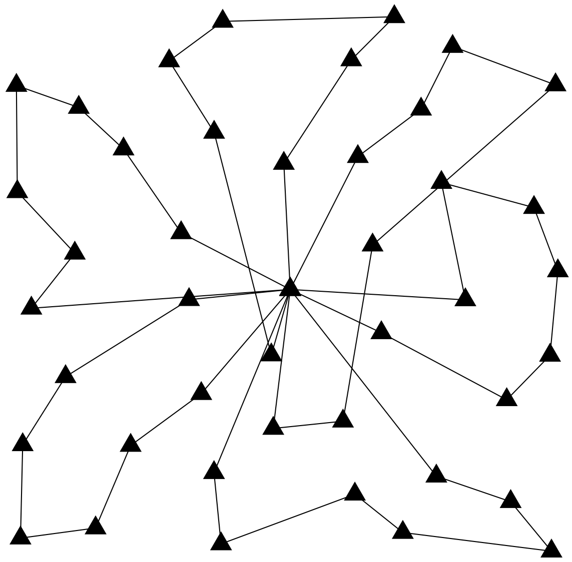
(a) p1-scenario00-HITL



(b) p1-scenario01-HITL

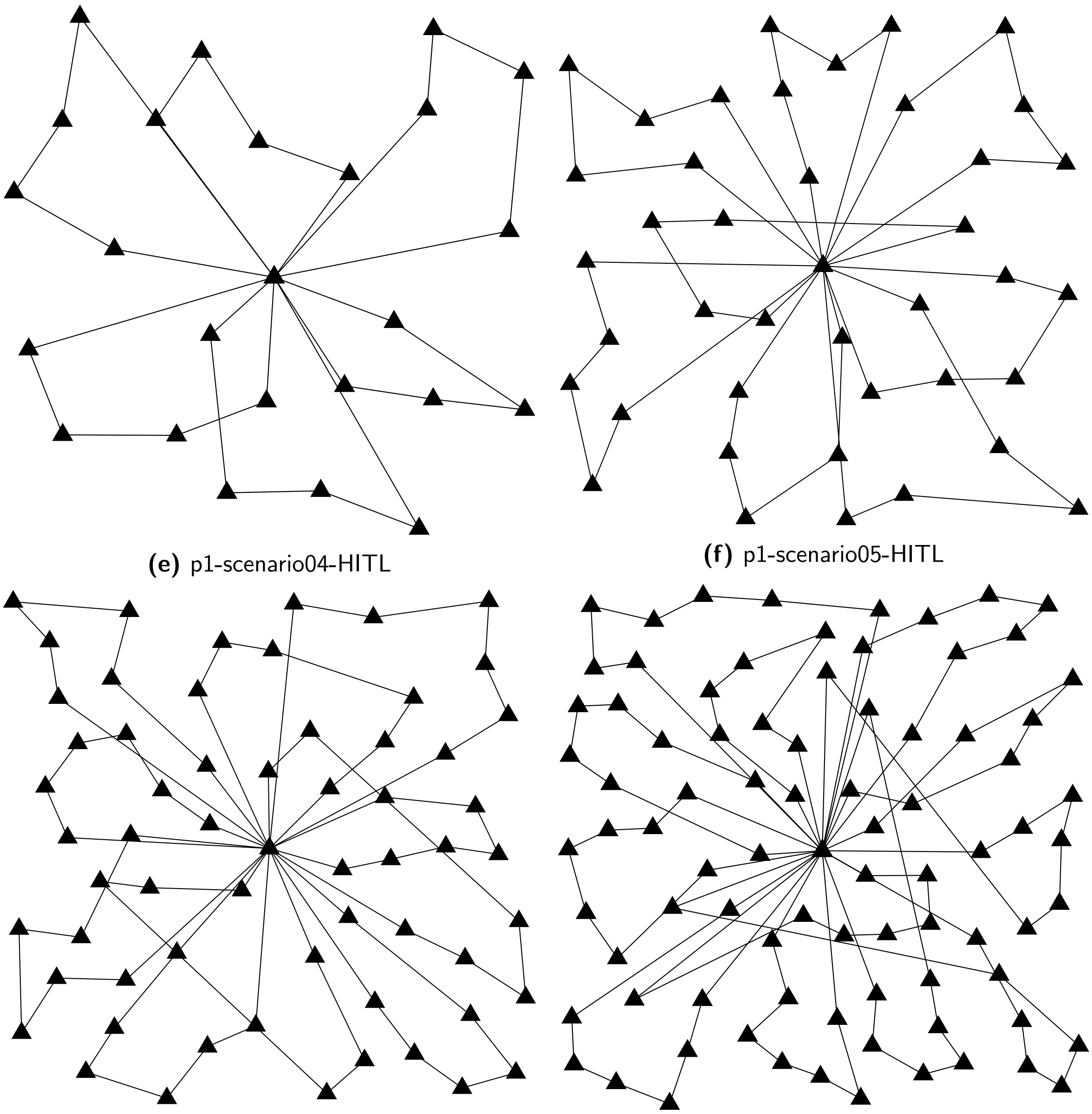


(c) p1-scenario02-HITL



(d) p1-scenario03-HITL

Figure H-2: Participant 1 solutions



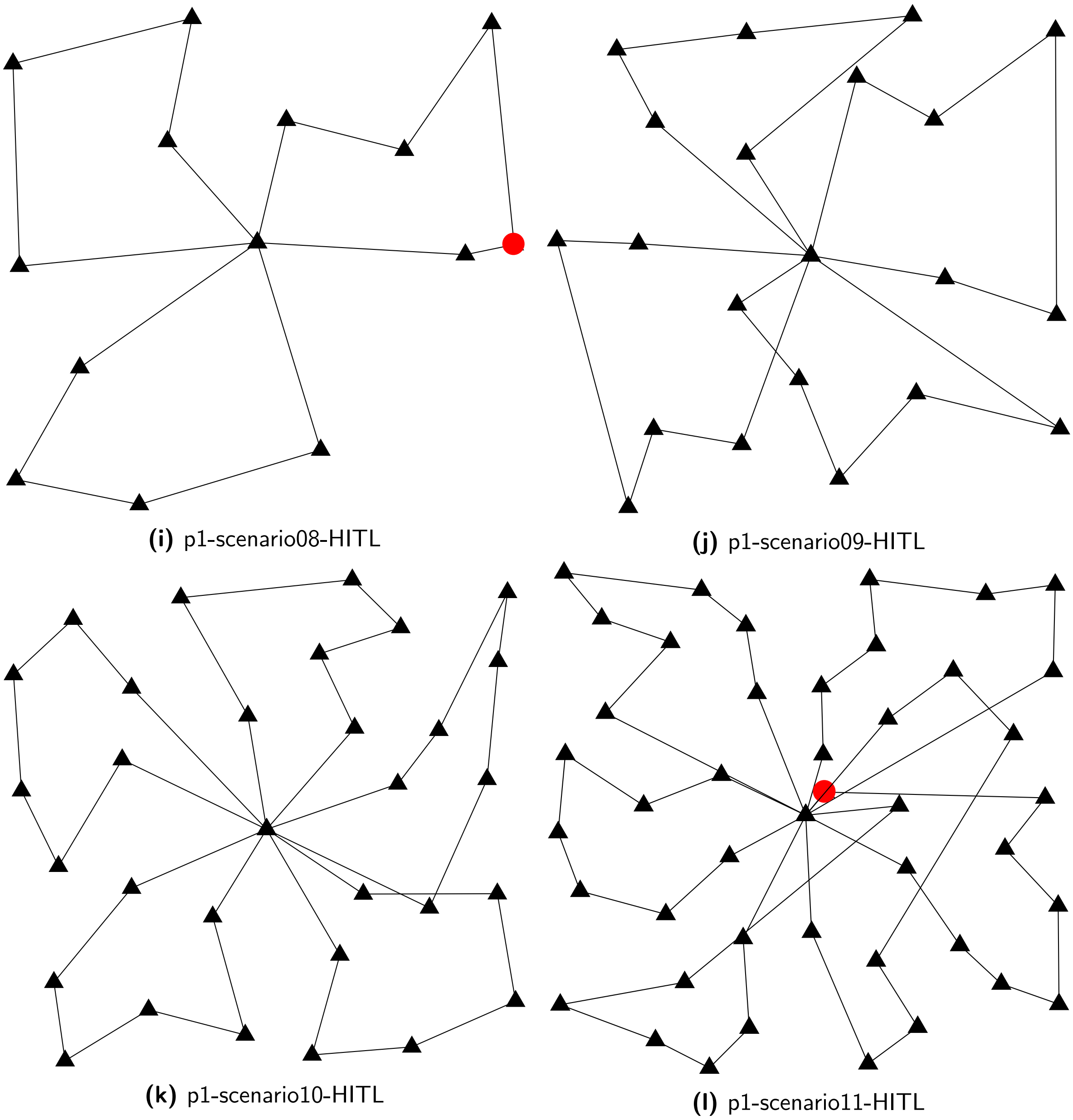
(e) p1-scenario04-HITL

(f) p1-scenario05-HITL

(g) p1-scenario06-HITL

(h) p1-scenario07-HITL

Figure H-2: Participant 1 solutions (continued)



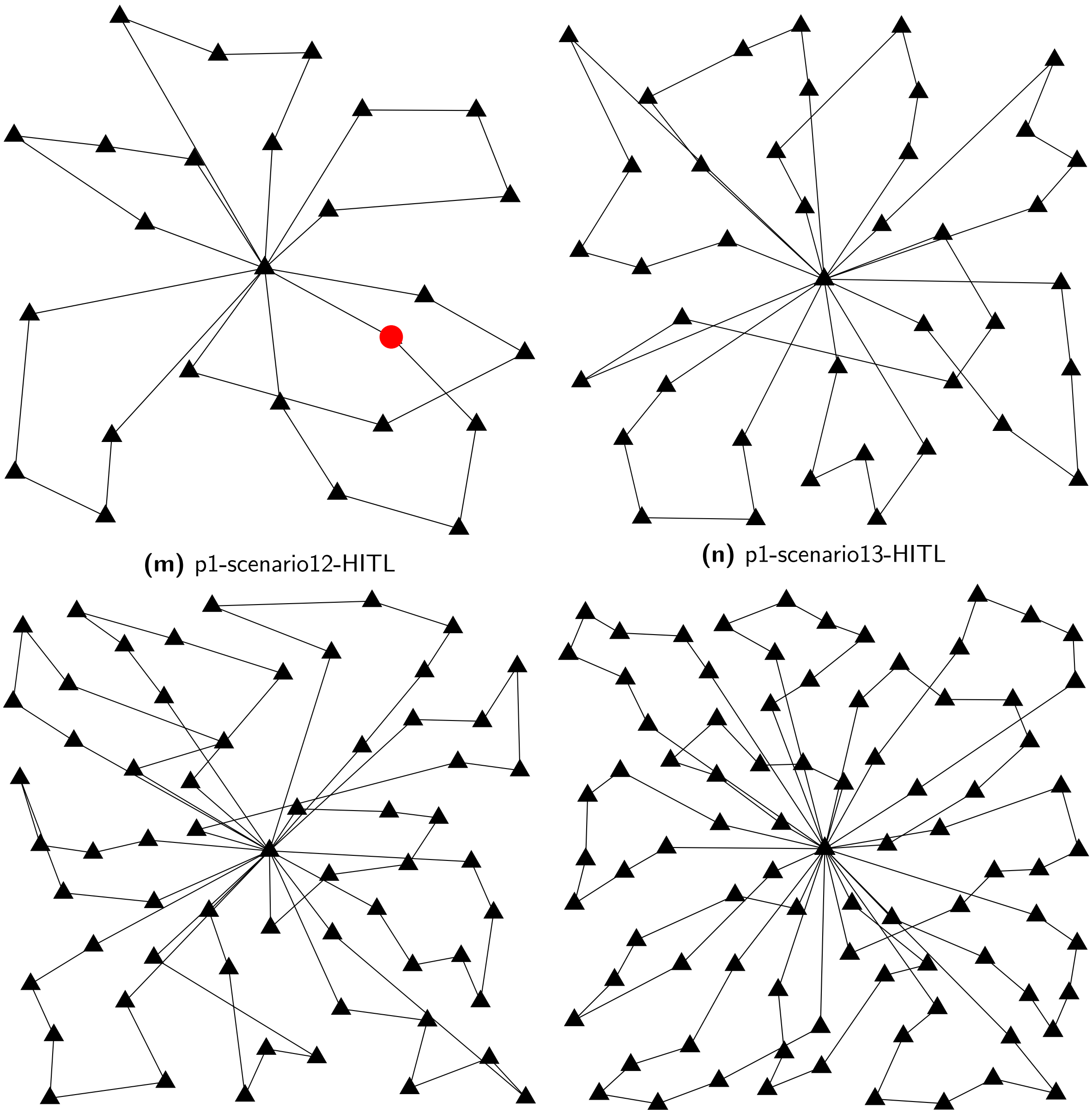
(i) p1-scenario08-HITL

(j) p1-scenario09-HITL

(k) p1-scenario10-HITL

(l) p1-scenario11-HITL

Figure H-2: Participant 1 solutions (continued)



(m) p1-scenario12-HITL

(n) p1-scenario13-HITL

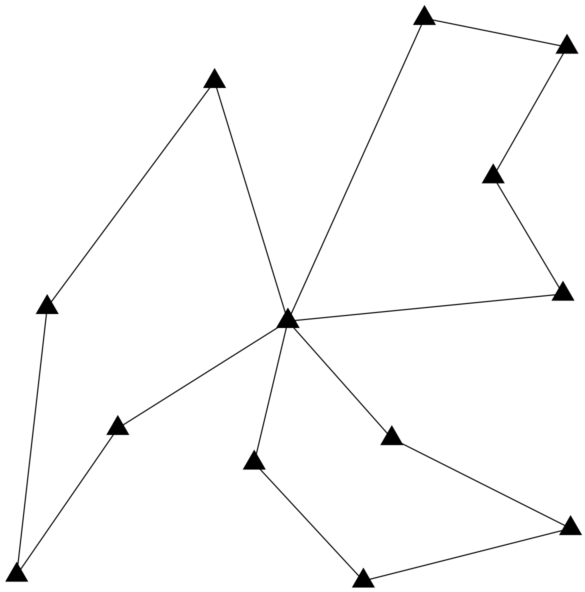
(o) p1-scenario14-HITL

(p) p1-scenario15-HITL

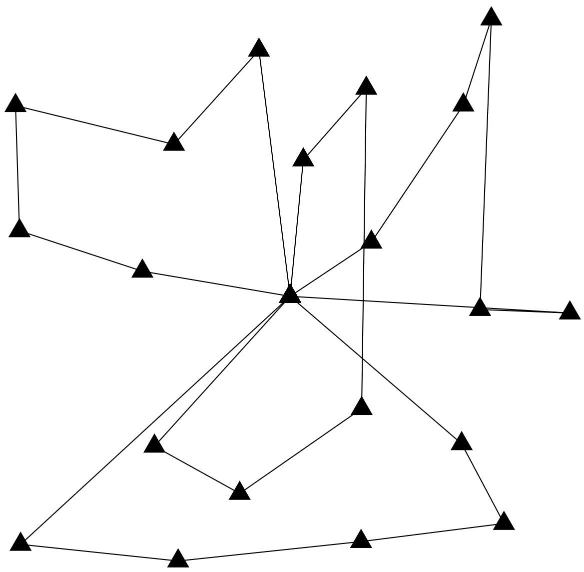
Figure H-2: Participant 1 solutions (continued)

H-3-2 Participant 2

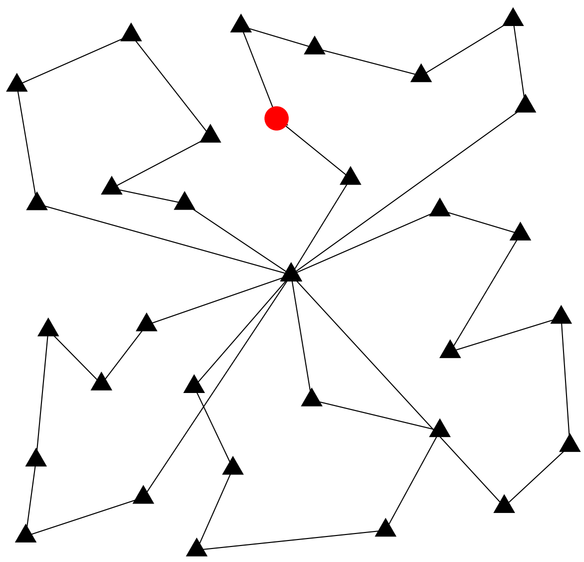
Participant 2 solutions.



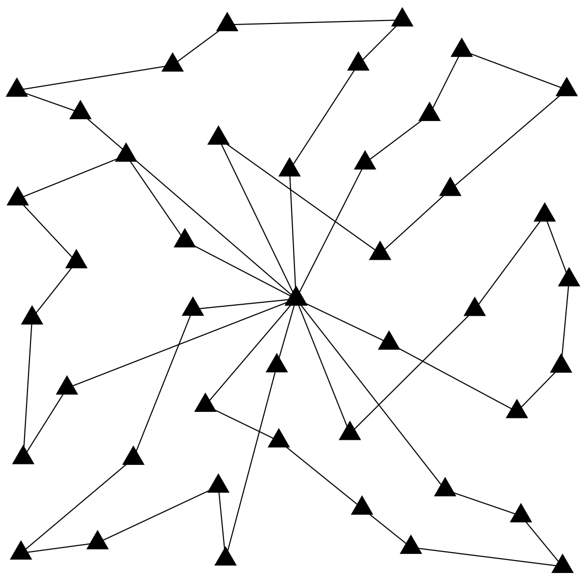
(a) p2-scenario00-HITL



(b) p2-scenario01-HITL

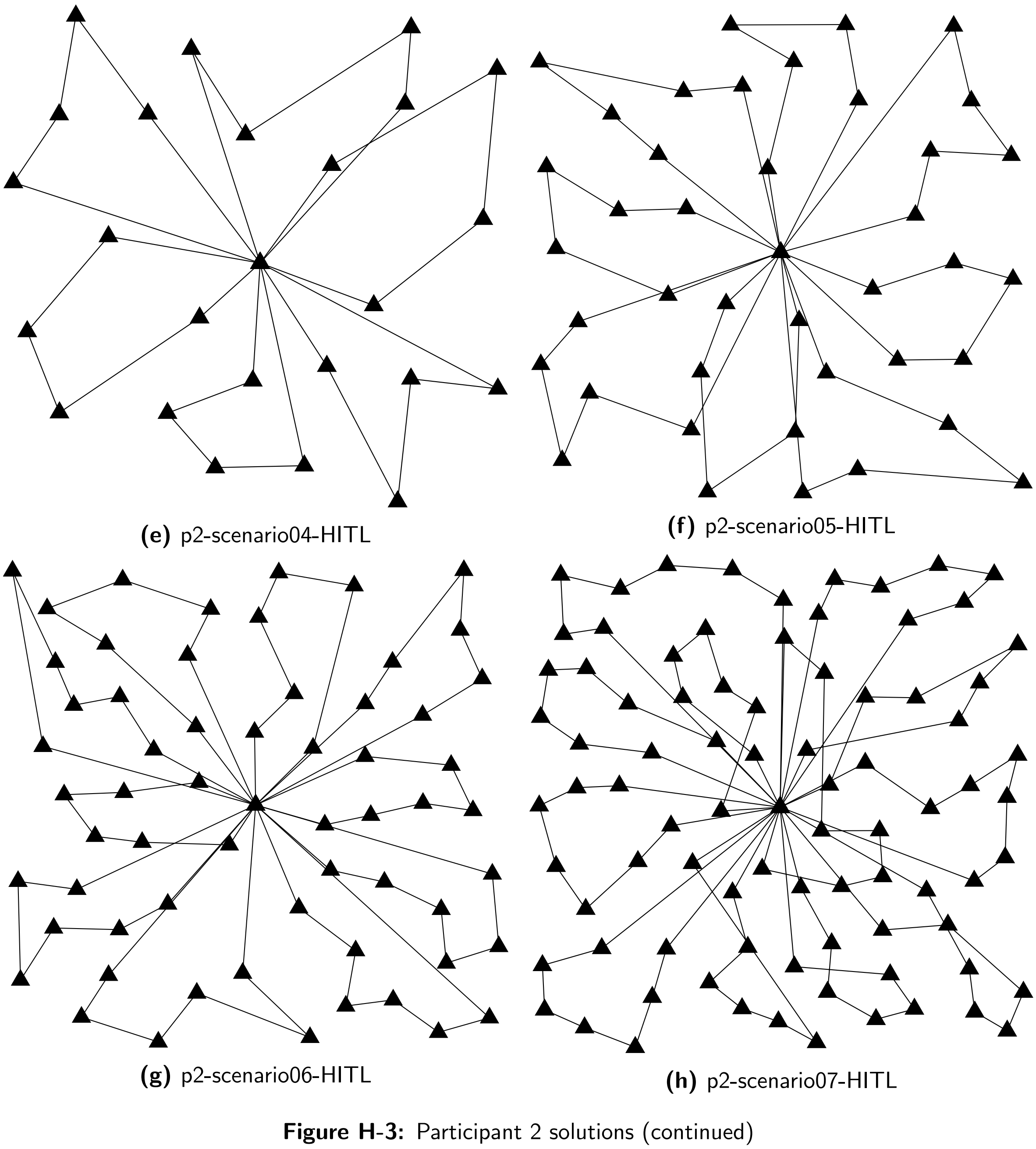


(c) p2-scenario02-HITL



(d) p2-scenario03-HITL

Figure H-3: Participant 2 solutions



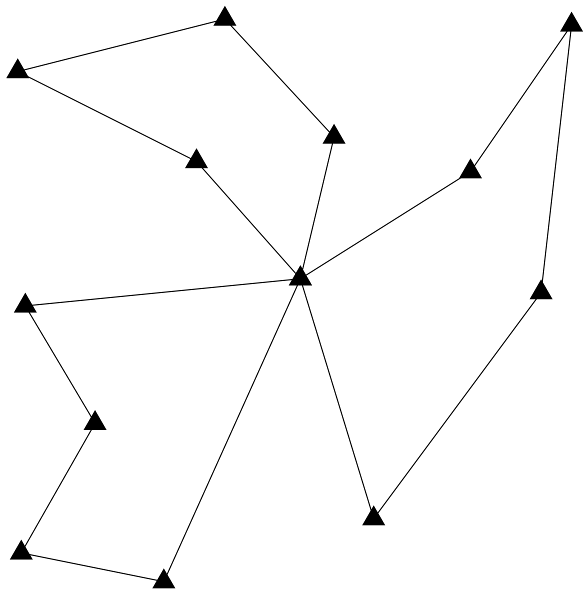
(e) p2-scenario04-HITL

(f) p2-scenario05-HITL

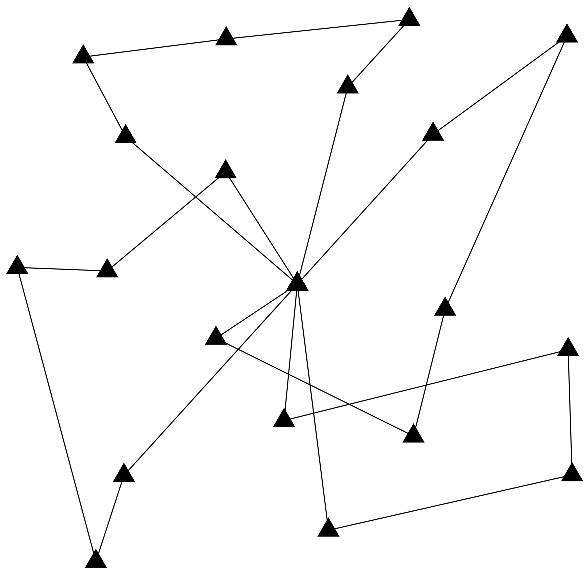
(g) p2-scenario06-HITL

(h) p2-scenario07-HITL

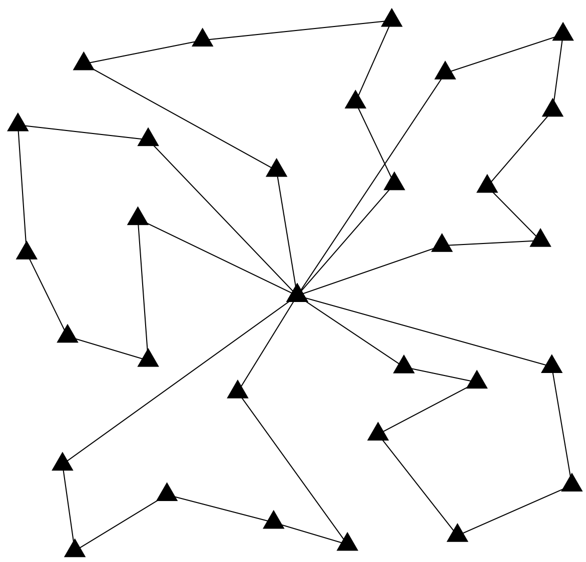
Figure H-3: Participant 2 solutions (continued)



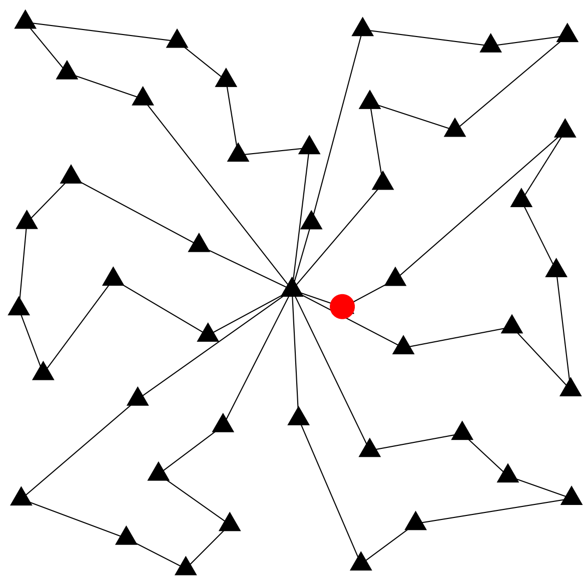
(i) p2-scenario08-HITL



(j) p2-scenario09-HITL

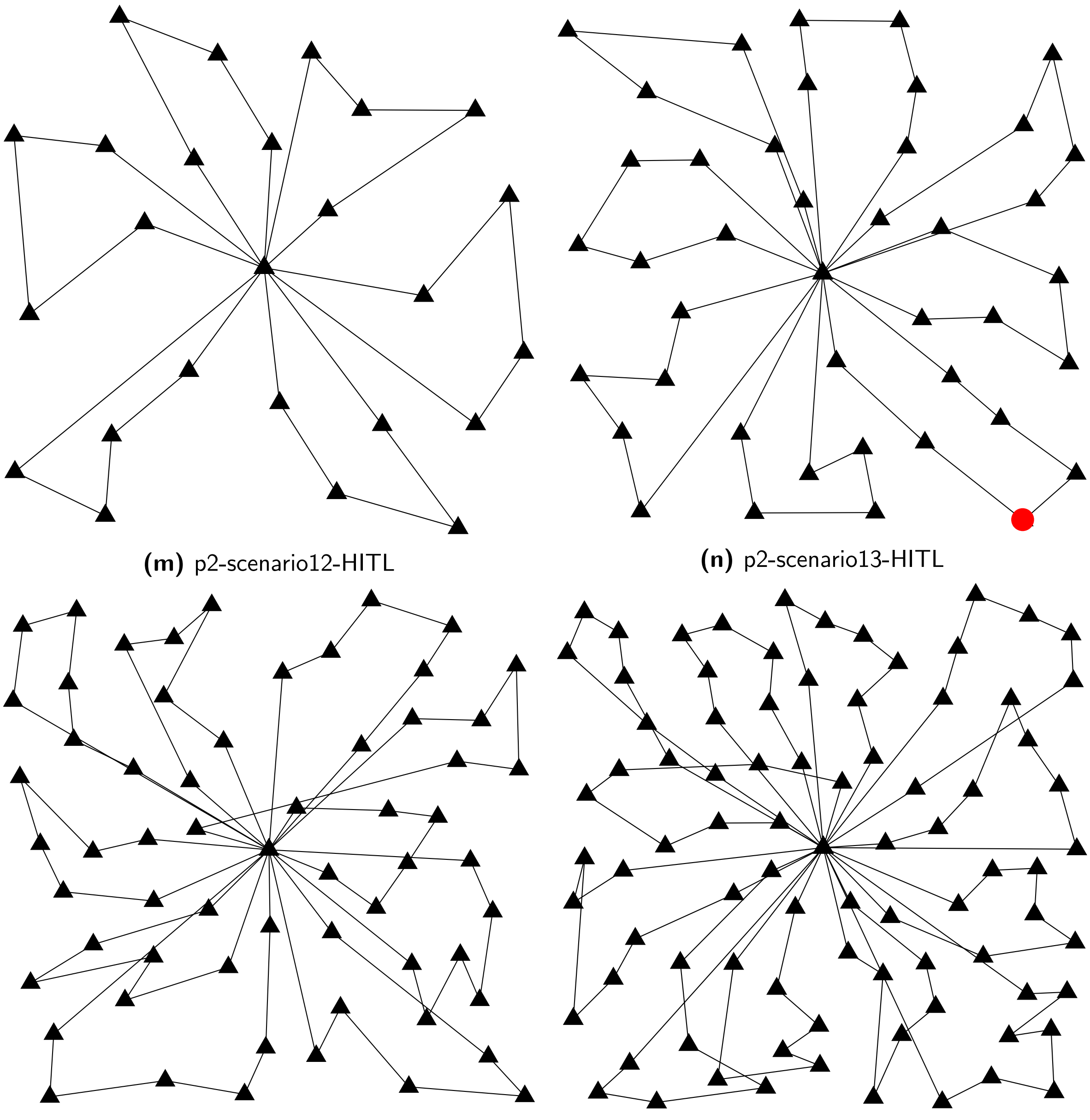


(k) p2-scenario10-HITL



(l) p2-scenario11-HITL

Figure H-3: Participant 2 solutions (continued)



(m) p2-scenario12-HITL

(n) p2-scenario13-HITL

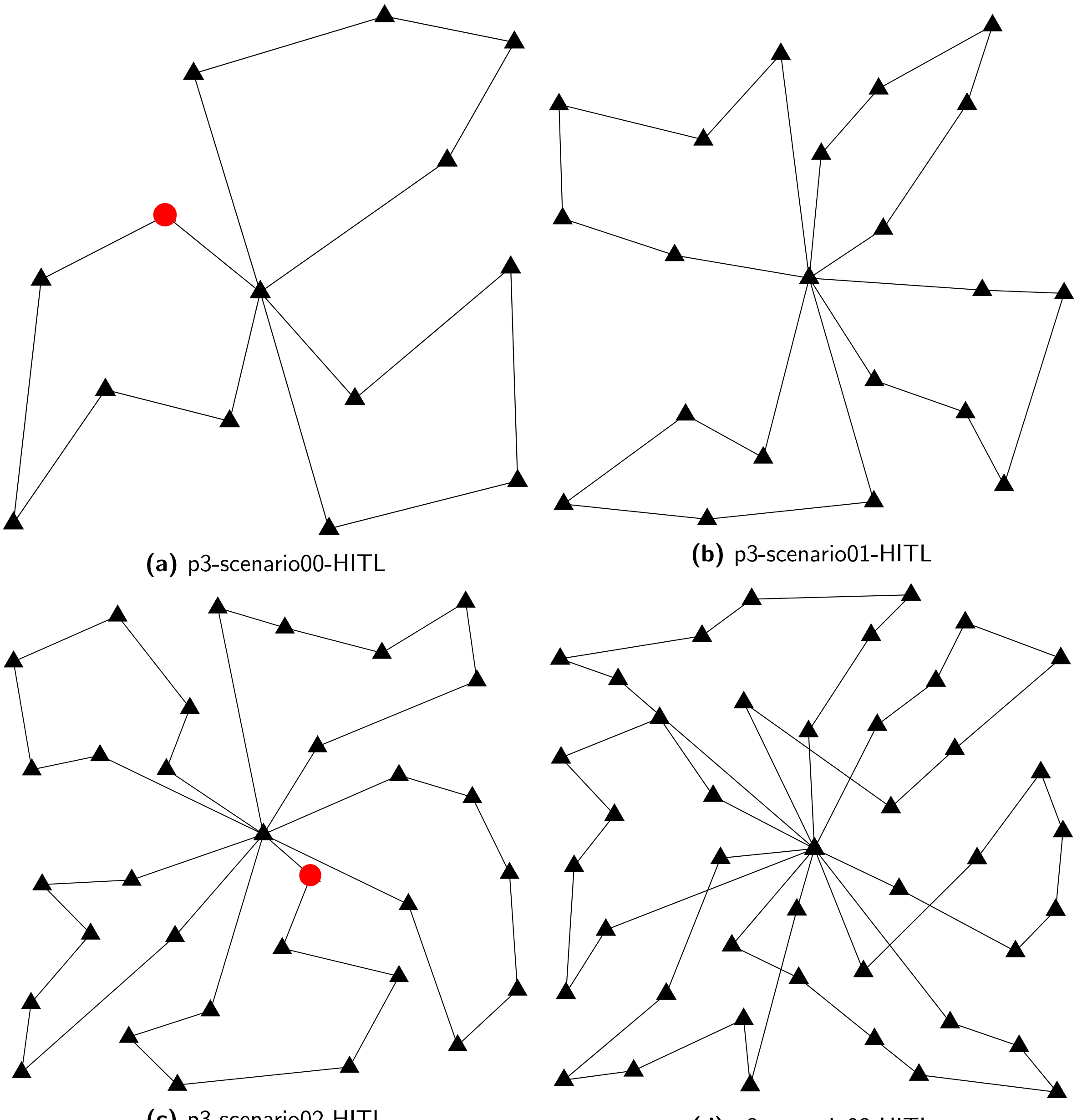
(o) p2-scenario14-HITL

(p) p2-scenario15-HITL

Figure H-3: Participant 2 solutions (continued)

H-3-3 Participant 3

Participant 3 solutions.



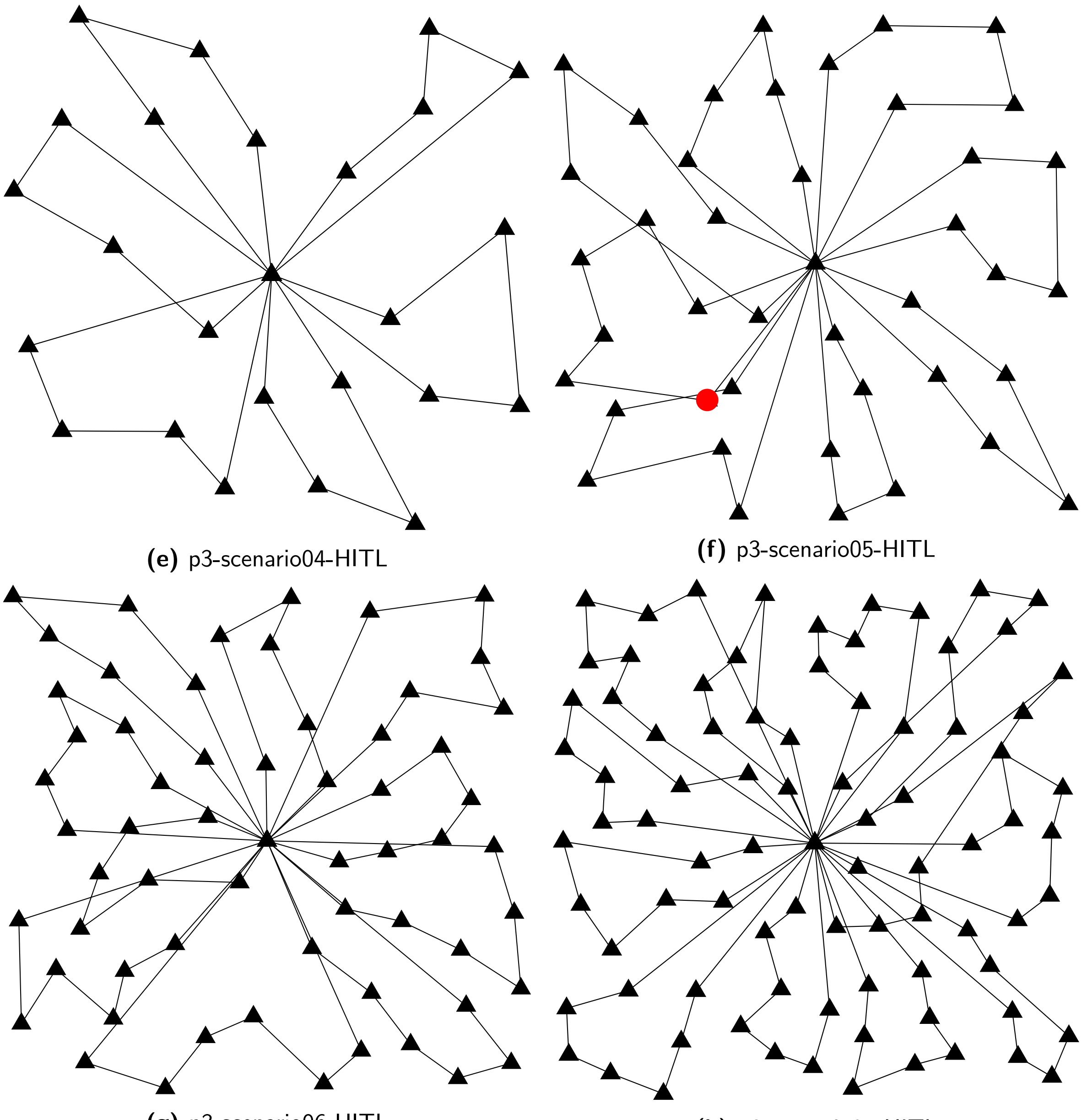
(a) p3-scenario00-HITL

(b) p3-scenario01-HITL

(c) p3-scenario02-HITL

(d) p3-scenario03-HITL

Figure H-4: Participant 3 solutions



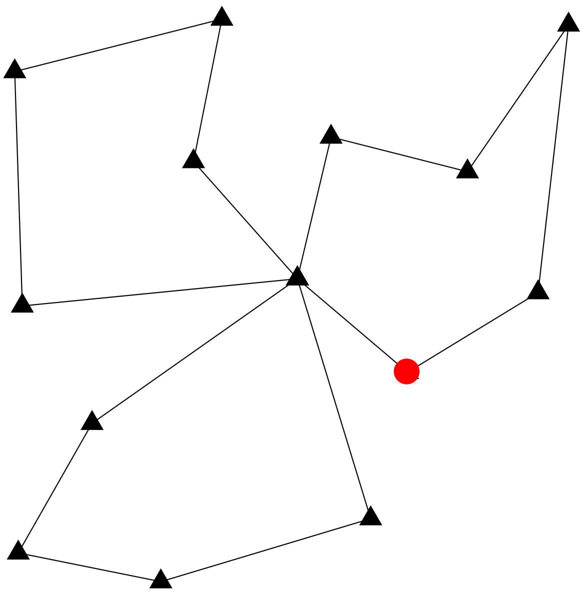
(e) p3-scenario04-HITL

(f) p3-scenario05-HITL

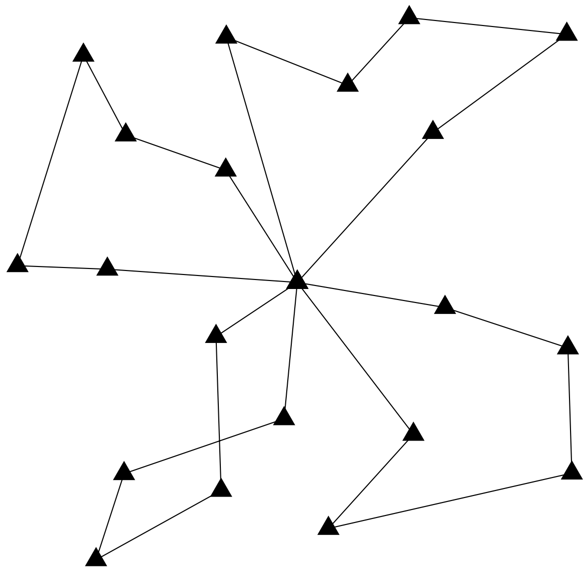
(g) p3-scenario06-HITL

(h) p3-scenario07-HITL

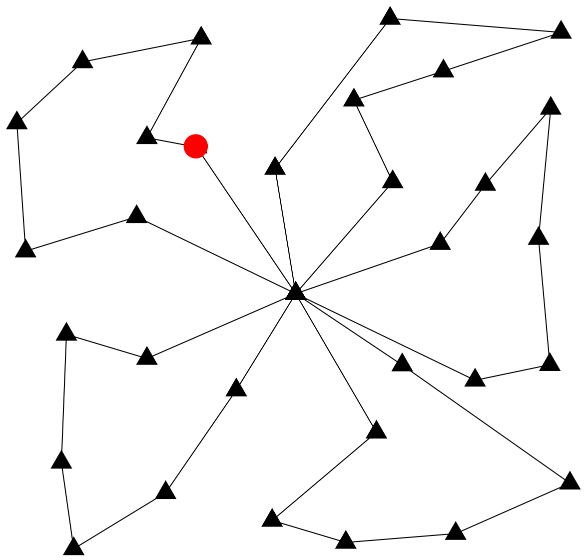
Figure H-4: Participant 3 solutions (continued)



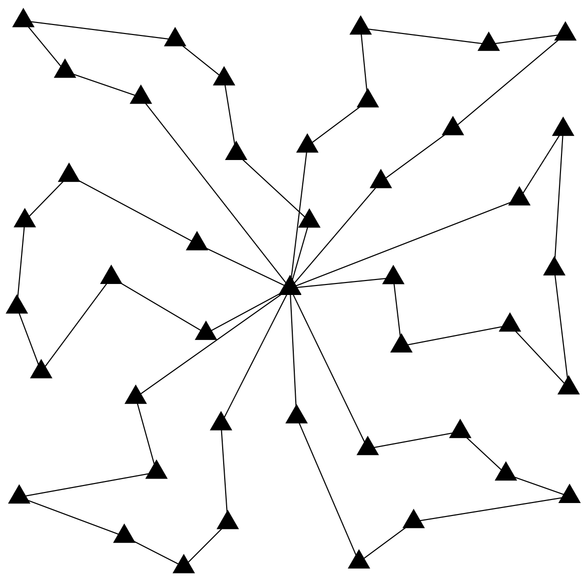
(i) p3-scenario08-HITL



(j) p3-scenario09-HITL

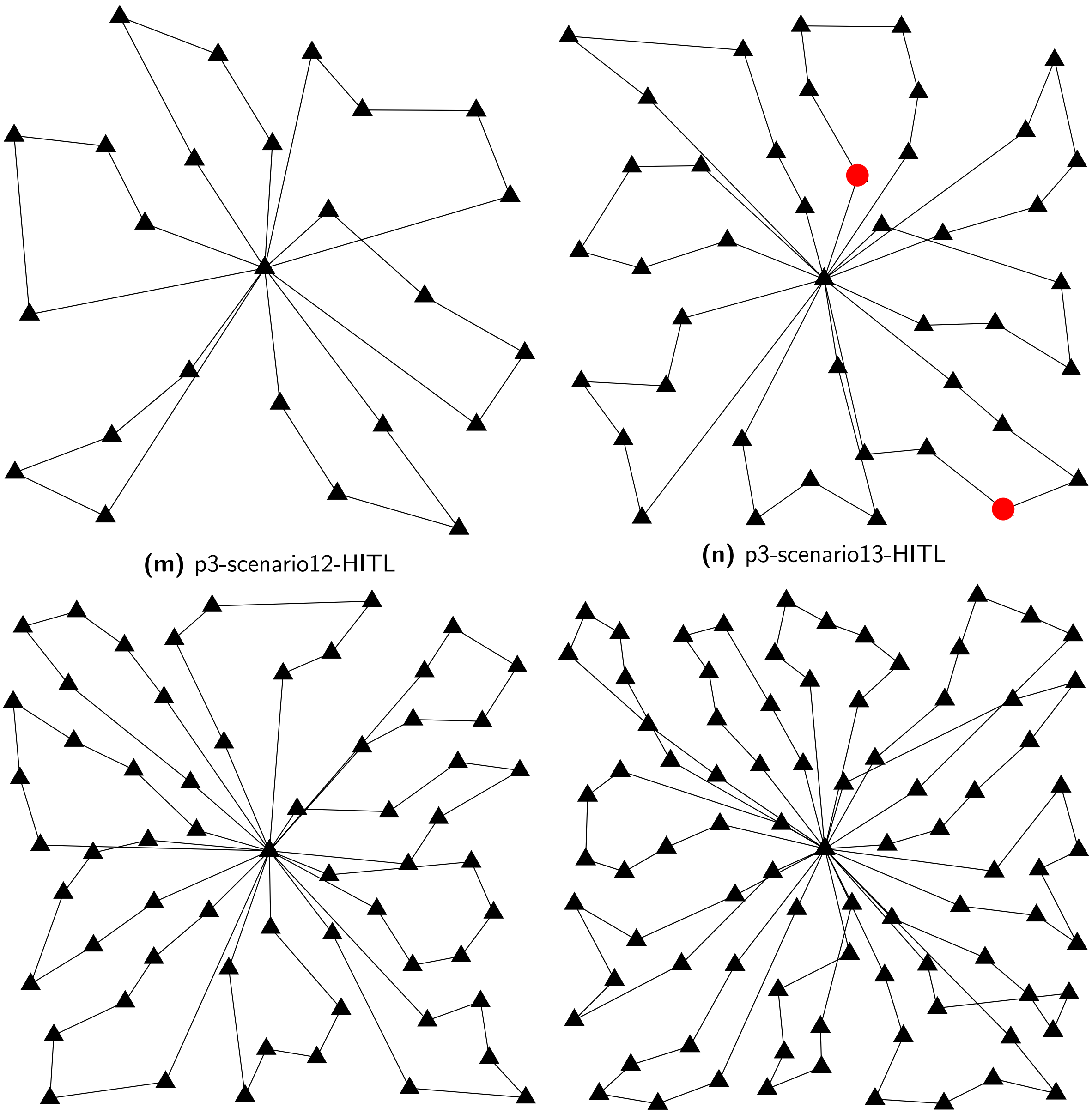


(k) p3-scenario10-HITL



(l) p3-scenario11-HITL

Figure H-4: Participant 3 solutions (continued)



(m) p3-scenario12-HITL

(n) p3-scenario13-HITL

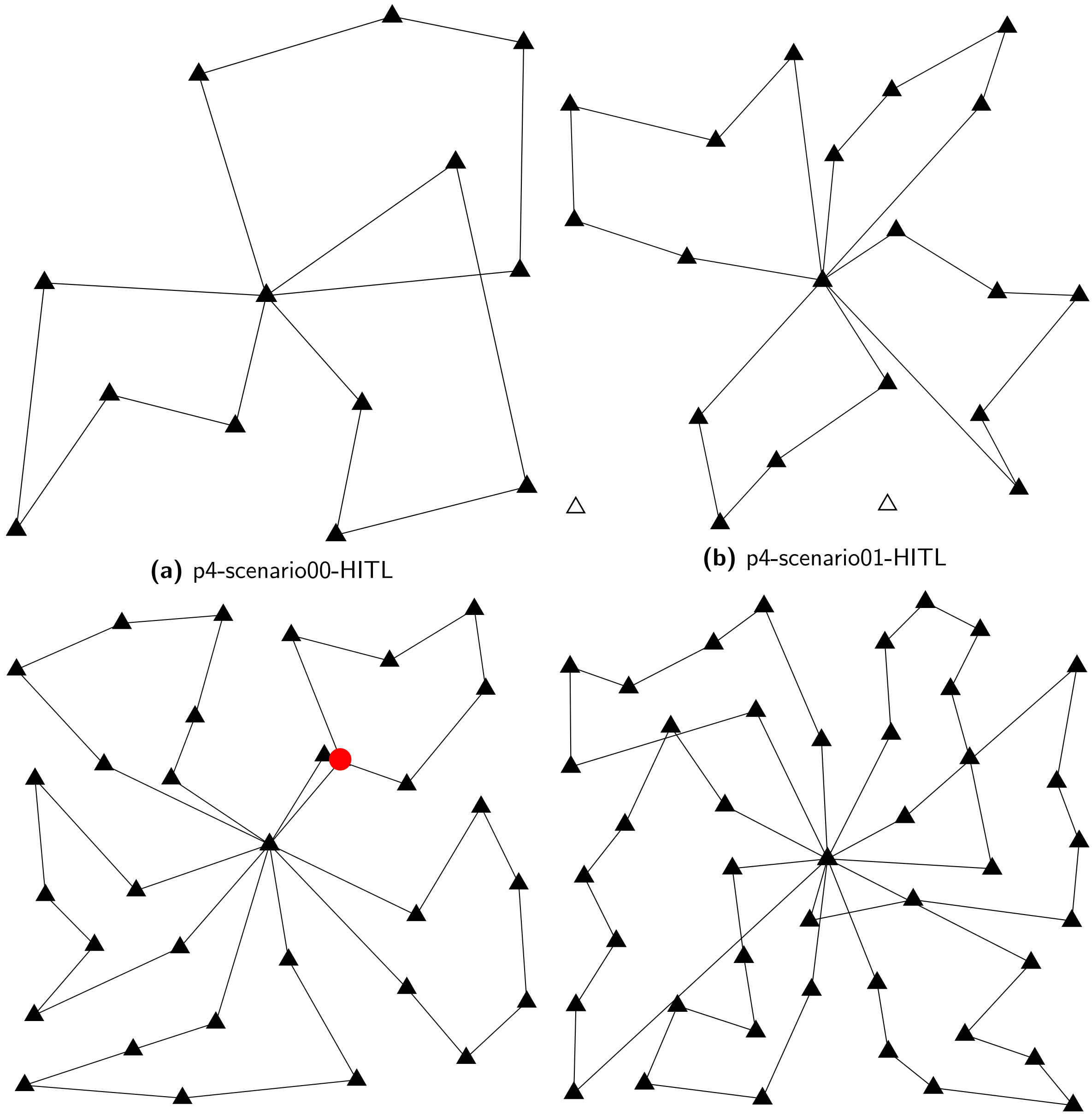
(o) p3-scenario14-HITL

(p) p3-scenario15-HITL

Figure H-4: Participant 3 solutions (continued)

H-3-4 Participant 4

Participant 4 solutions.



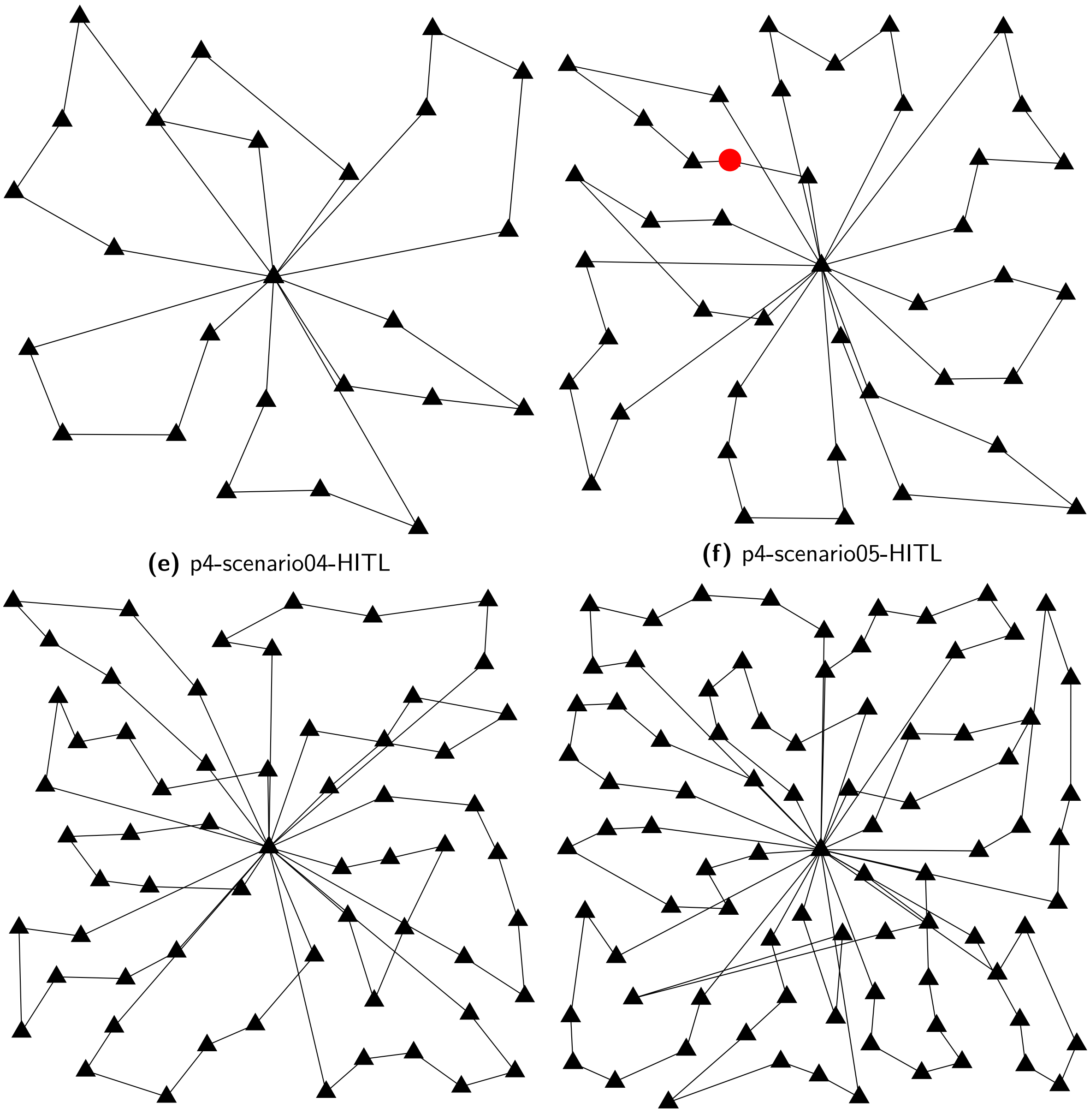
(a) p4-scenario00-HITL

(b) p4-scenario01-HITL

(c) p4-scenario02-HITL

(d) p4-scenario03-HITL

Figure H-5: Participant 4 solutions



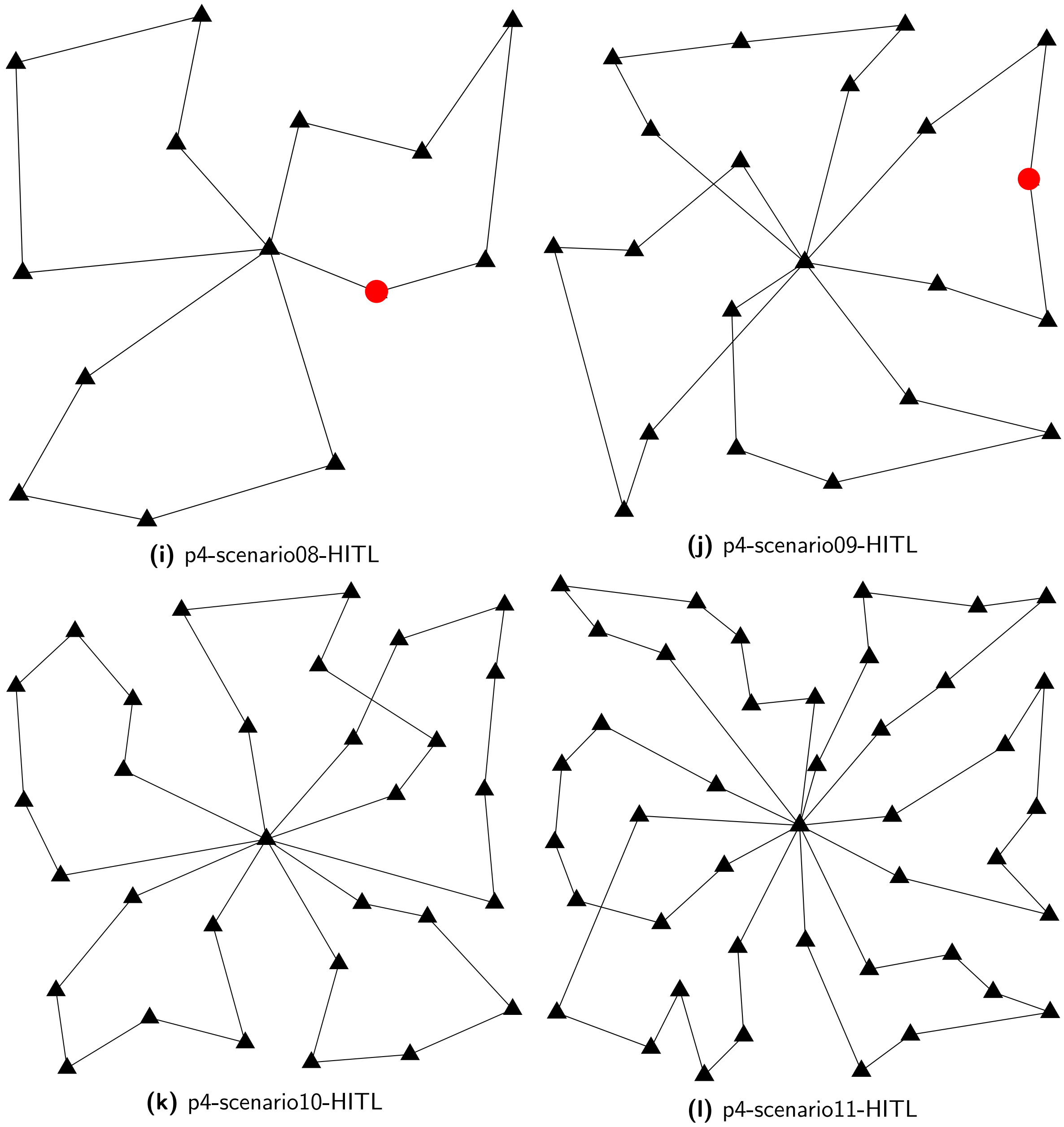
(e) p4-scenario04-HITL

(f) p4-scenario05-HITL

(g) p4-scenario06-HITL

(h) p4-scenario07-HITL

Figure H-5: Participant 4 solutions (continued)



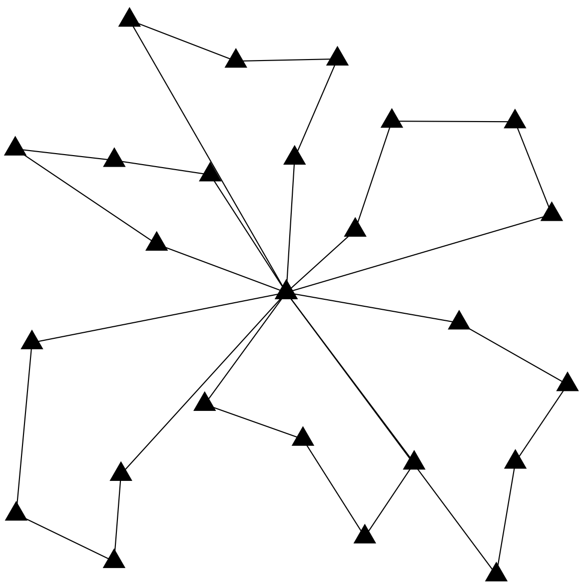
(i) p4-scenario08-HITL

(j) p4-scenario09-HITL

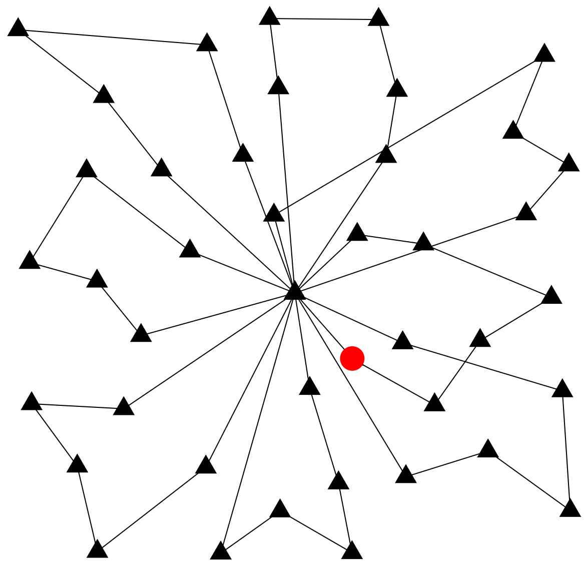
(k) p4-scenario10-HITL

(l) p4-scenario11-HITL

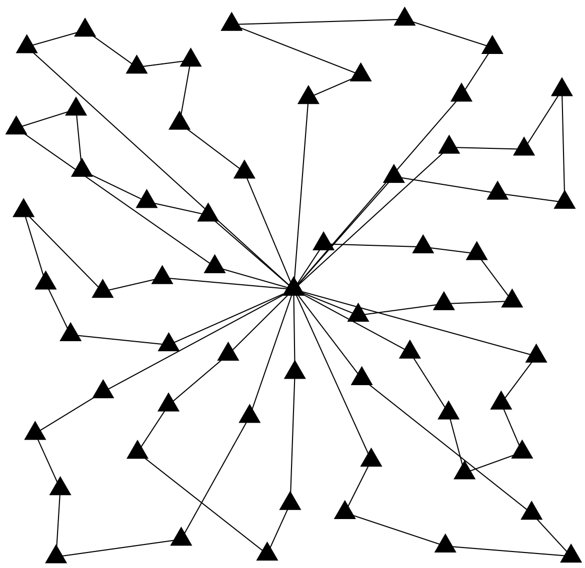
Figure H-5: Participant 4 solutions (continued)



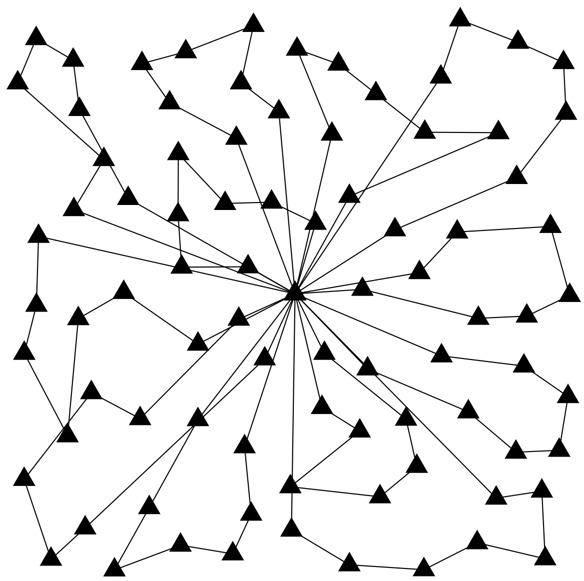
(m) p4-scenario12-HITL



(n) p4-scenario13-HITL



(o) p4-scenario14-HITL

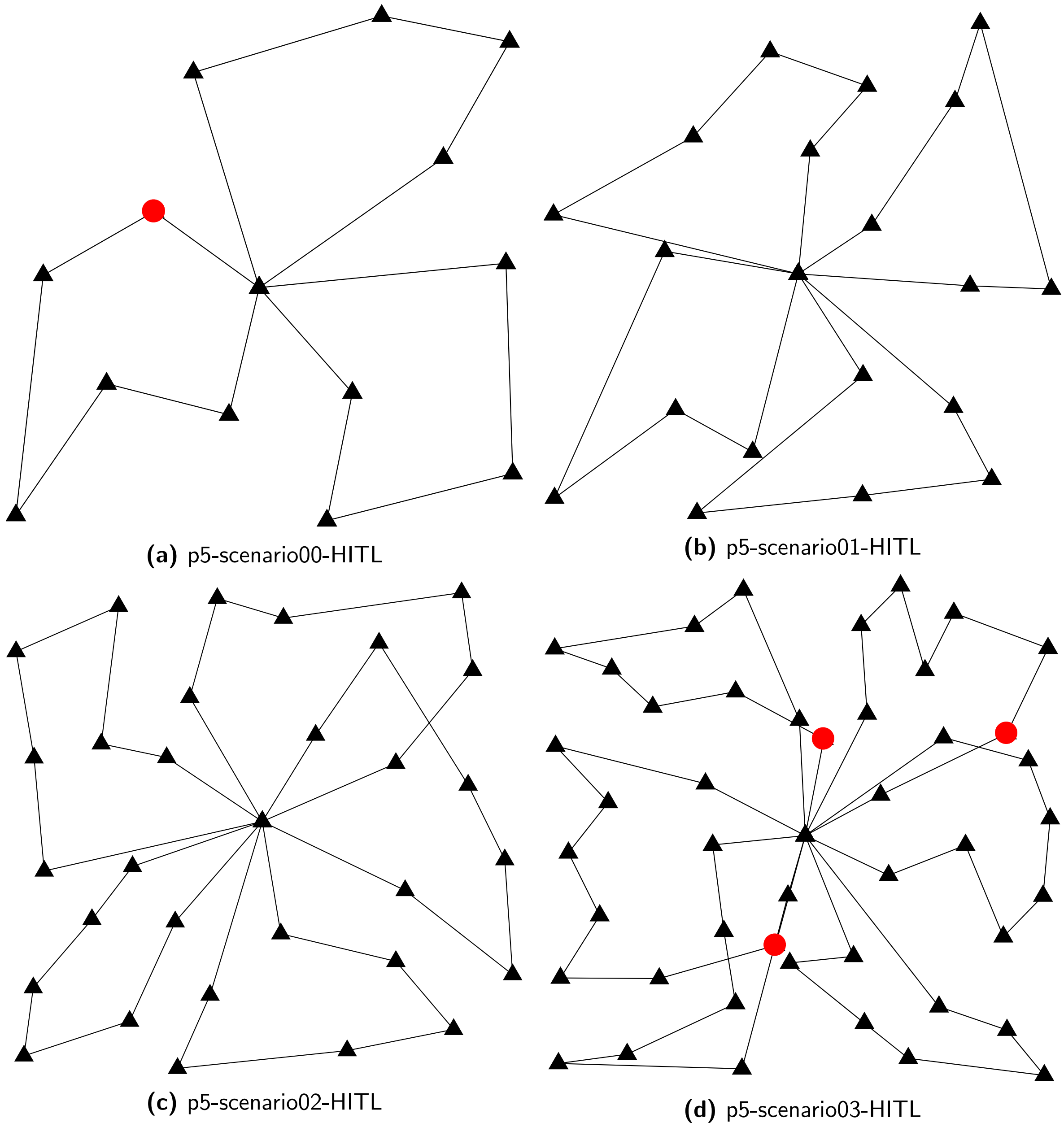


(p) p4-scenario15-HITL

Figure H-5: Participant 4 solutions (continued)

H-3-5 Participant 5

Participant 5 solutions.



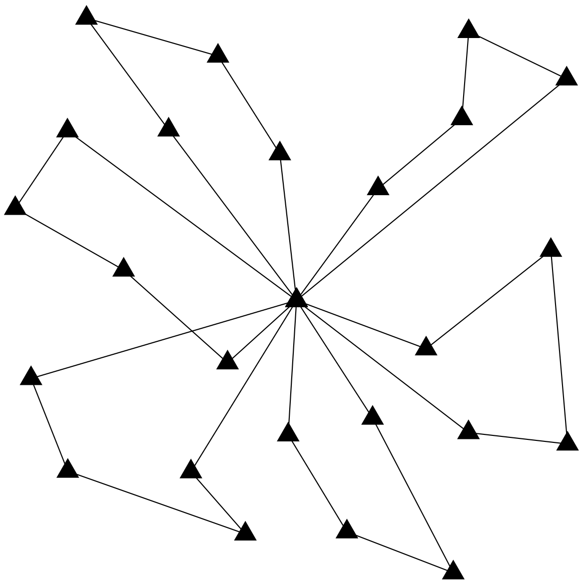
(a) p5-scenario00-HITL

(b) p5-scenario01-HITL

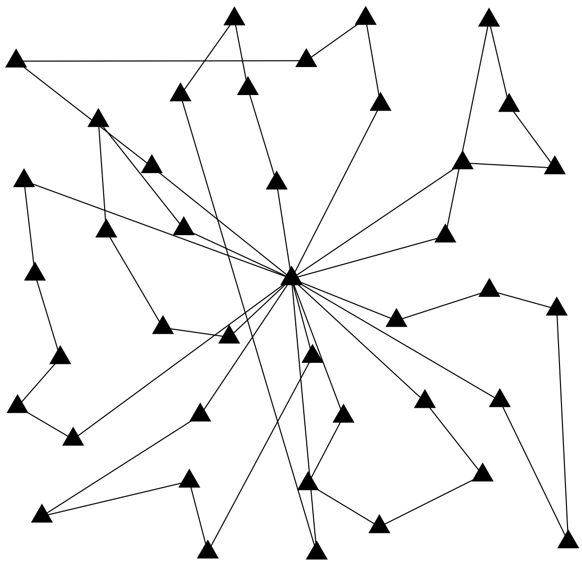
(c) p5-scenario02-HITL

(d) p5-scenario03-HITL

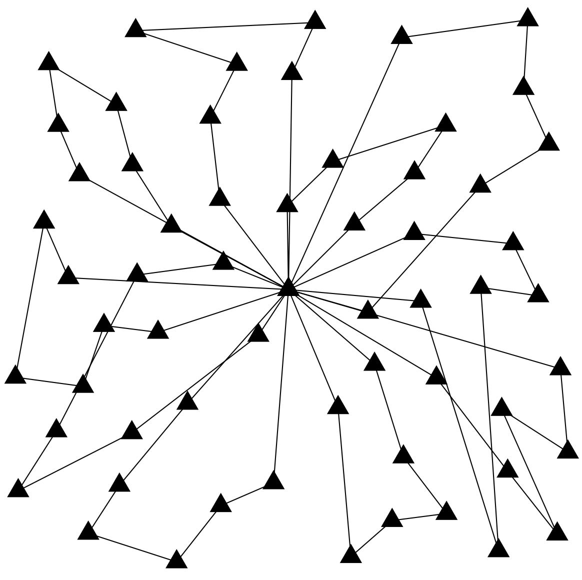
Figure H-6: Participant 5 solutions



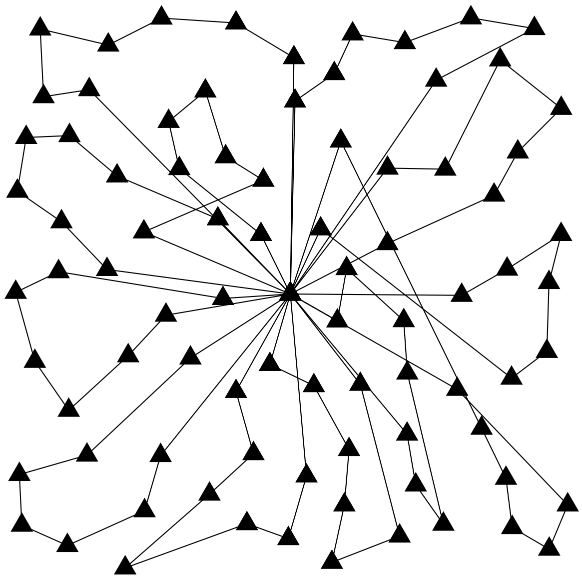
(e) p5-scenario04-HITL



(f) p5-scenario05-HITL

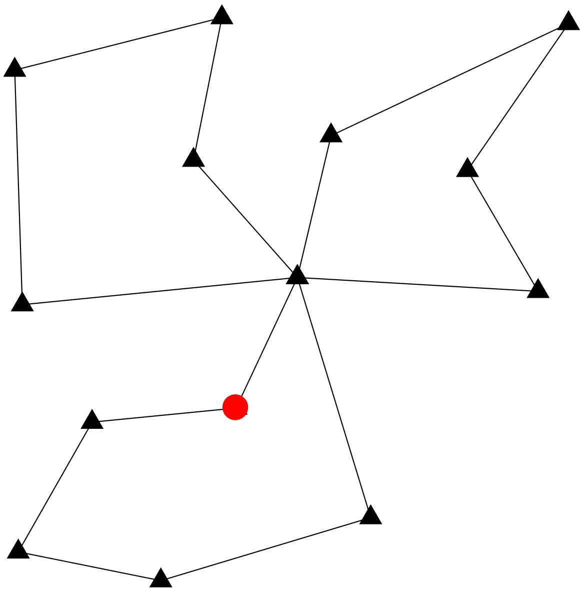


(g) p5-scenario06-HITL

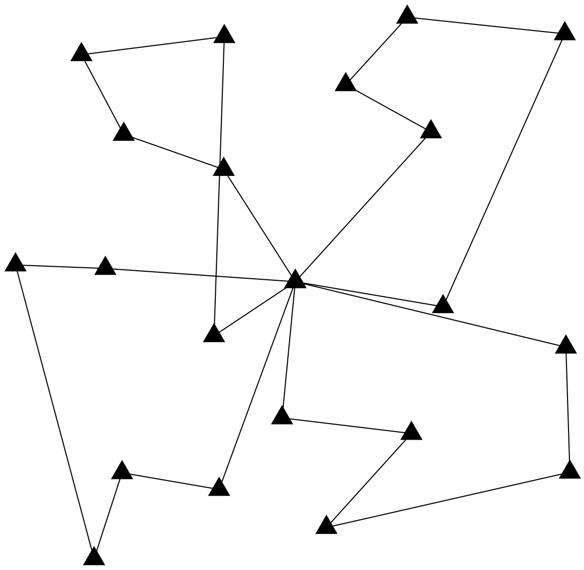


(h) p5-scenario07-HITL

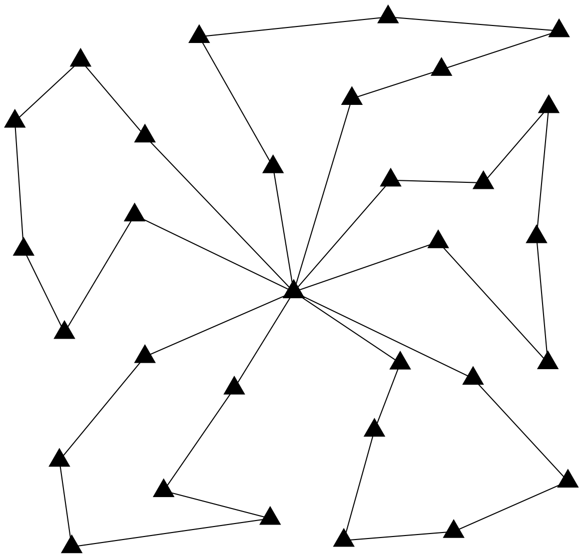
Figure H-6: Participant 5 solutions (continued)



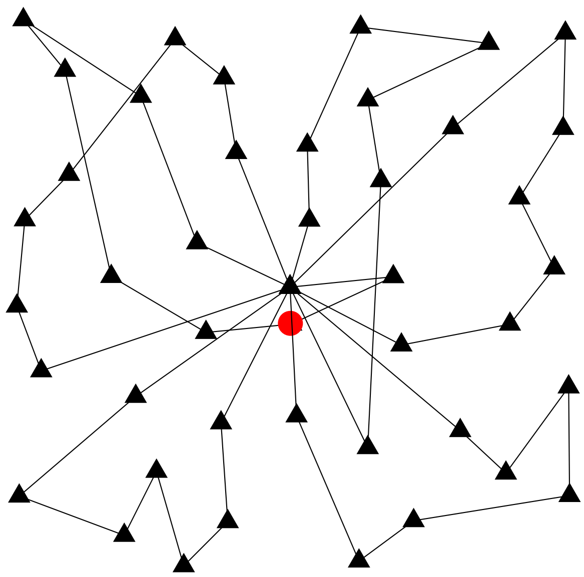
(i) p5-scenario08-HITL



(j) p5-scenario09-HITL

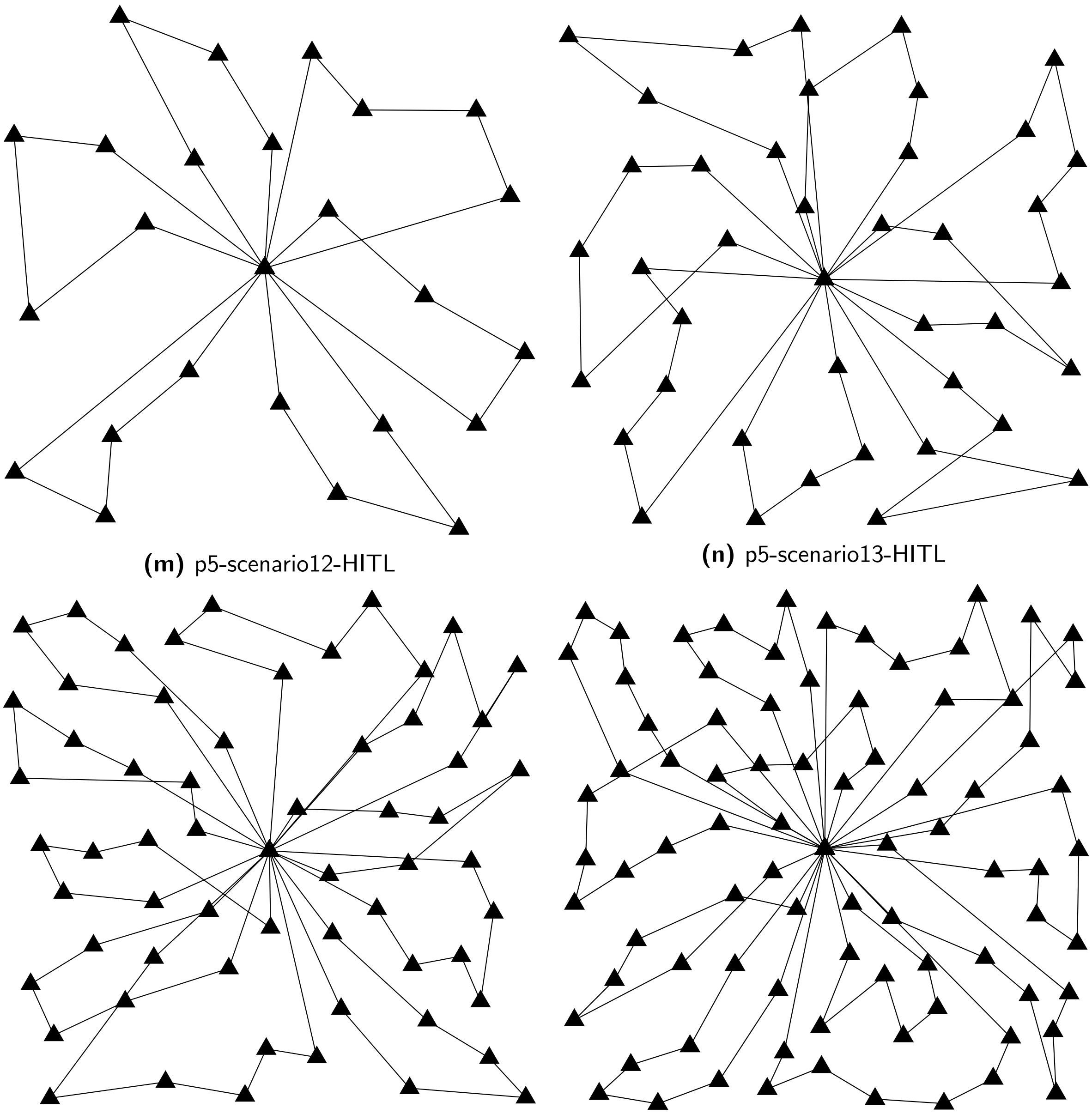


(k) p5-scenario10-HITL



(l) p5-scenario11-HITL

Figure H-6: Participant 5 solutions (continued)



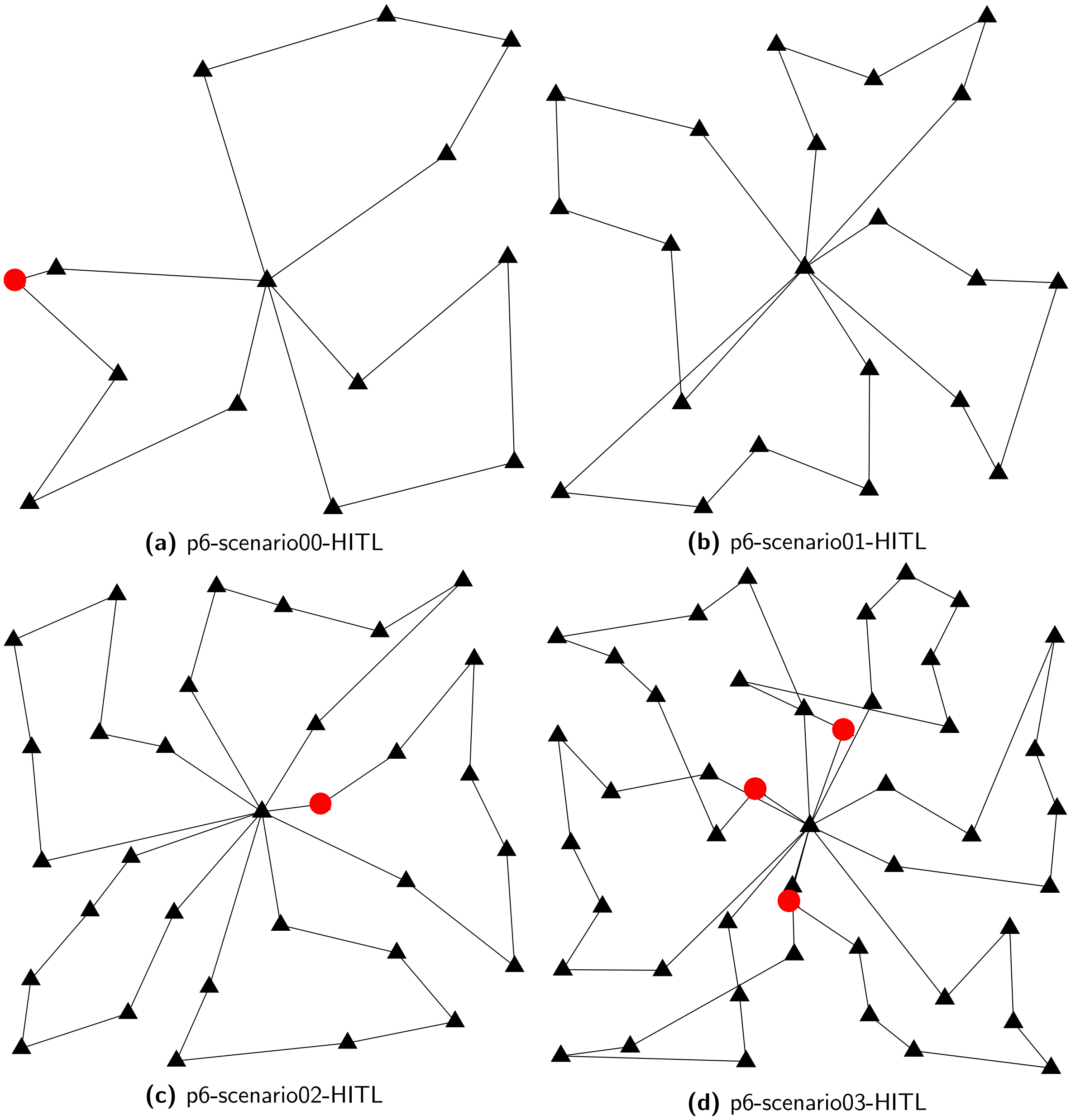
(m) p5-scenario12-HITL **(n)** p5-scenario13-HITL

(o) p5-scenario14-HITL **(p)** p5-scenario15-HITL

Figure H-6: Participant 5 solutions (continued)

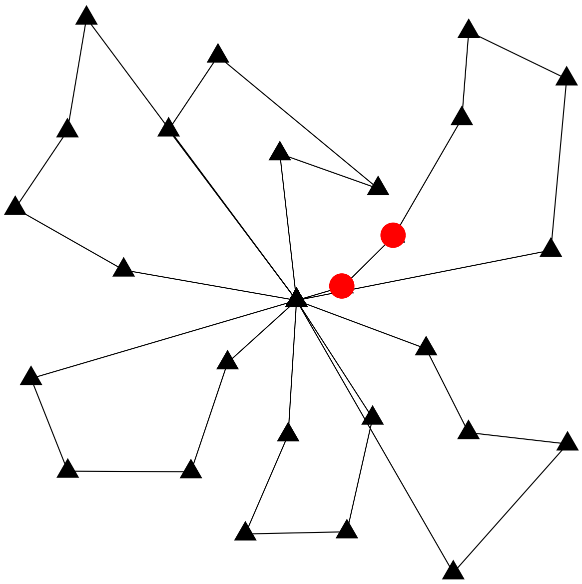
H-3-6 Participant 6

Participant 6 solutions.

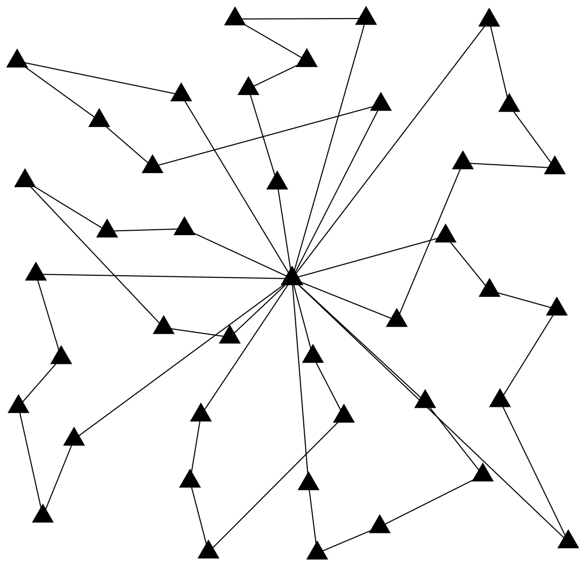


(a) p6-scenario00-HITL **(b)** p6-scenario01-HITL **(c)** p6-scenario02-HITL **(d)** p6-scenario03-HITL

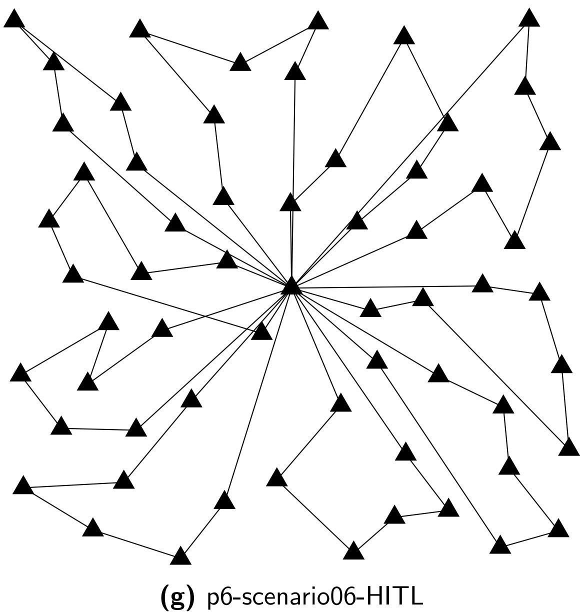
Figure H-7: Participant 6 solutions



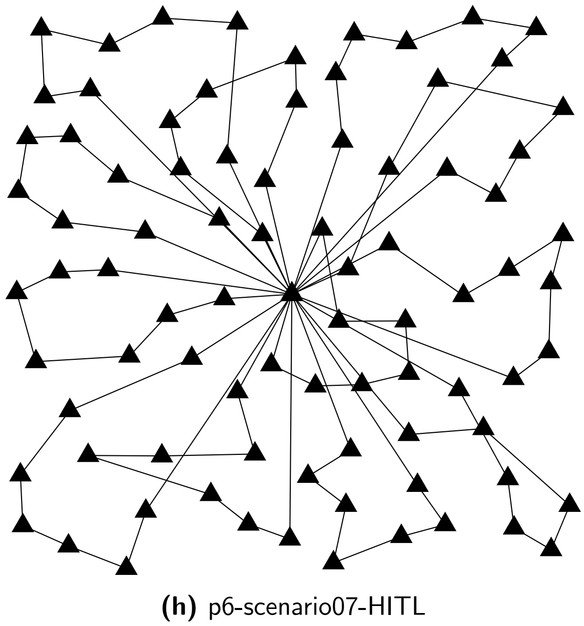
(e) p6-scenario04-HITL



(f) p6-scenario05-HITL

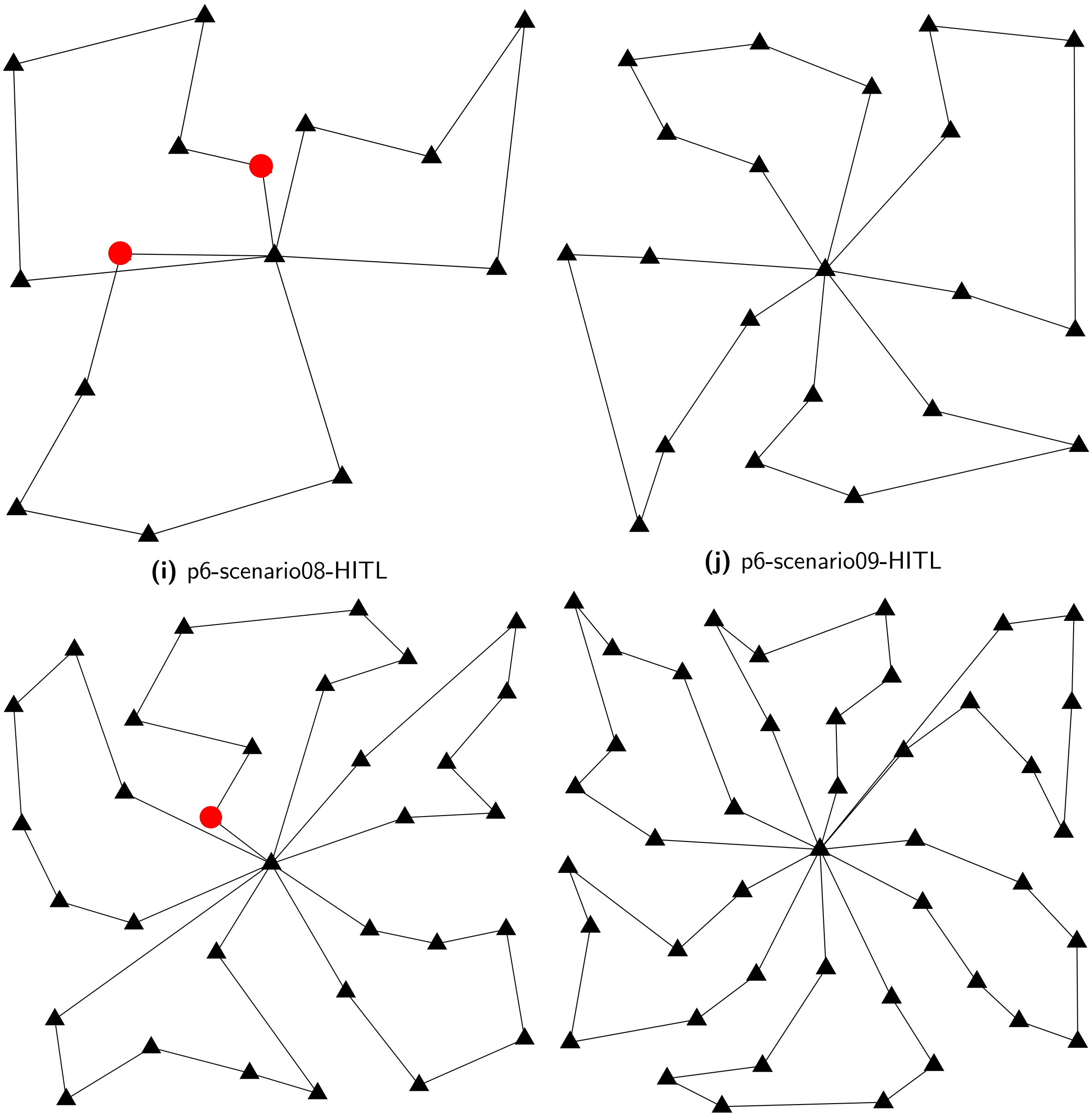


(g) p6-scenario06-HITL



(h) p6-scenario07-HITL

Figure H-7: Participant 6 solutions (continued)



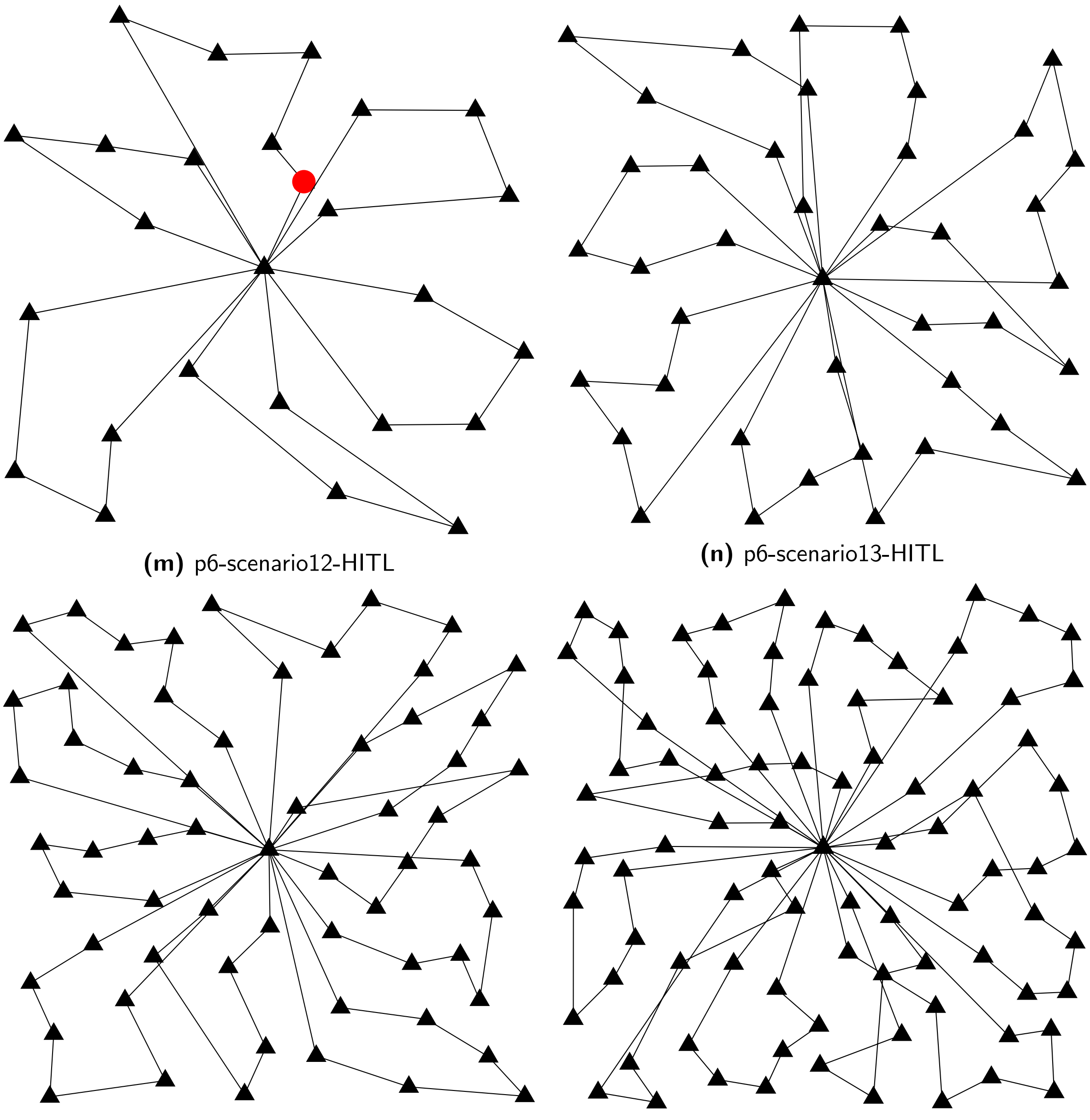
(i) p6-scenario08-HITL

(j) p6-scenario09-HITL

(k) p6-scenario10-HITL

(l) p6-scenario11-HITL

Figure H-7: Participant 6 solutions (continued)



(m) p6-scenario12-HITL

(n) p6-scenario13-HITL

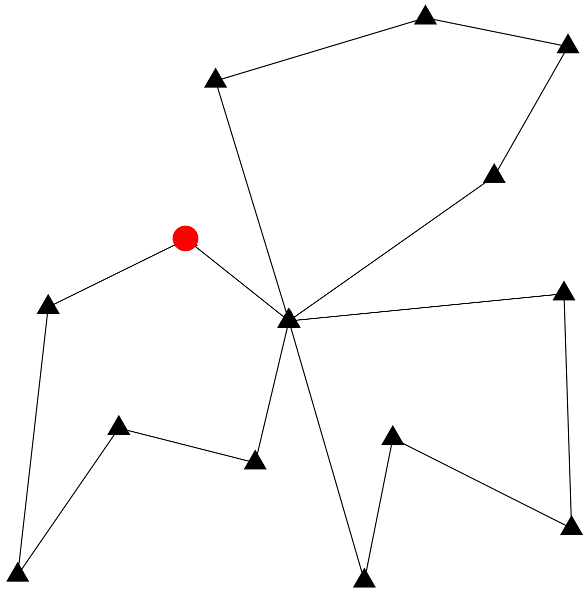
(o) p6-scenario14-HITL

(p) p6-scenario15-HITL

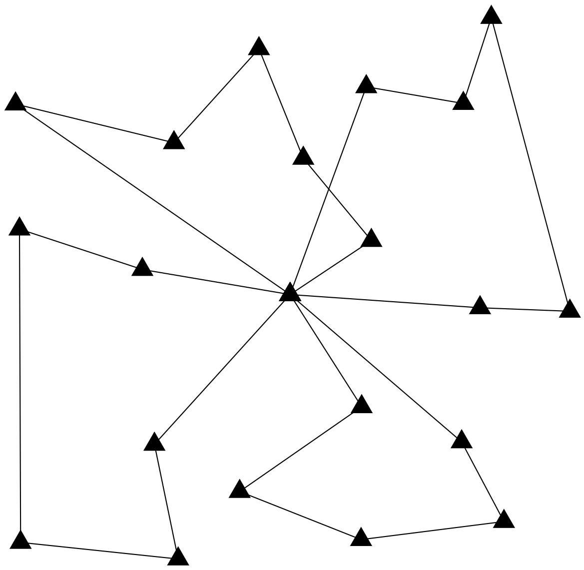
Figure H-7: Participant 6 solutions (continued)

H-3-7 Participant 7

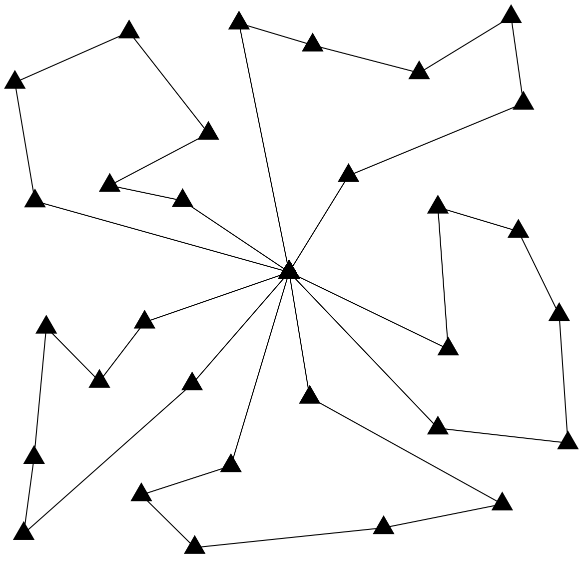
Participant 7 solutions.



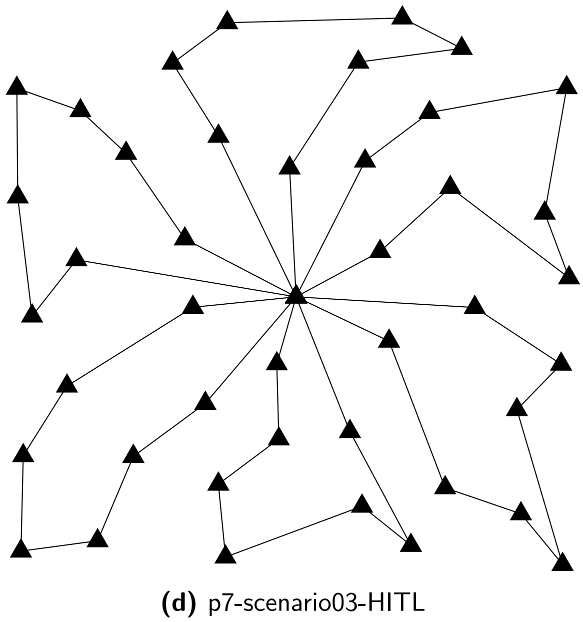
(a) p7-scenario00-HITL



(b) p7-scenario01-HITL

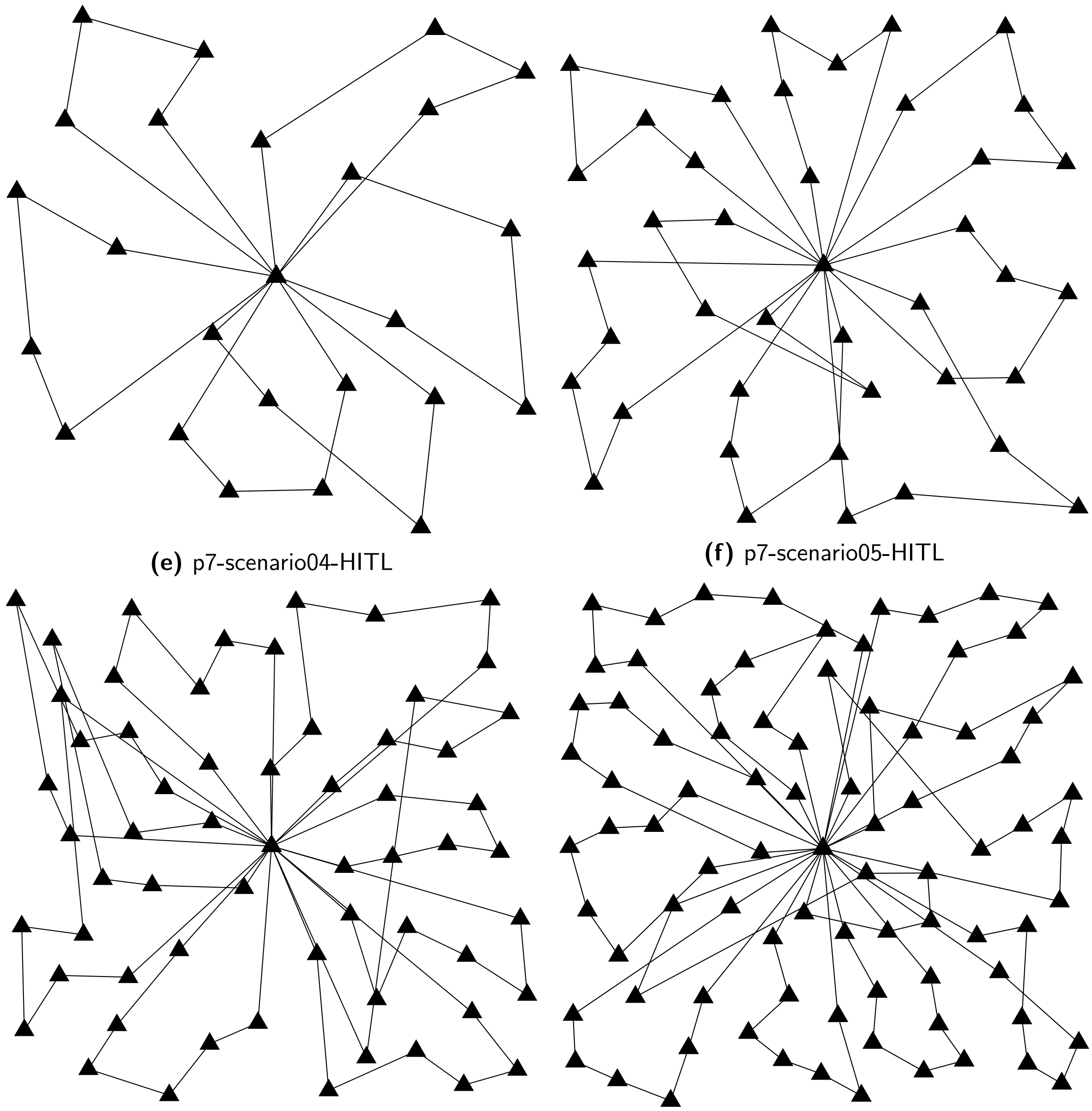


(c) p7-scenario02-HITL



(d) p7-scenario03-HITL

Figure H-8: Participant 7 solutions



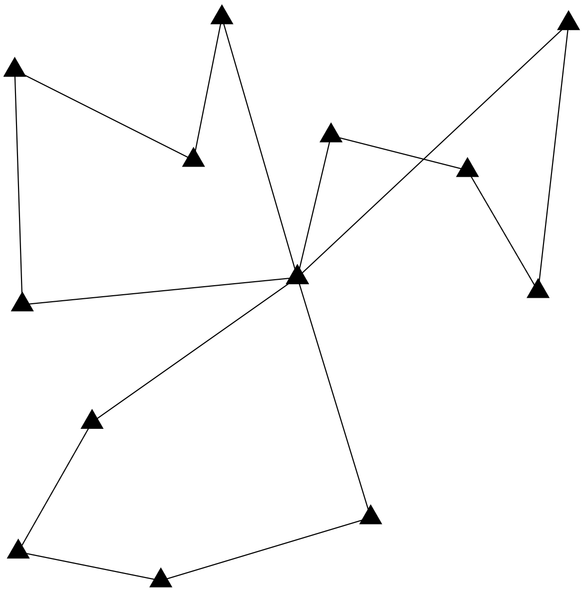
(e) p7-scenario04-HITL

(f) p7-scenario05-HITL

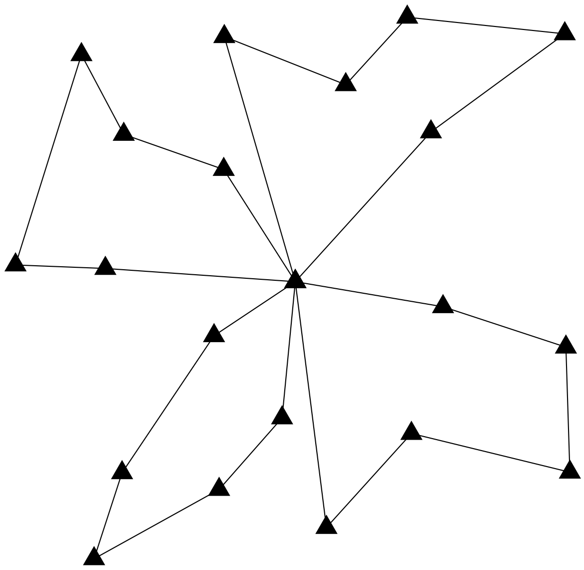
(g) p7-scenario06-HITL

(h) p7-scenario07-HITL

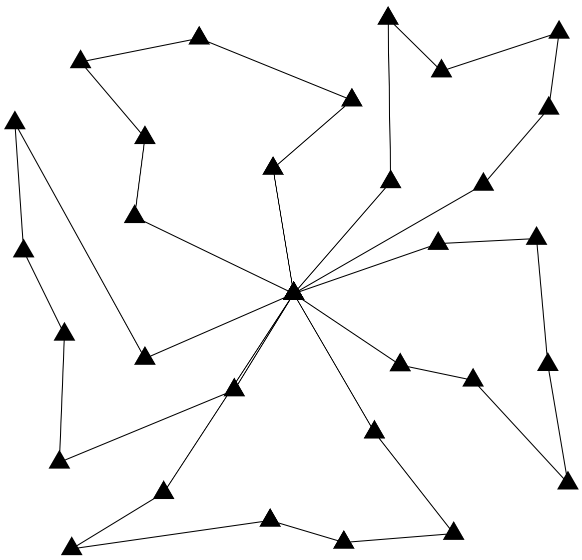
Figure H-8: Participant 7 solutions (continued)



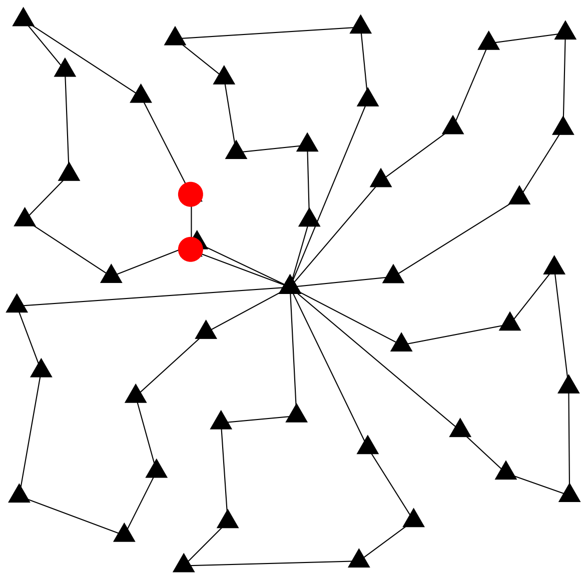
(i) p7-scenario08-HITL



(j) p7-scenario09-HITL

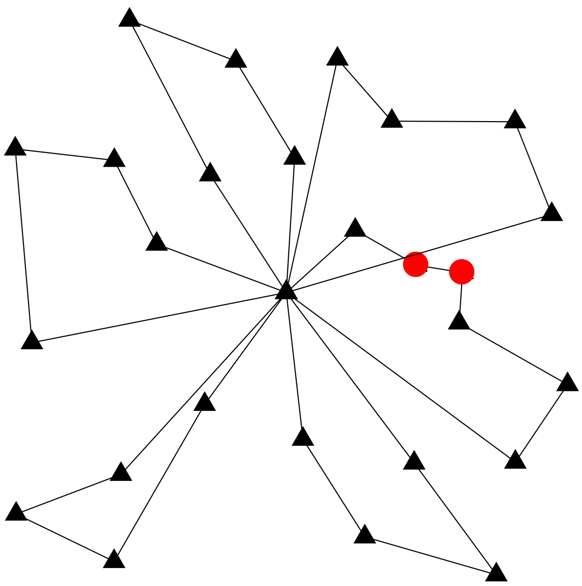


(k) p7-scenario10-HITL

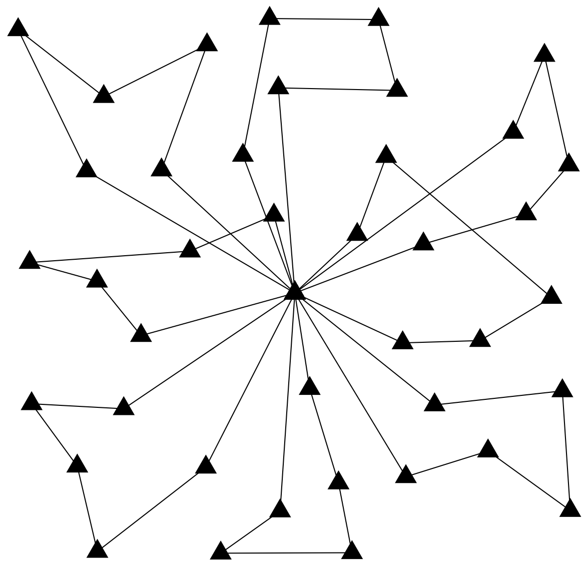


(l) p7-scenario11-HITL

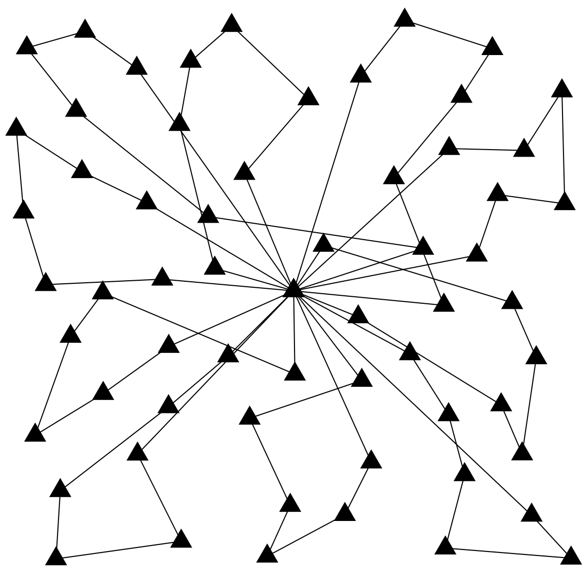
Figure H-8: Participant 7 solutions (continued)



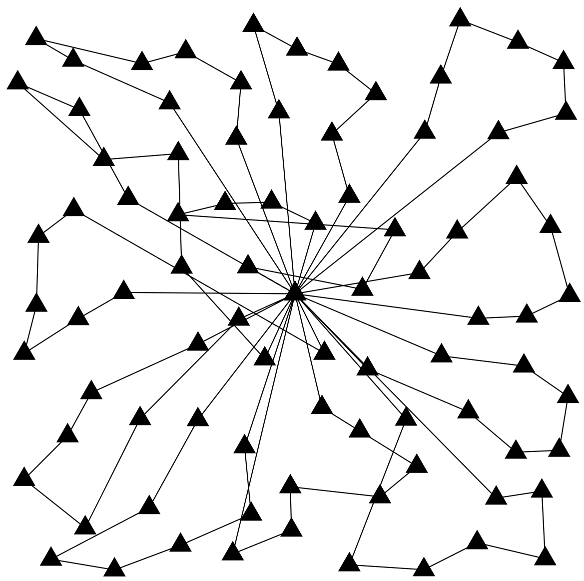
(m) p7-scenario12-HITL



(n) p7-scenario13-HITL



(o) p7-scenario14-HITL

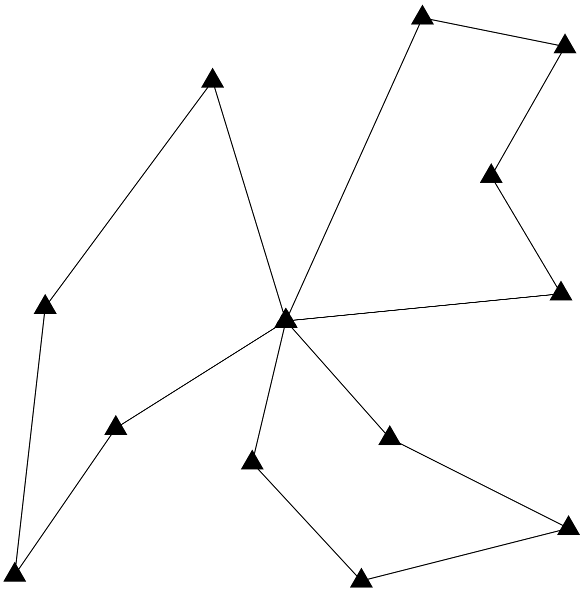


(p) p7-scenario15-HITL

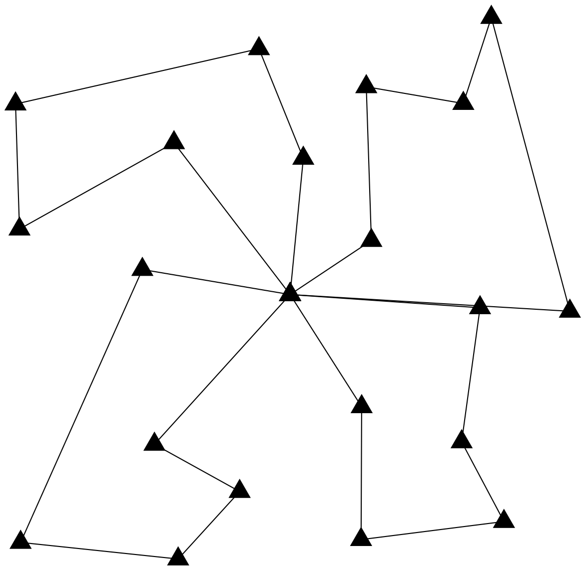
Figure H-8: Participant 7 solutions (continued)

H-3-8 Participant 8

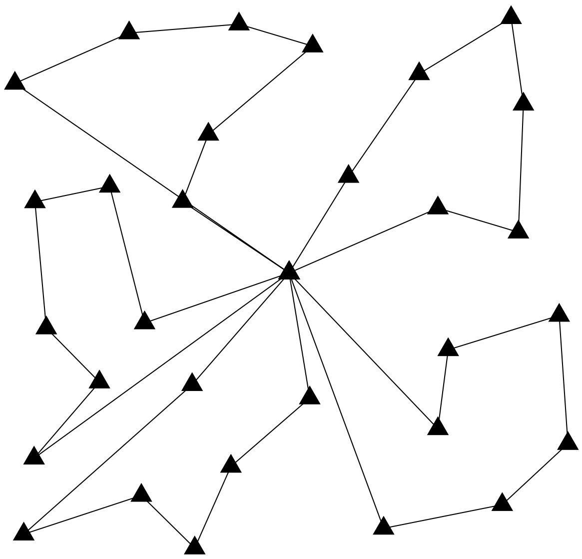
Participant 8 solutions.



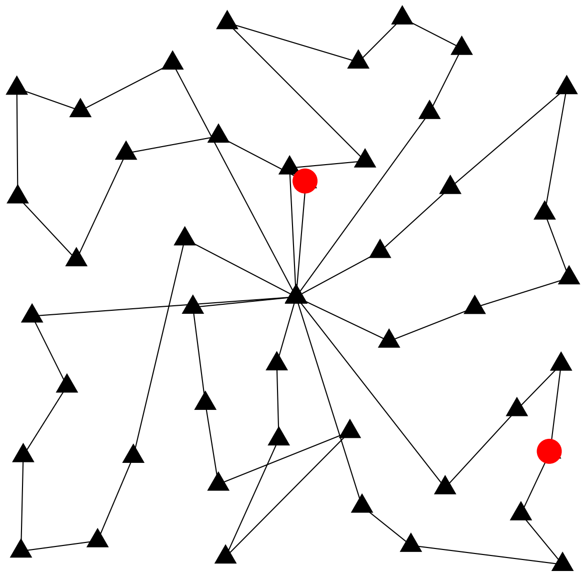
(a) p8-scenario00-HITL



(b) p8-scenario01-HITL

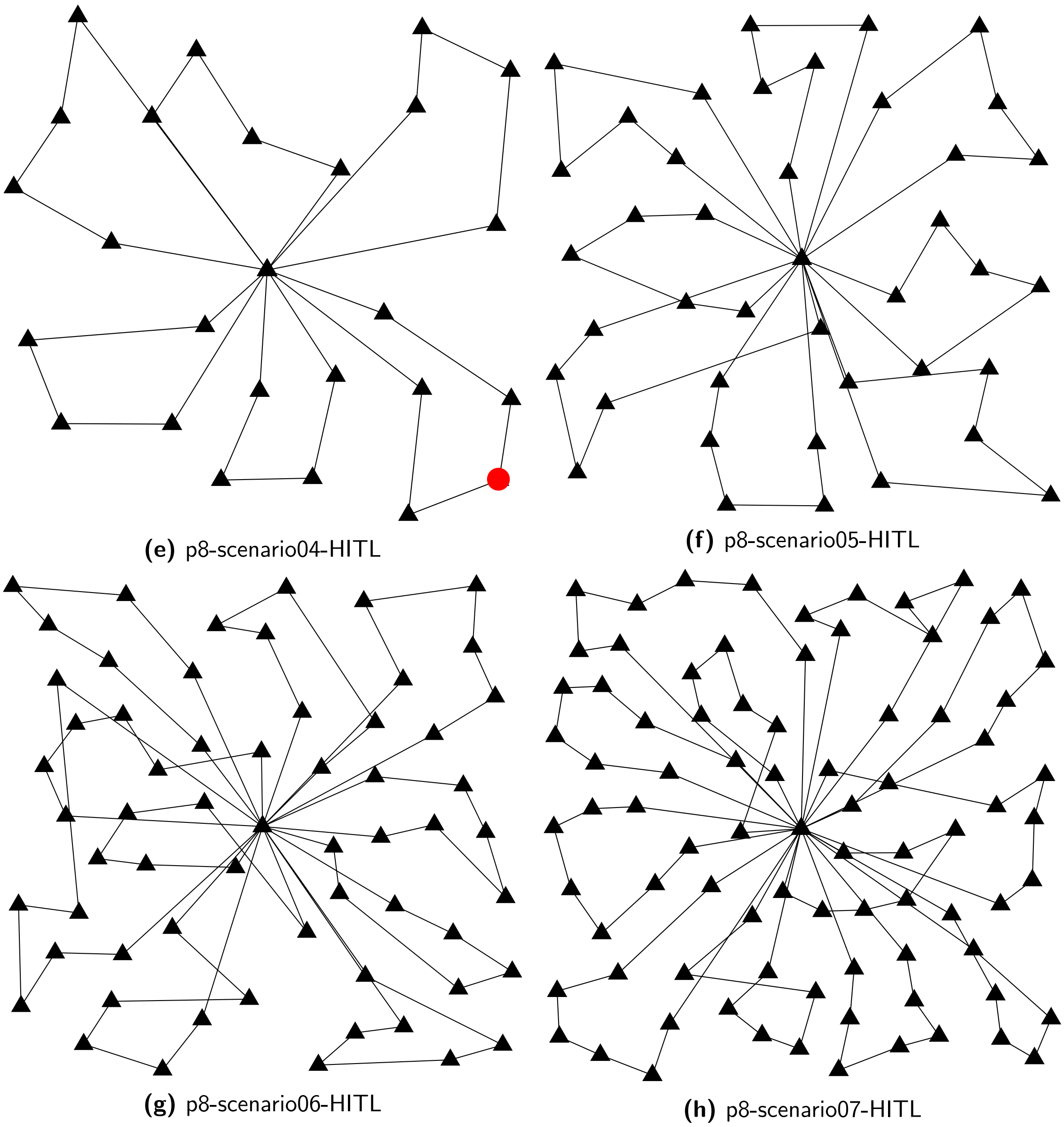


(c) p8-scenario02-HITL



(d) p8-scenario03-HITL

Figure H-9: Participant 8 solutions



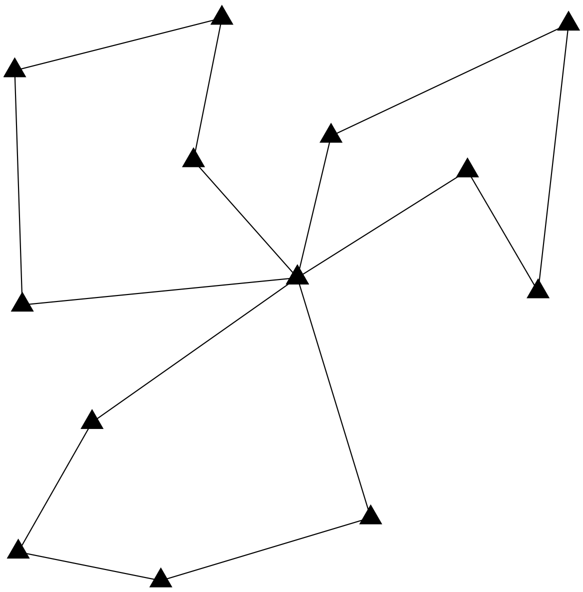
(e) p8-scenario04-HITL

(f) p8-scenario05-HITL

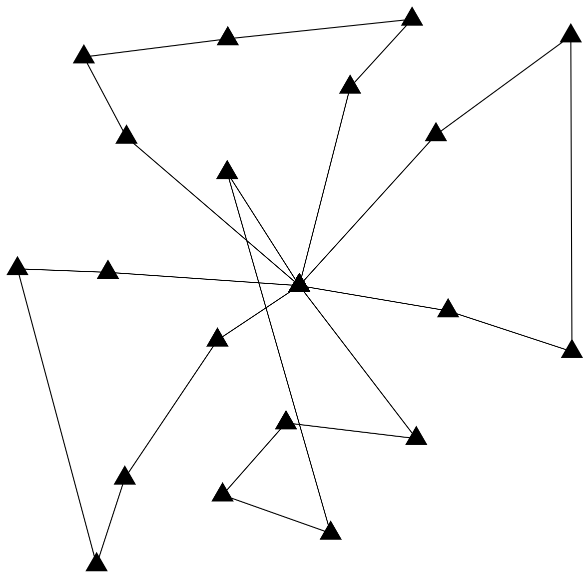
(g) p8-scenario06-HITL

(h) p8-scenario07-HITL

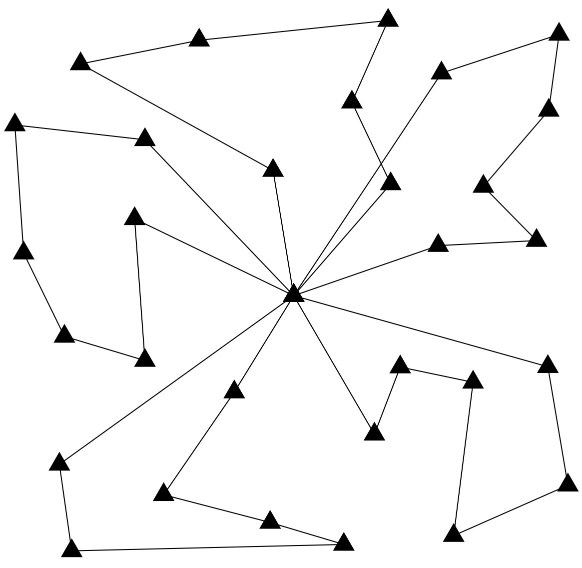
Figure H-9: Participant 8 solutions (continued)



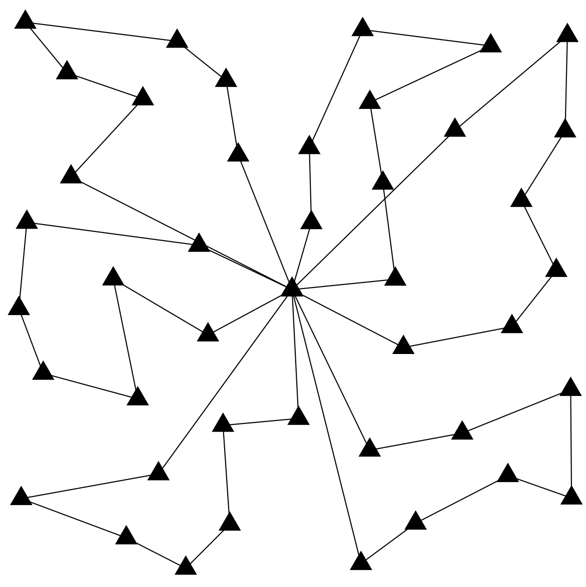
(i) p8-scenario08-HITL



(j) p8-scenario09-HITL

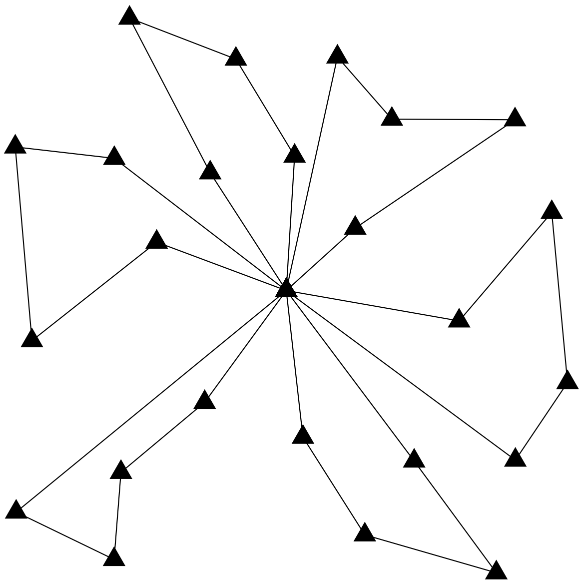


(k) p8-scenario10-HITL

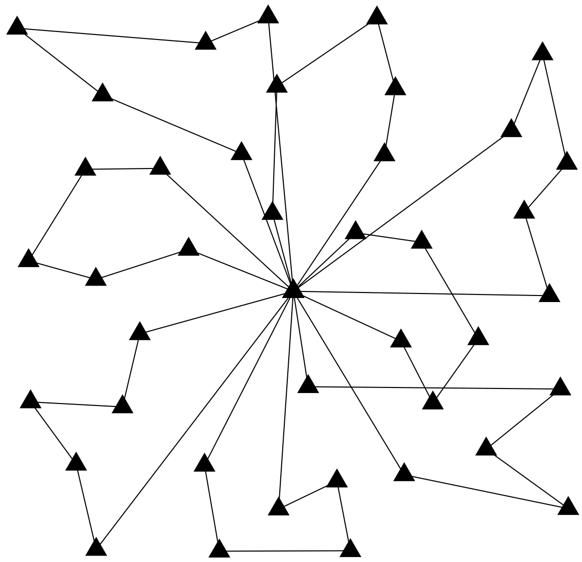


(l) p8-scenario11-HITL

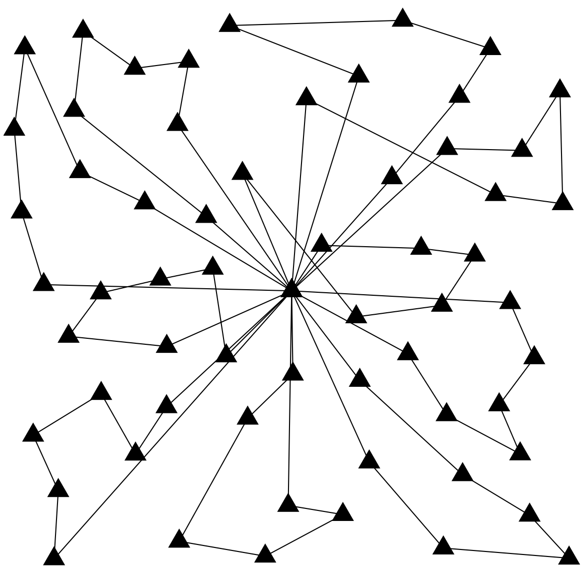
Figure H-9: Participant 8 solutions (continued)



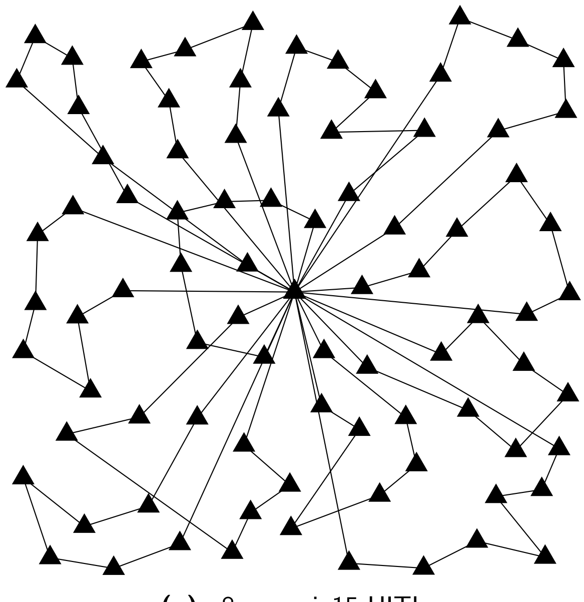
(m) p8-scenario12-HITL



(n) p8-scenario13-HITL



(o) p8-scenario14-HITL

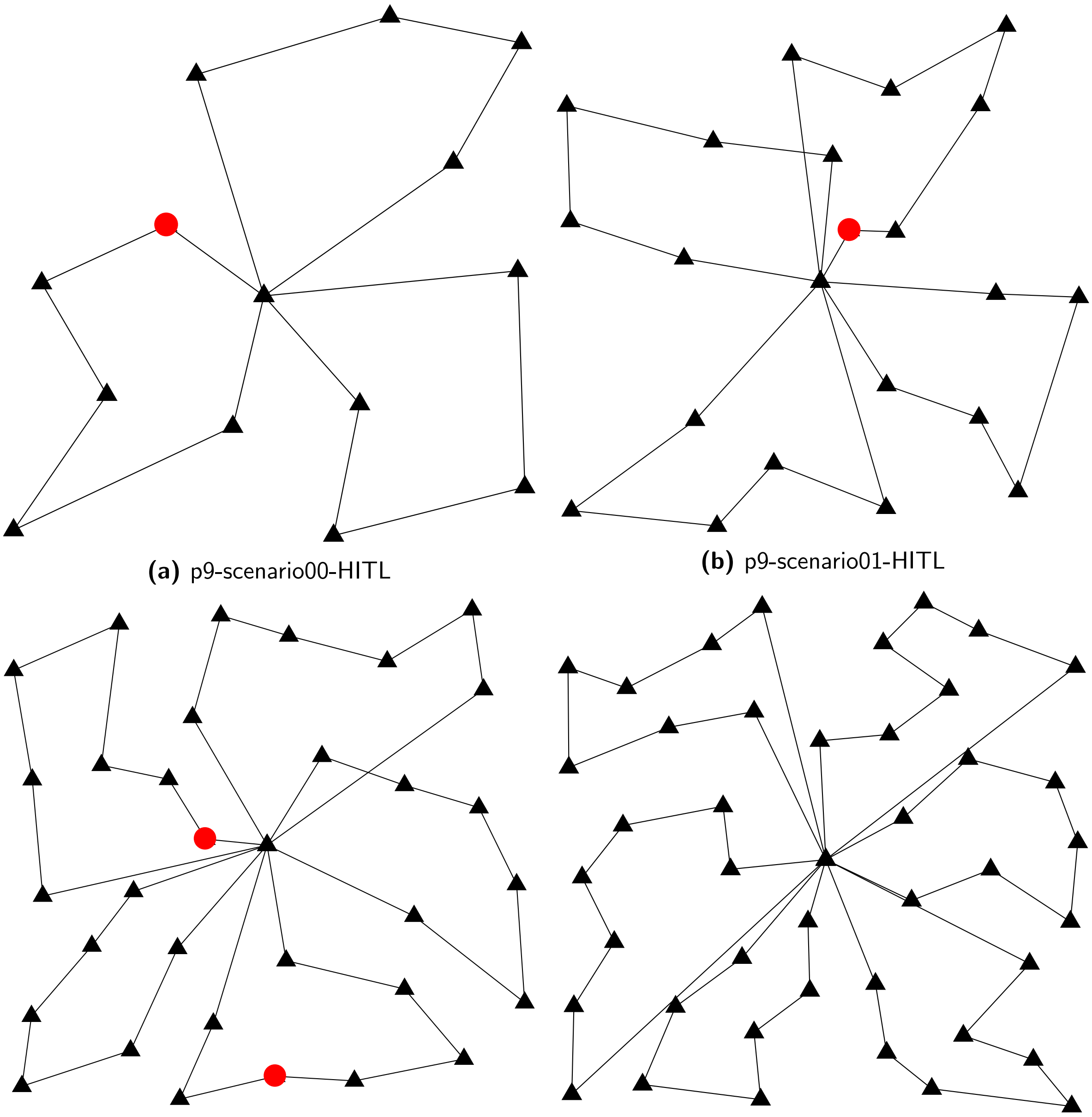


(p) p8-scenario15-HITL

Figure H-9: Participant 8 solutions (continued)

H-3-9 Participant 9

Participant 9 solutions.



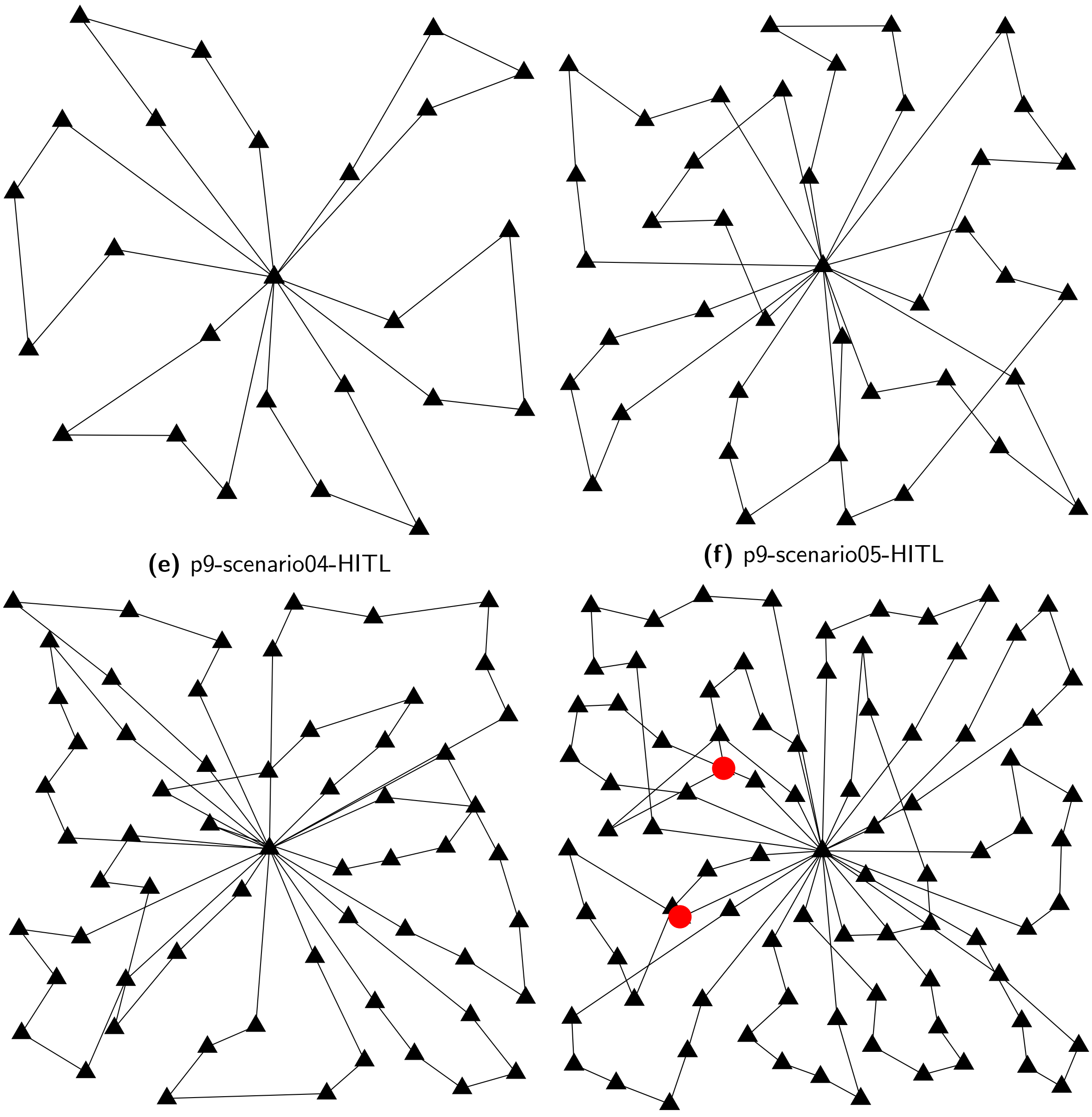
(a) p9-scenario00-HITL

(b) p9-scenario01-HITL

(c) p9-scenario02-HITL

(d) p9-scenario03-HITL

Figure H-10: Participant 9 solutions



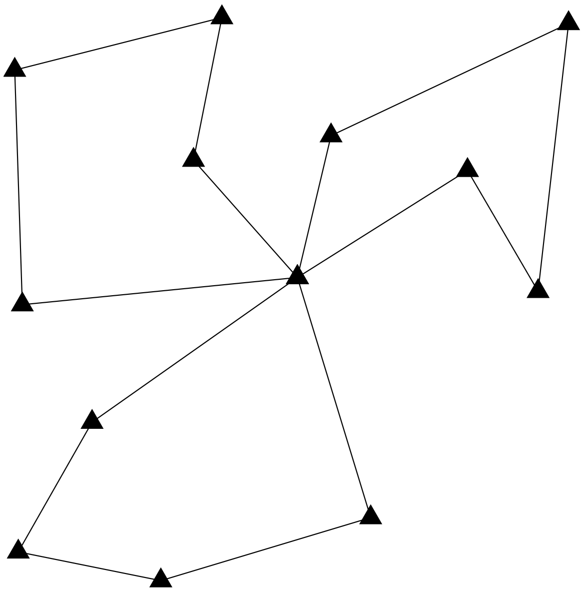
(e) p9-scenario04-HITL

(f) p9-scenario05-HITL

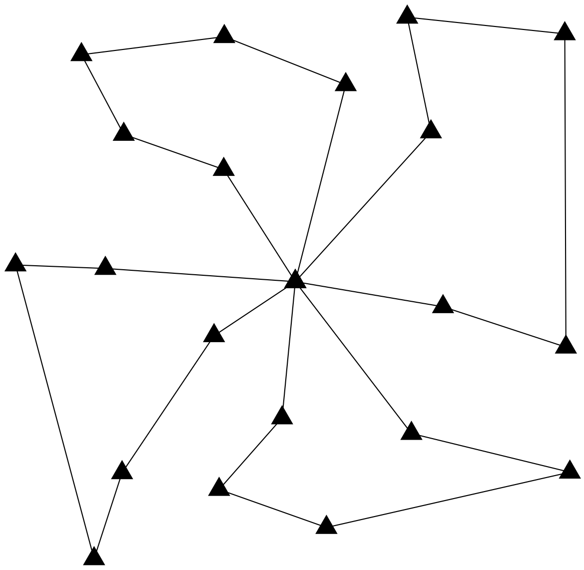
(g) p9-scenario06-HITL

(h) p9-scenario07-HITL

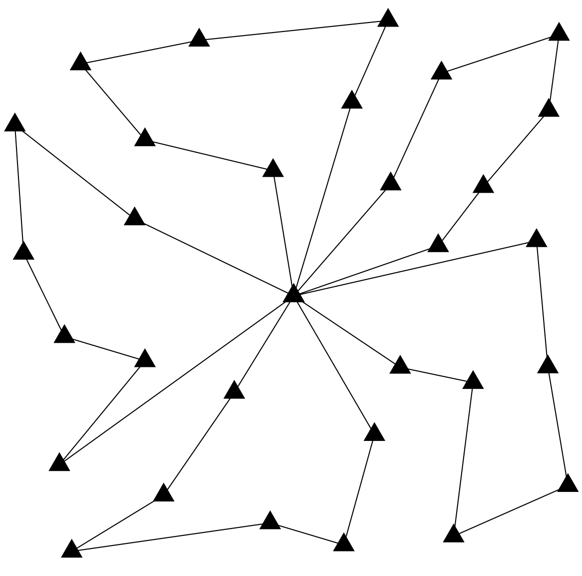
Figure H-10: Participant 9 solutions (continued)



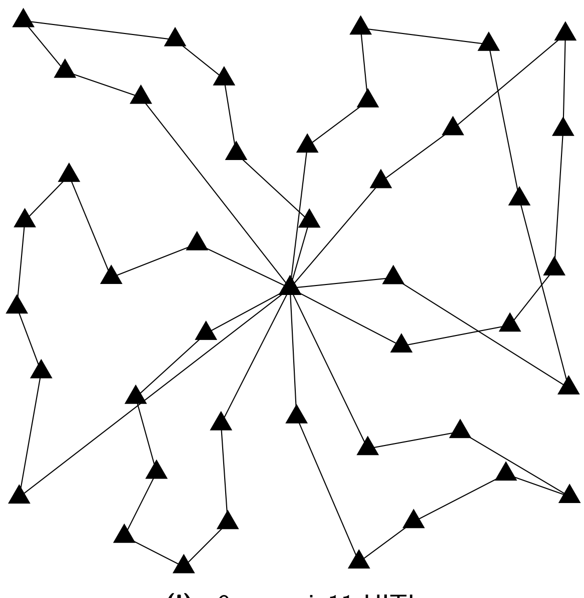
(i) p9-scenario08-HITL



(j) p9-scenario09-HITL

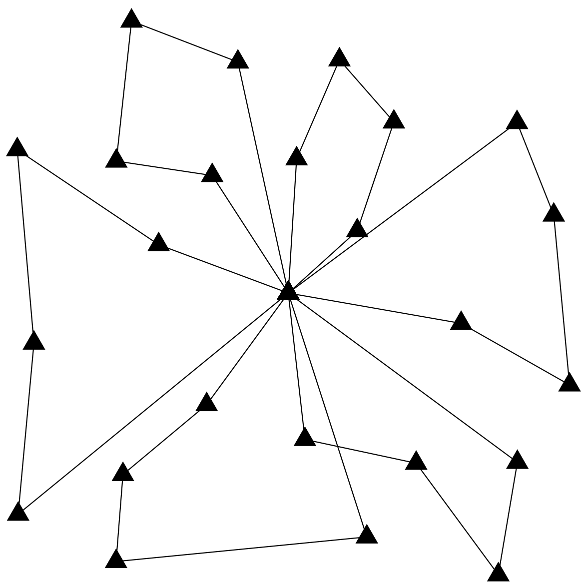


(k) p9-scenario10-HITL

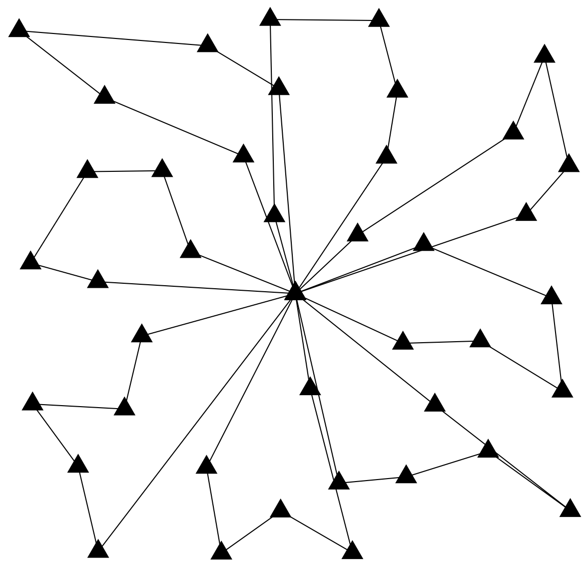


(l) p9-scenario11-HITL

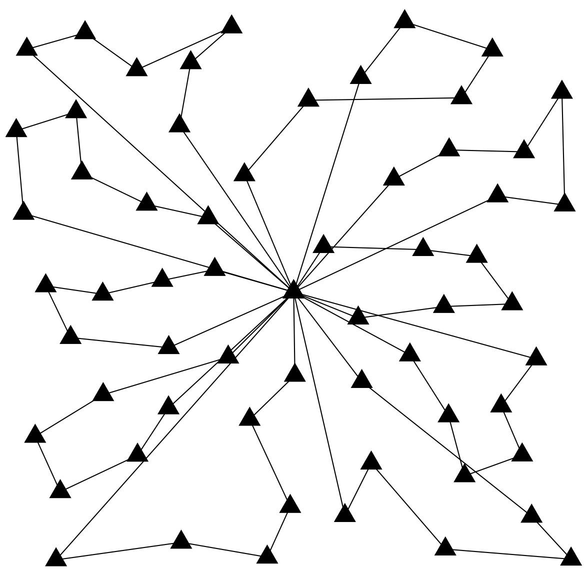
Figure H-10: Participant 9 solutions (continued)



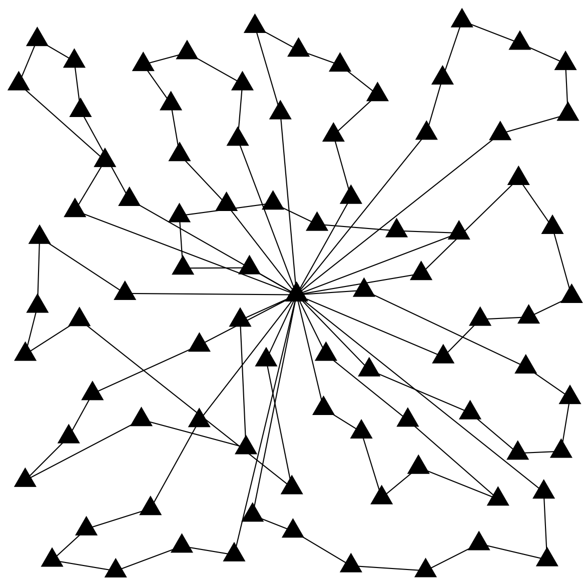
(m) p9-scenario12-HITL



(n) p9-scenario13-HITL



(o) p9-scenario14-HITL

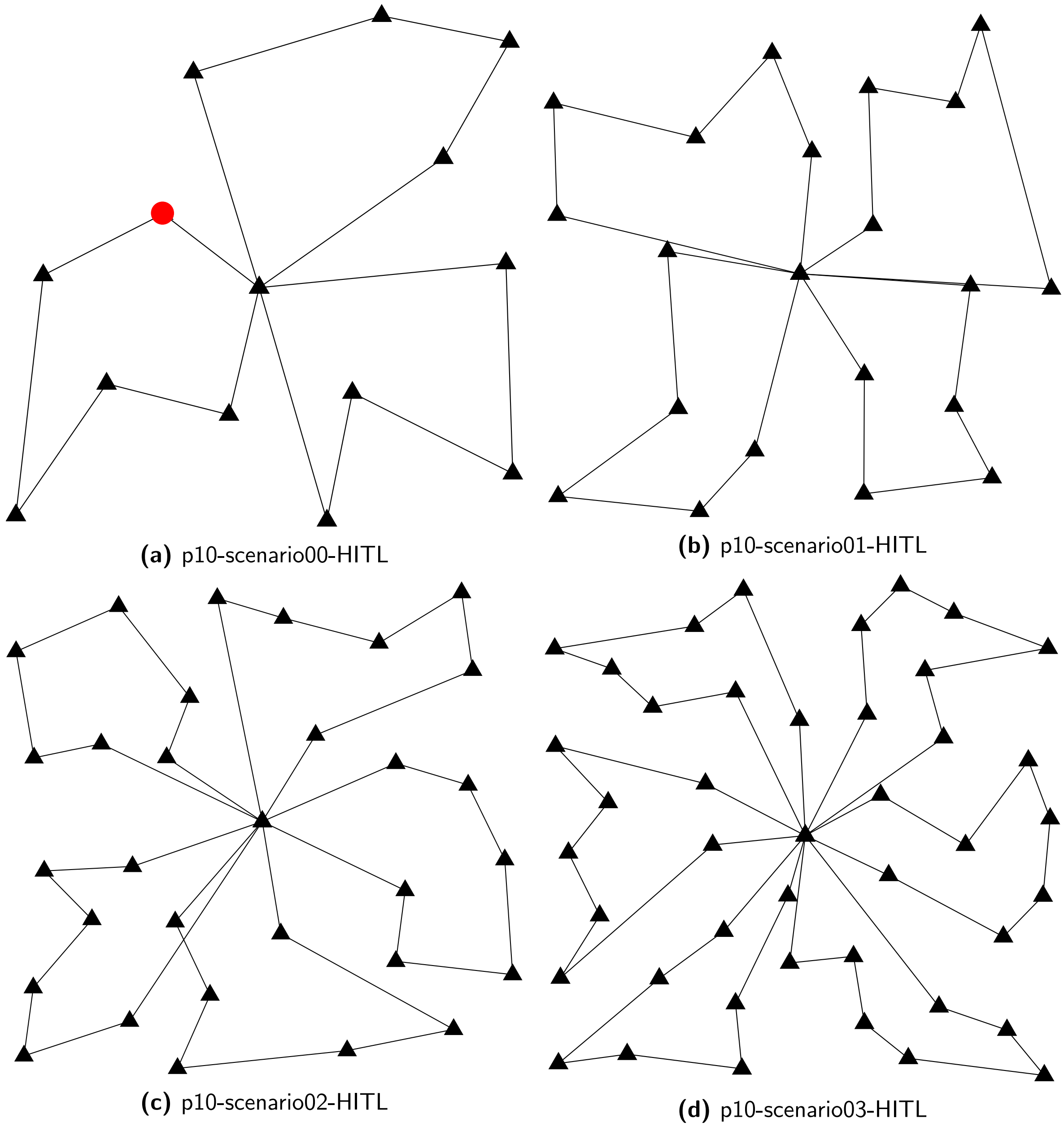


(p) p9-scenario15-HITL

Figure H-10: Participant 9 solutions (continued)

H-3-10 Participant 10

Participant 10 solutions.



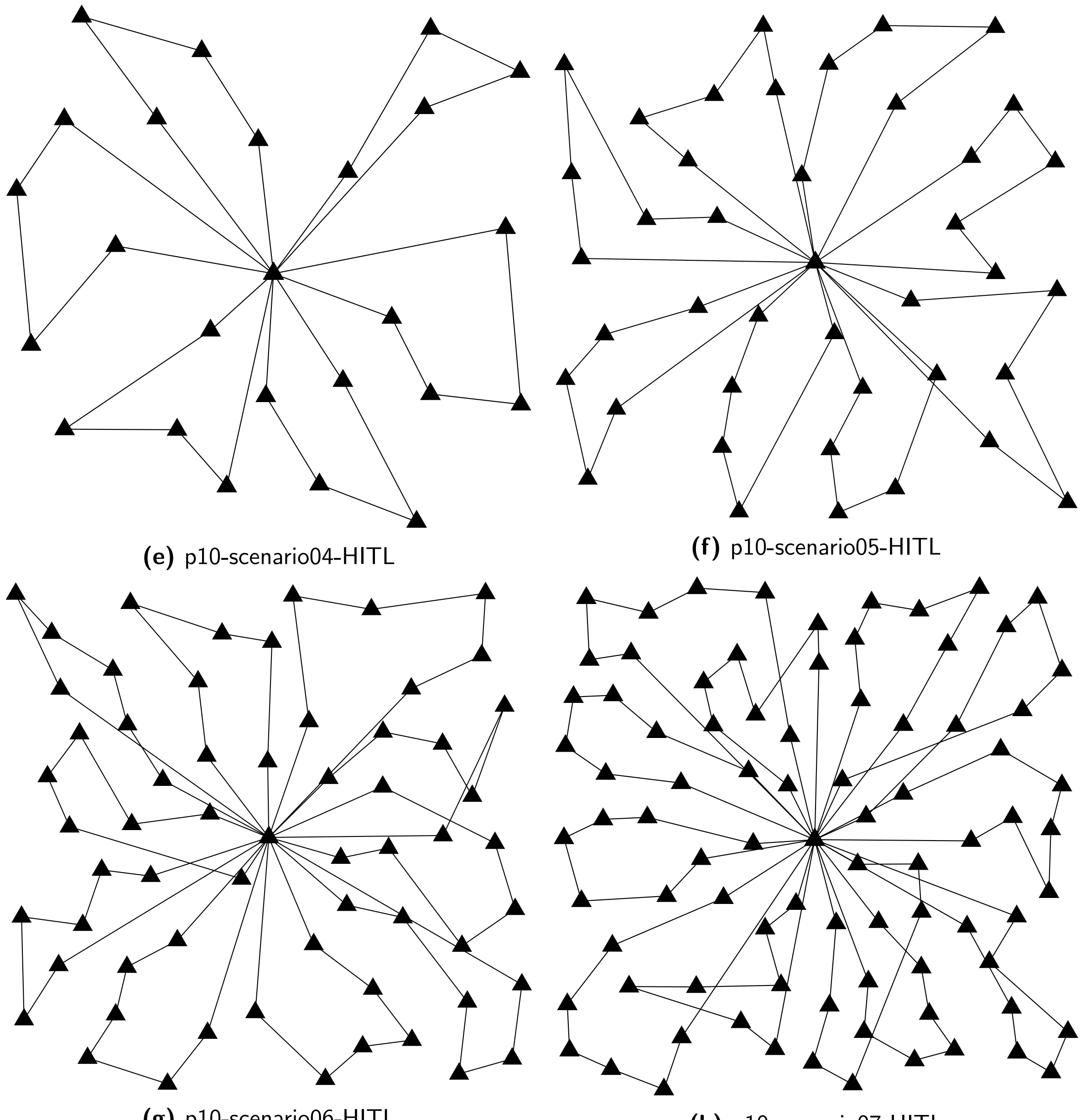
(a) p10-scenario00-HITL

(b) p10-scenario01-HITL

(c) p10-scenario02-HITL

(d) p10-scenario03-HITL

Figure H-11: Participant 10 solutions



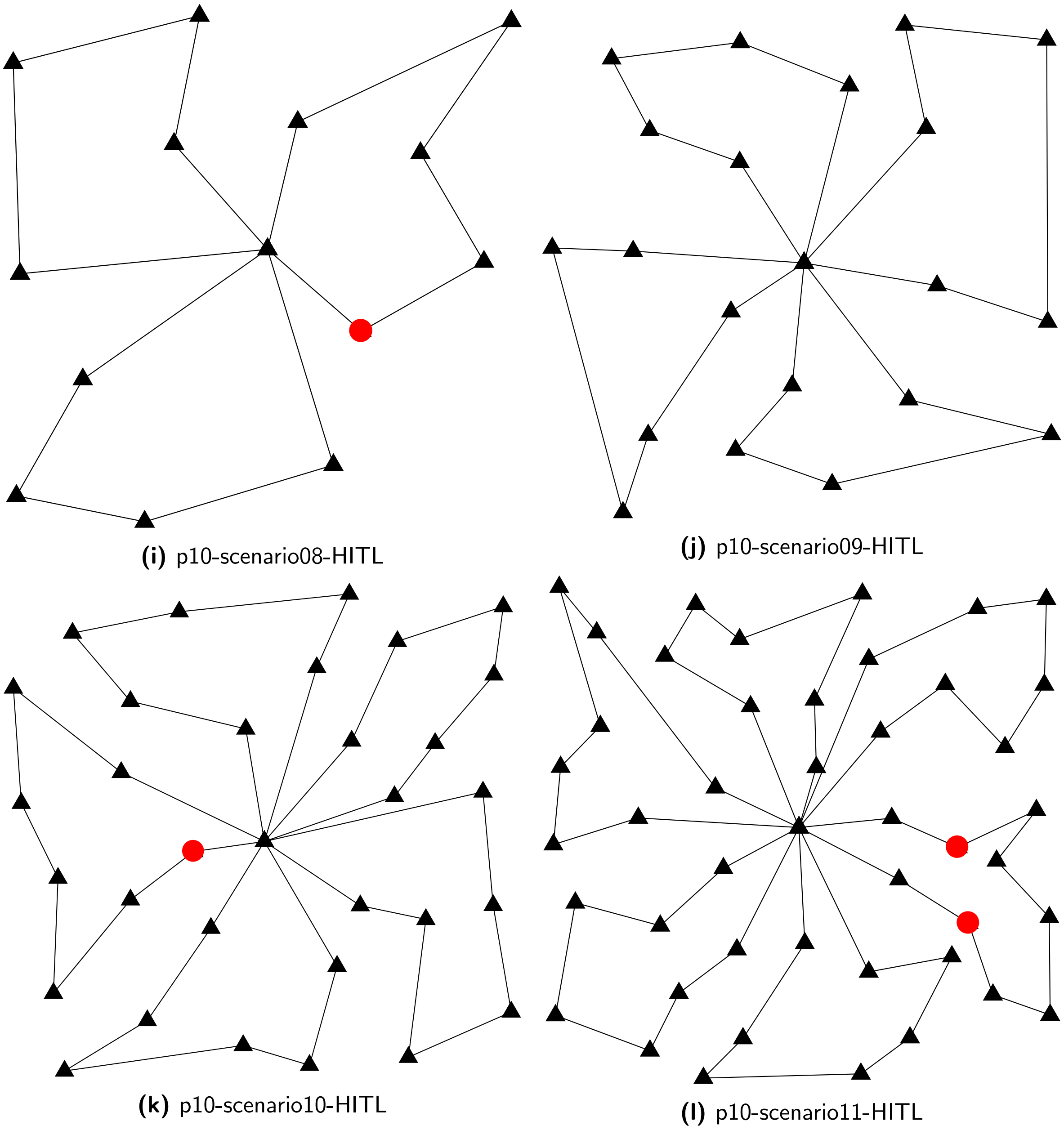
(e) p10-scenario04-HITL

(f) p10-scenario05-HITL

(g) p10-scenario06-HITL

(h) p10-scenario07-HITL

Figure H-11: Participant 10 solutions (continued)



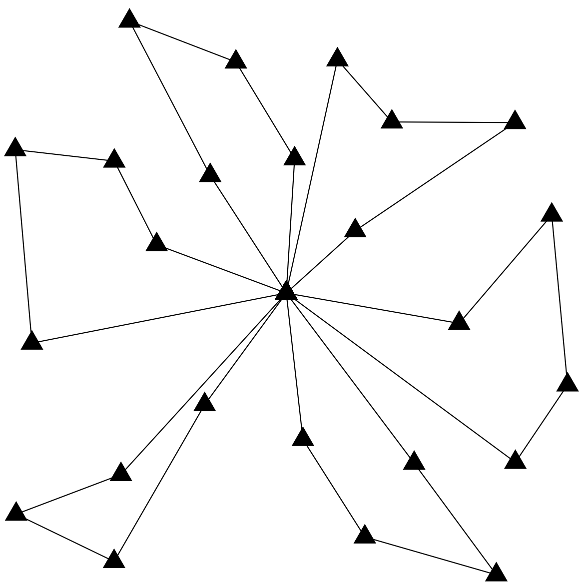
(i) p10-scenario08-HITL

(j) p10-scenario09-HITL

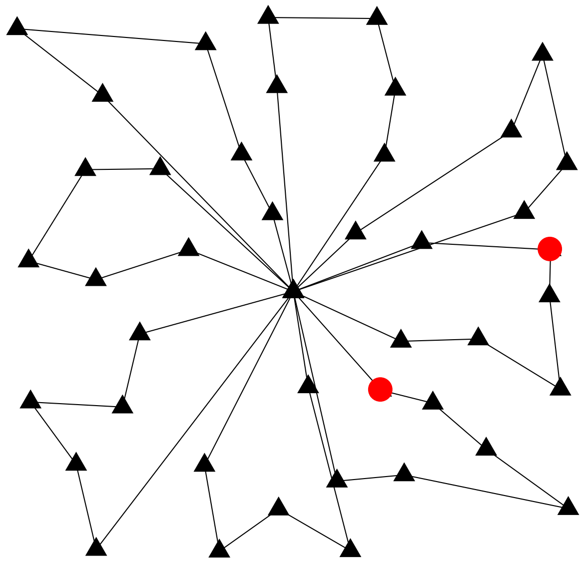
(k) p10-scenario10-HITL

(l) p10-scenario11-HITL

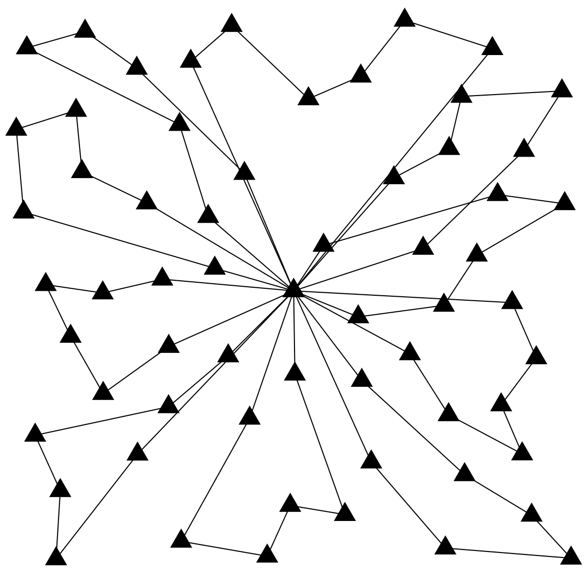
Figure H-11: Participant 10 solutions (continued)



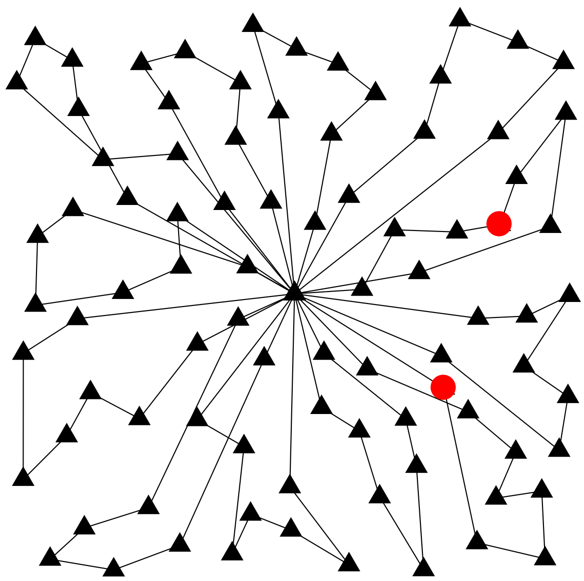
(m) p10-scenario12-HITL



(n) p10-scenario13-HITL



(o) p10-scenario14-HITL

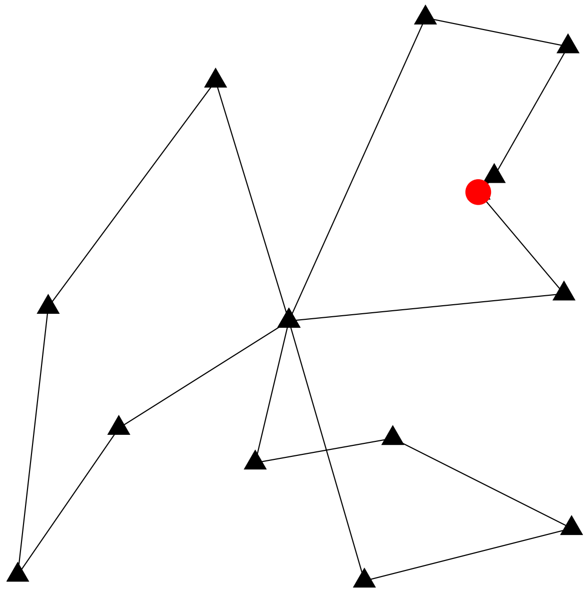


(p) p10-scenario15-HITL

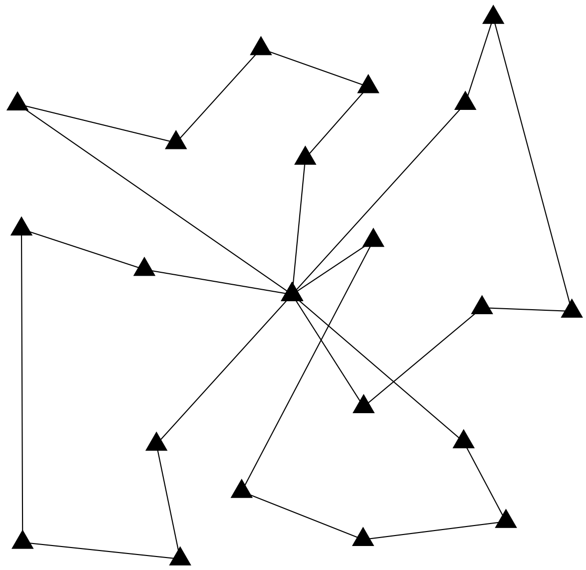
Figure H-11: Participant 10 solutions (continued)

H-3-11 Participant 11

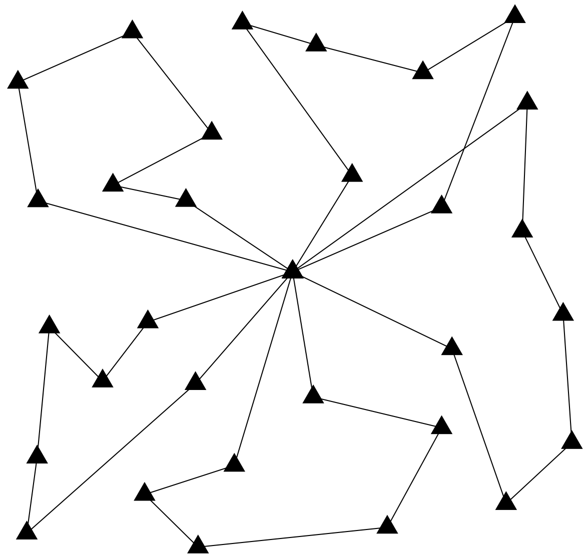
Participant 11 solutions.



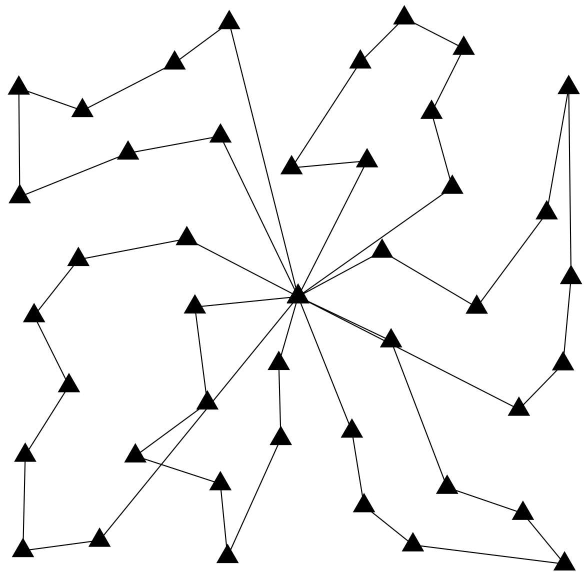
(a) p11-scenario00-HITL



(b) p11-scenario01-HITL

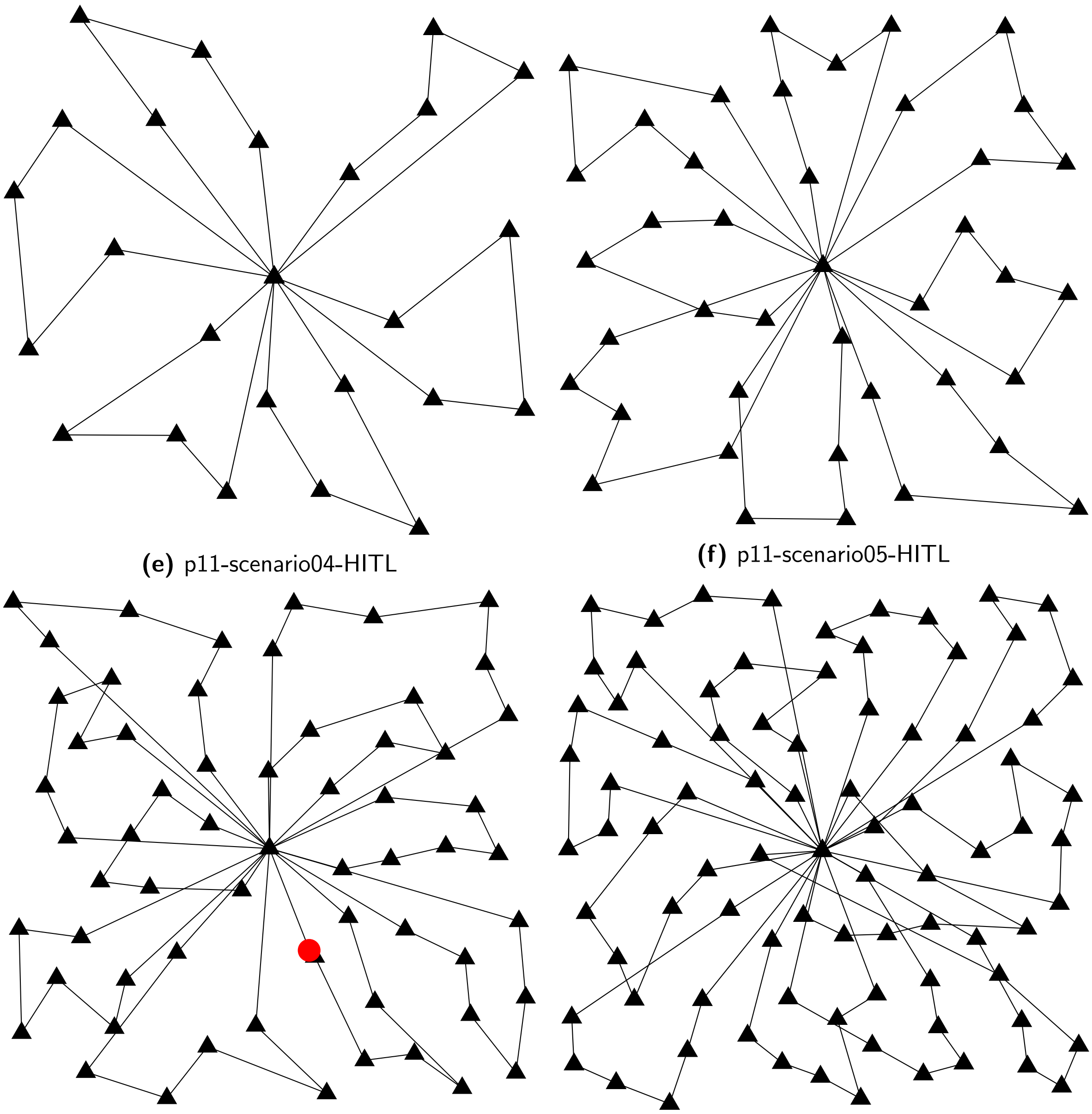


(c) p11-scenario02-HITL



(d) p11-scenario03-HITL

Figure H-12: Participant 11 solutions



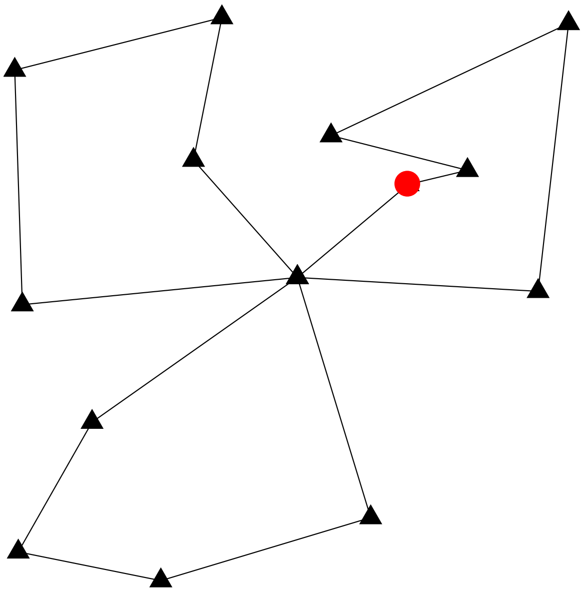
(e) p11-scenario04-HITL

(f) p11-scenario05-HITL

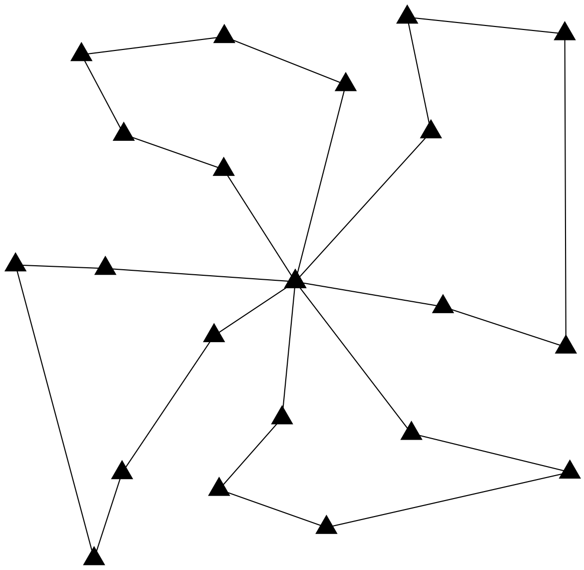
(g) p11-scenario06-HITL

(h) p11-scenario07-HITL

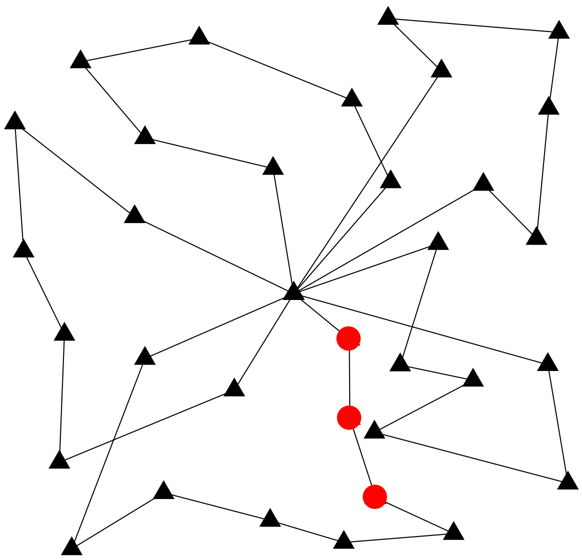
Figure H-12: Participant 11 solutions (continued)



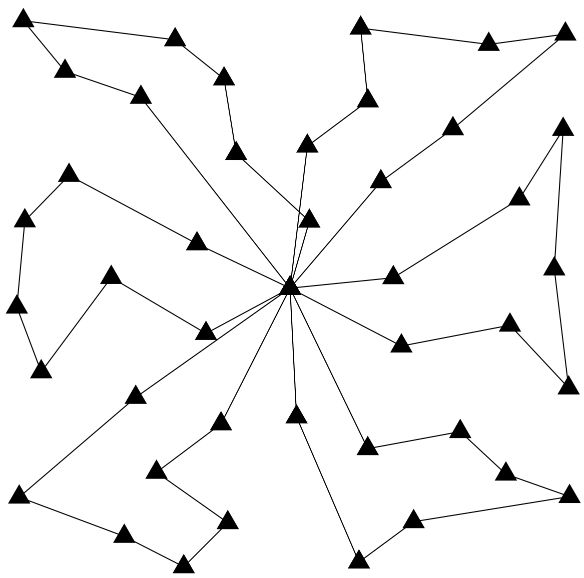
(i) p11-scenario08-HITL



(j) p11-scenario09-HITL

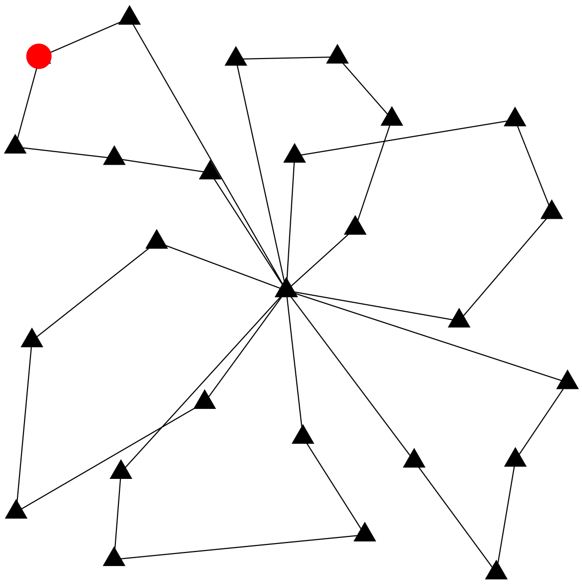


(k) p11-scenario10-HITL

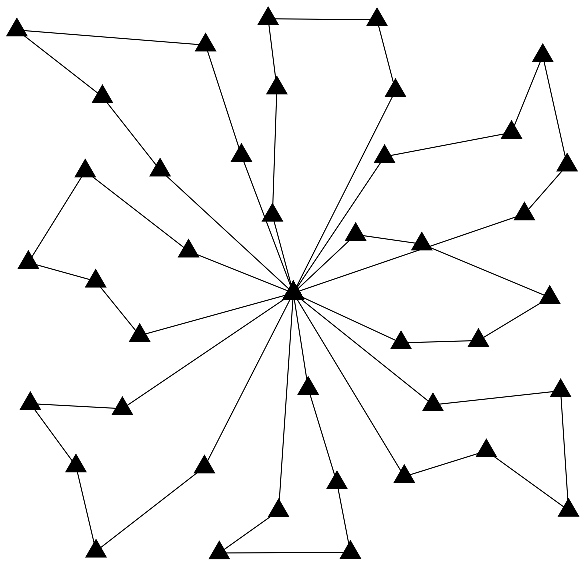


(l) p11-scenario11-HITL

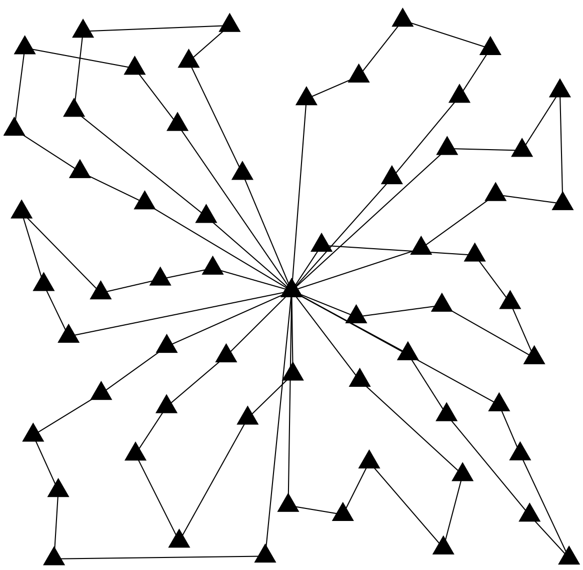
Figure H-12: Participant 11 solutions (continued)



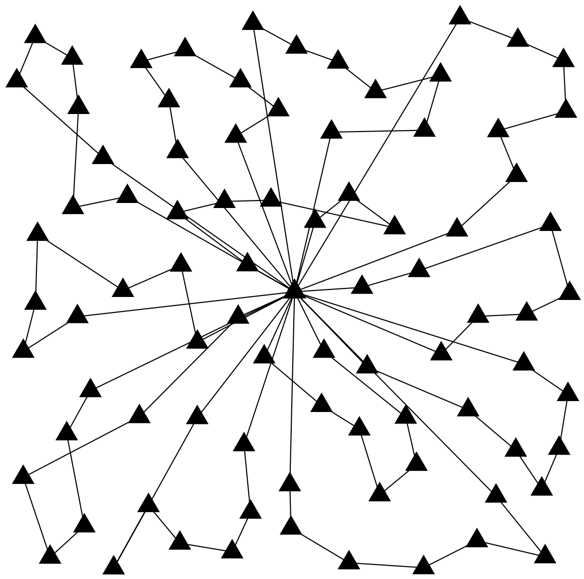
(m) p11-scenario12-HITL



(n) p11-scenario13-HITL



(o) p11-scenario14-HITL

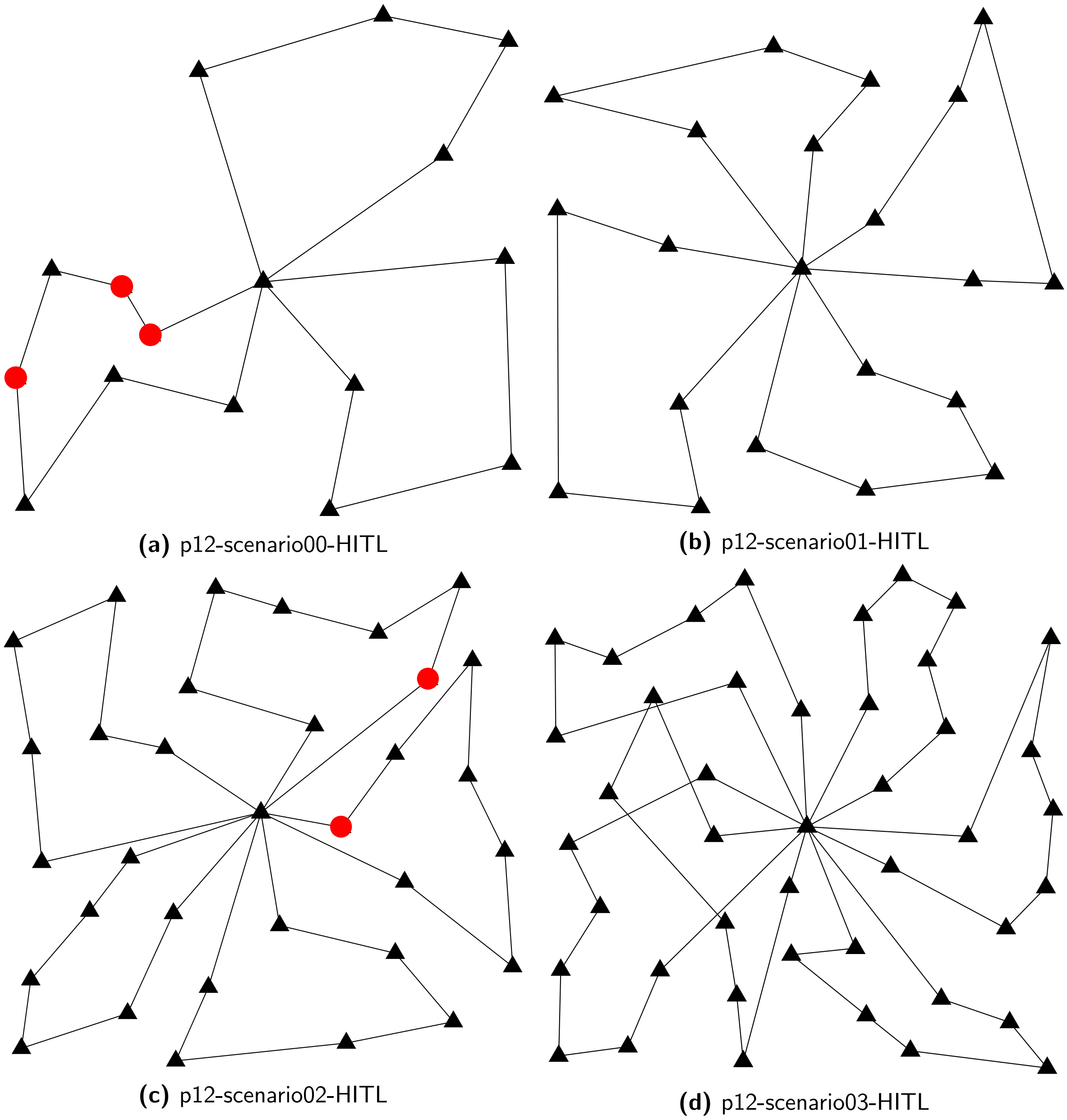


(p) p11-scenario15-HITL

Figure H-12: Participant 11 solutions (continued)

H-3-12 Participant 12

Participant 12 solutions.



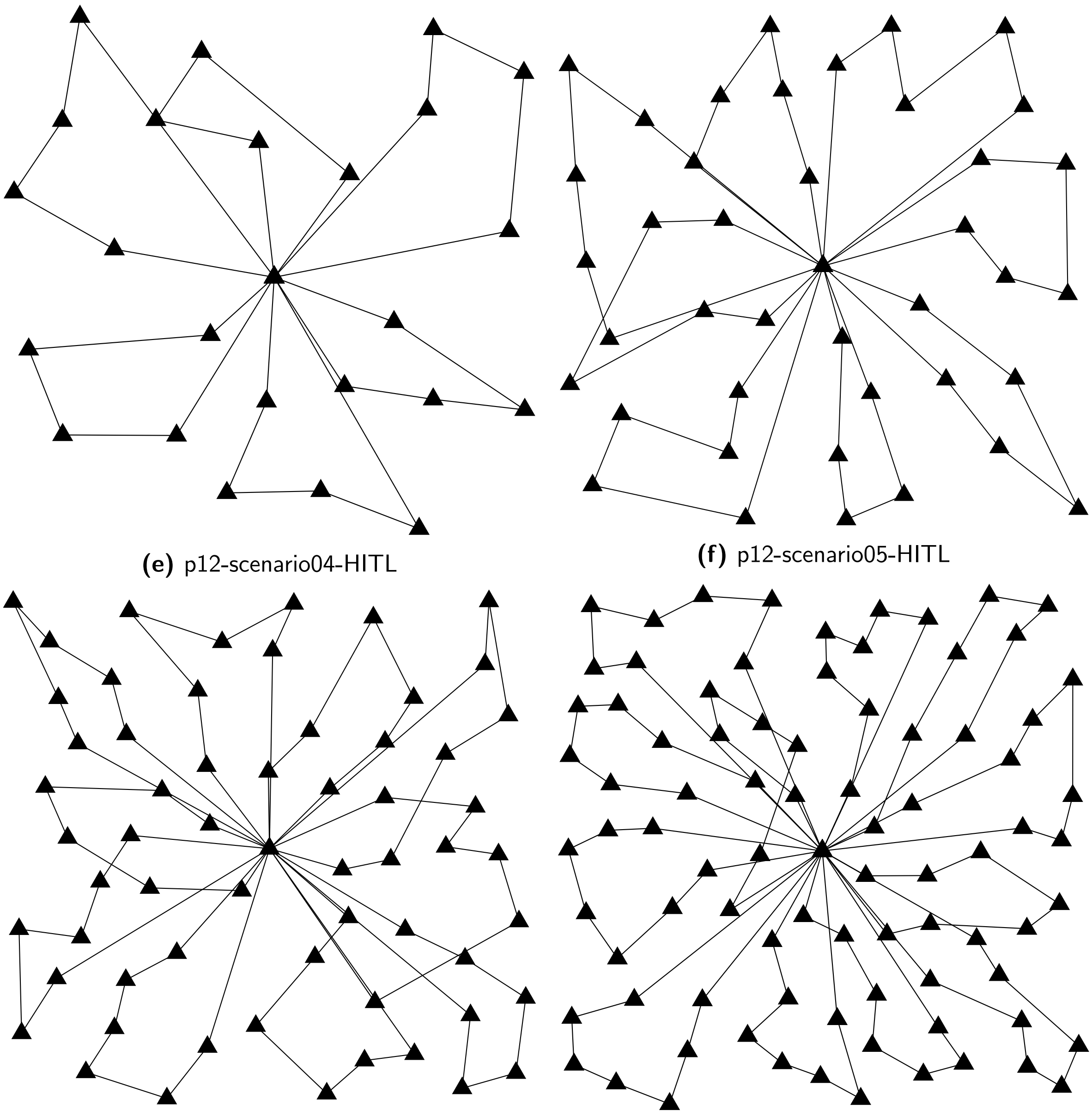
(a) p12-scenario00-HITL

(b) p12-scenario01-HITL

(c) p12-scenario02-HITL

(d) p12-scenario03-HITL

Figure H-13: Participant 12 solutions



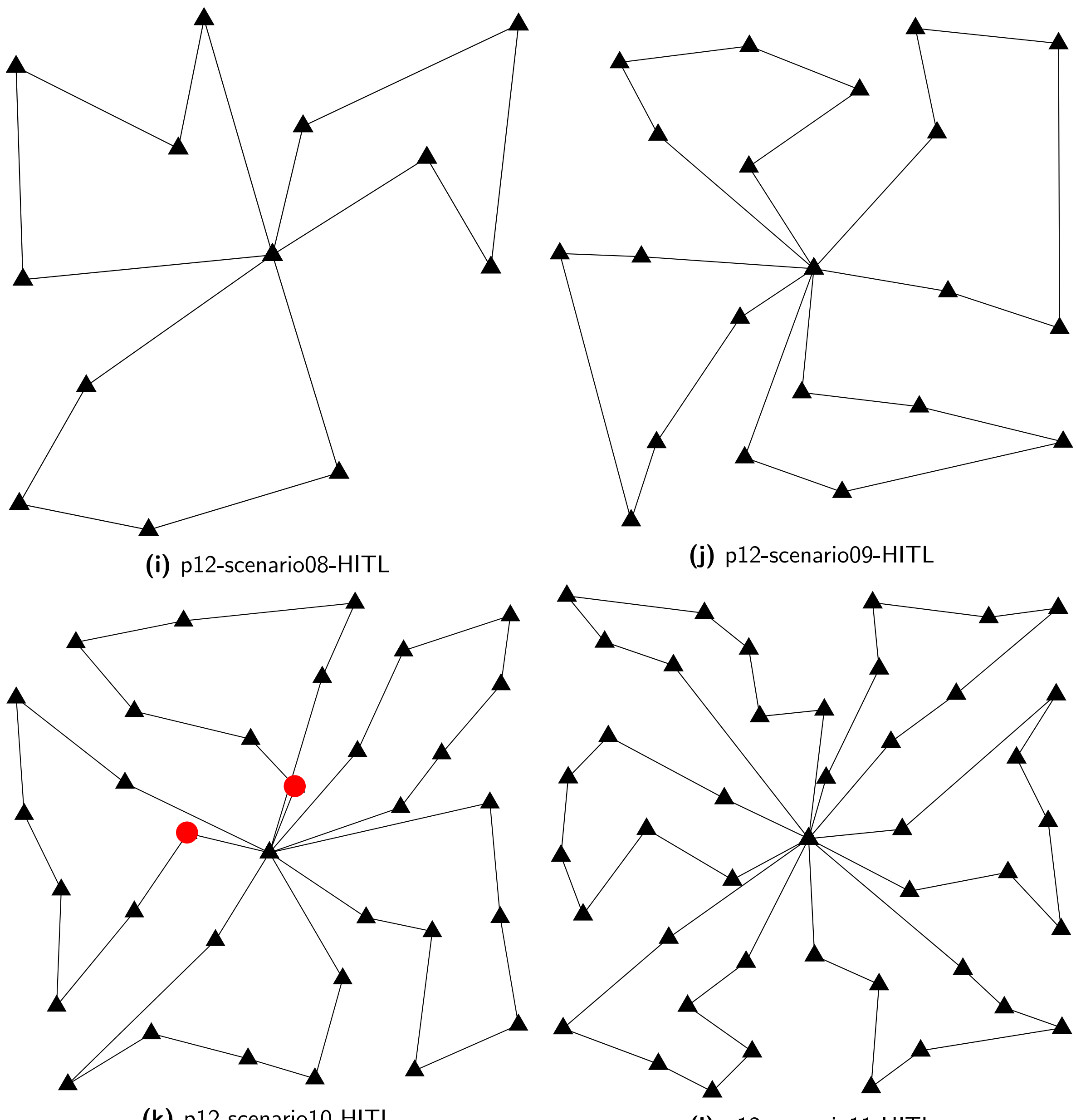
(e) p12-scenario04-HITL

(f) p12-scenario05-HITL

(g) p12-scenario06-HITL

(h) p12-scenario07-HITL

Figure H-13: Participant 12 solutions (continued)



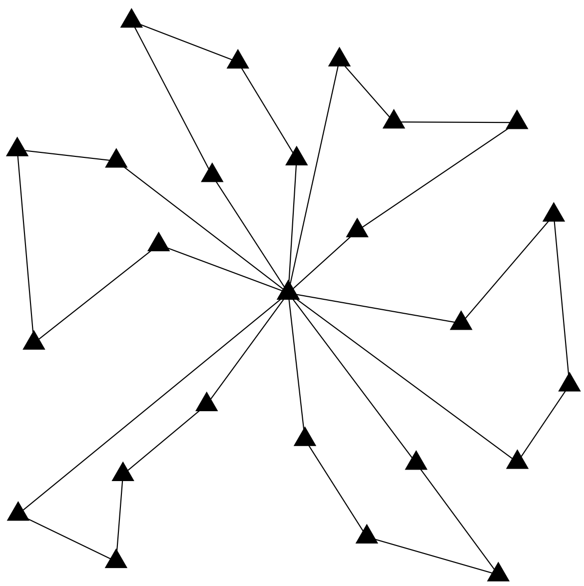
(i) p12-scenario08-HITL

(j) p12-scenario09-HITL

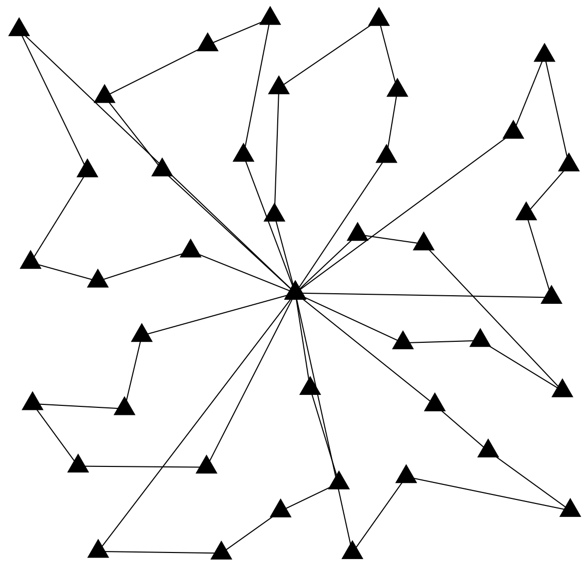
(k) p12-scenario10-HITL

(l) p12-scenario11-HITL

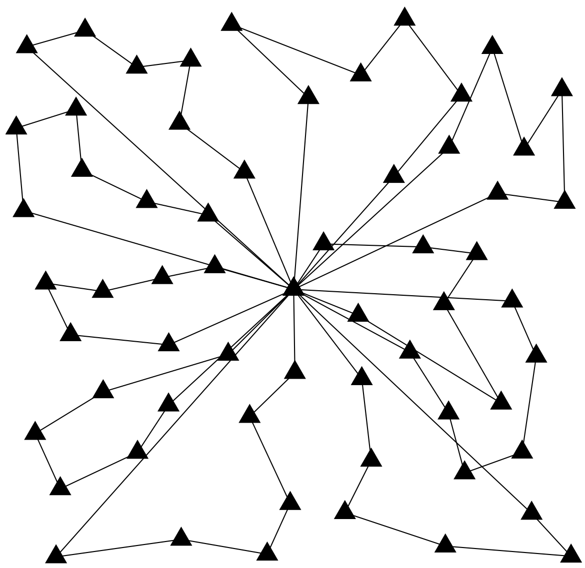
Figure H-13: Participant 12 solutions (continued)



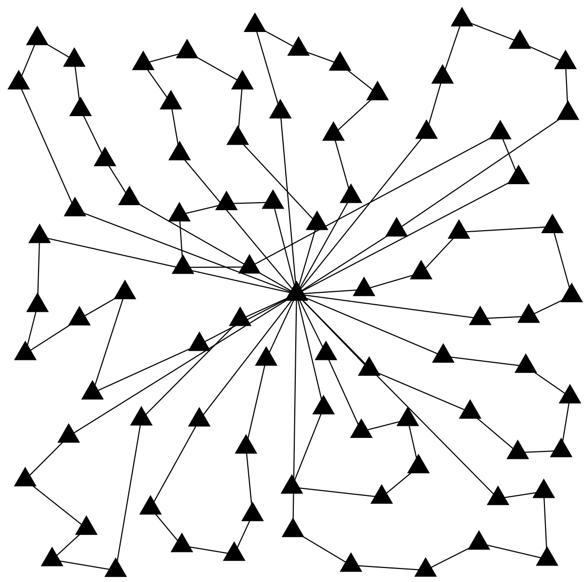
(m) p12-scenario12-HITL



(n) p12-scenario13-HITL



(o) p12-scenario14-HITL

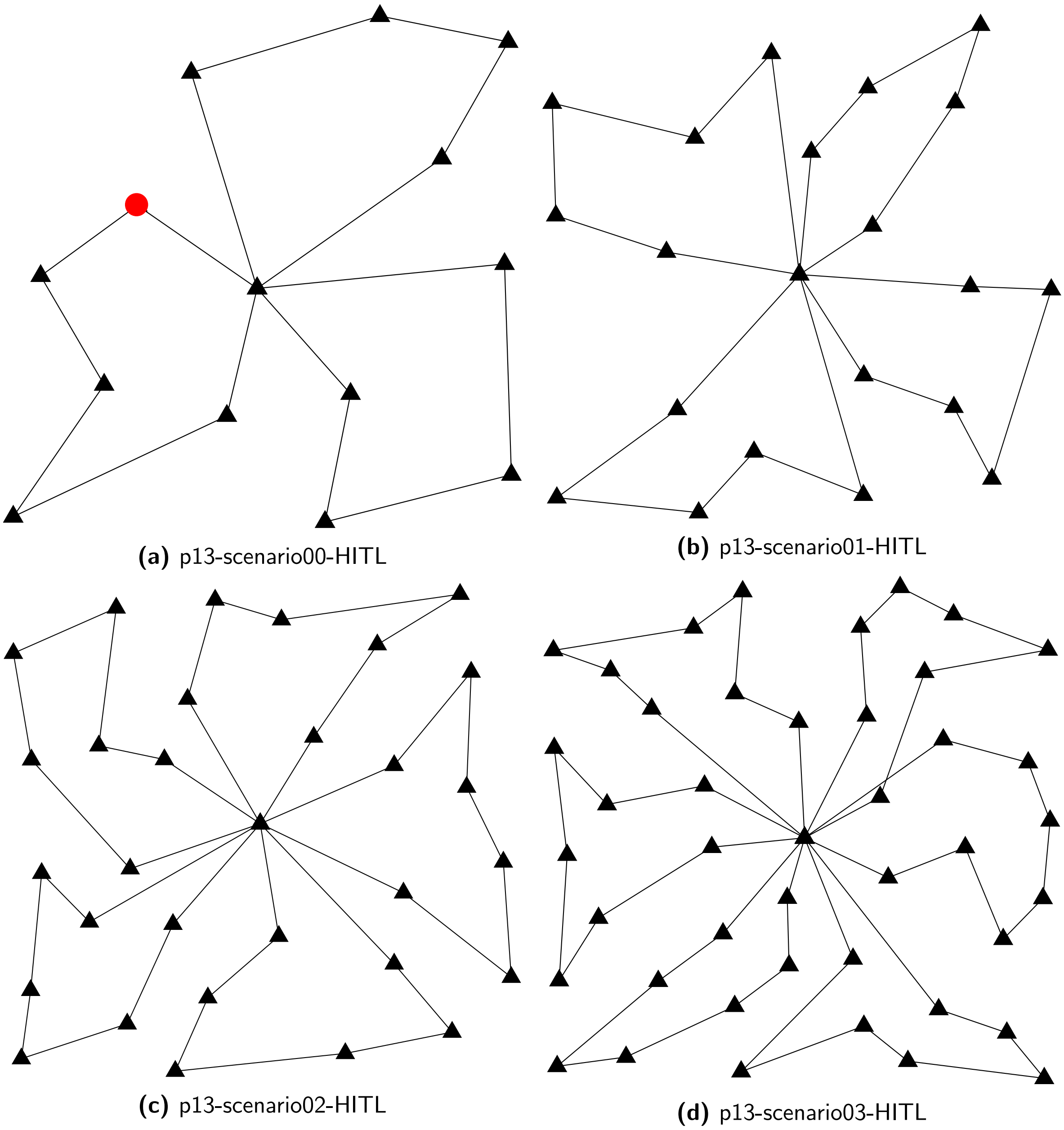


(p) p12-scenario15-HITL

Figure H-13: Participant 12 solutions (continued)

H-3-13 Participant 13

Participant 13 solutions.



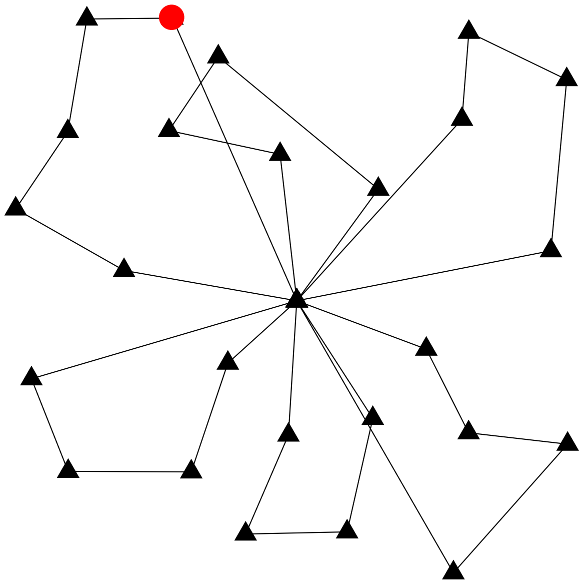
(a) p13-scenario00-HITL

(b) p13-scenario01-HITL

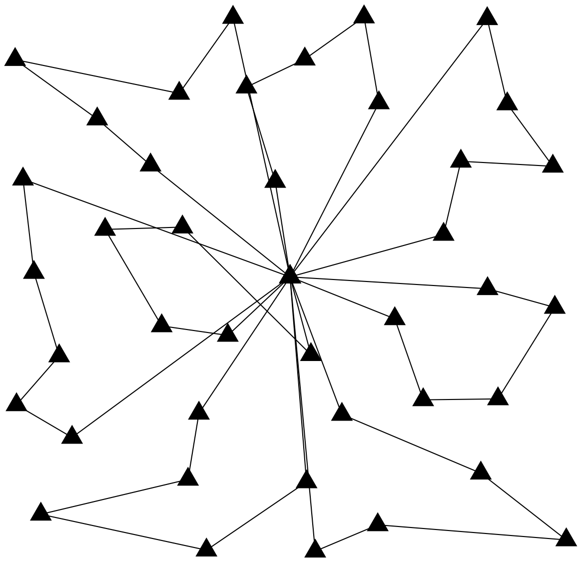
(c) p13-scenario02-HITL

(d) p13-scenario03-HITL

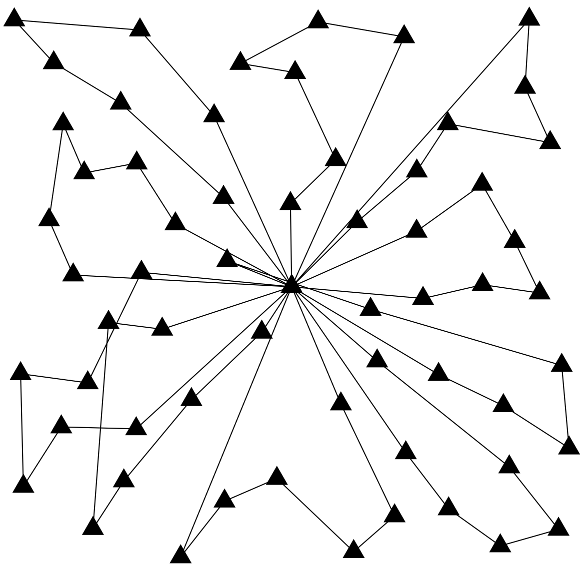
Figure H-14: Participant 13 solutions



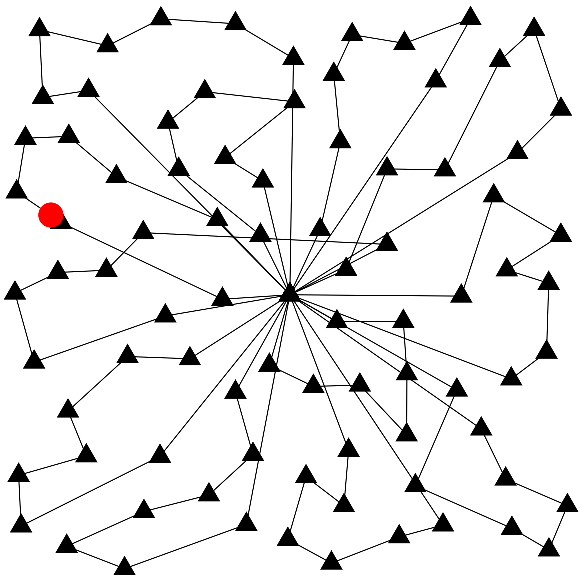
(e) p13-scenario04-HITL



(f) p13-scenario05-HITL

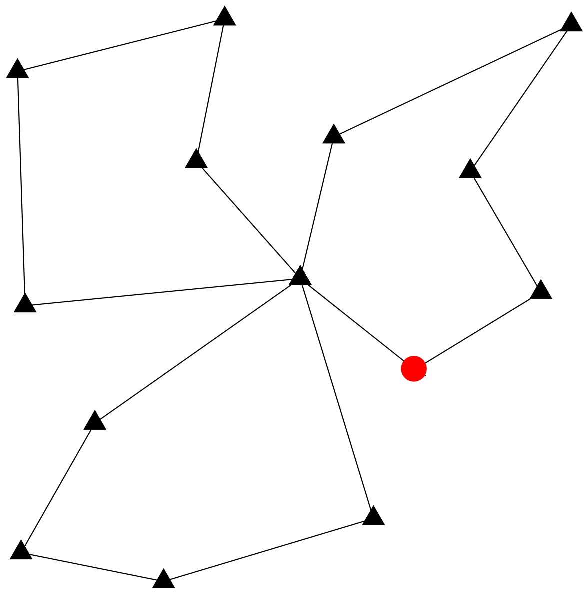


(g) p13-scenario06-HITL

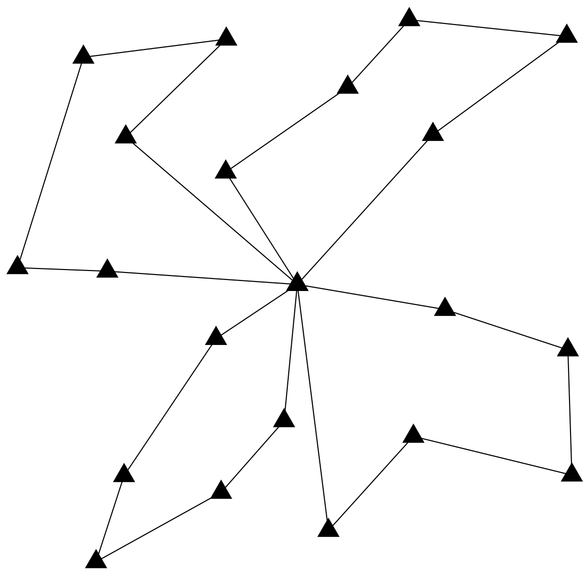


(h) p13-scenario07-HITL

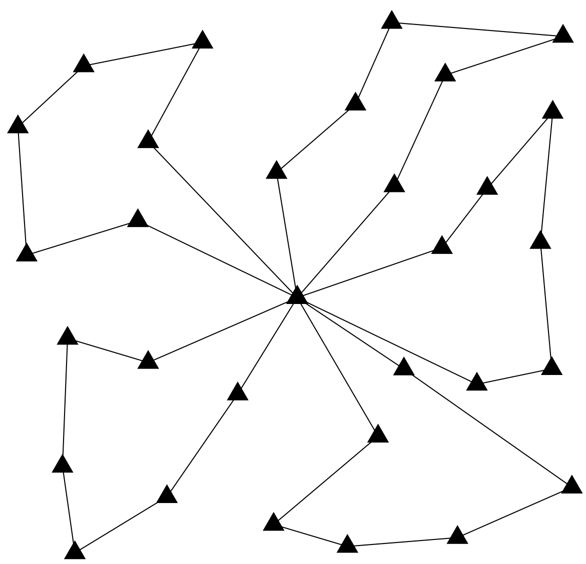
Figure H-14: Participant 13 solutions (continued)



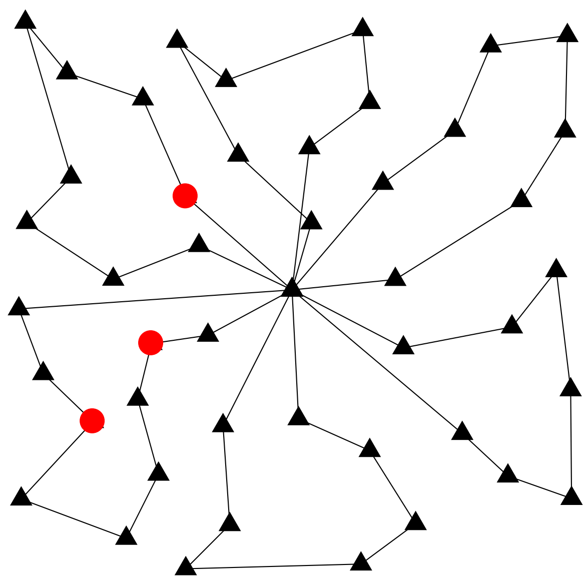
(i) p13-scenario08-HITL



(j) p13-scenario09-HITL

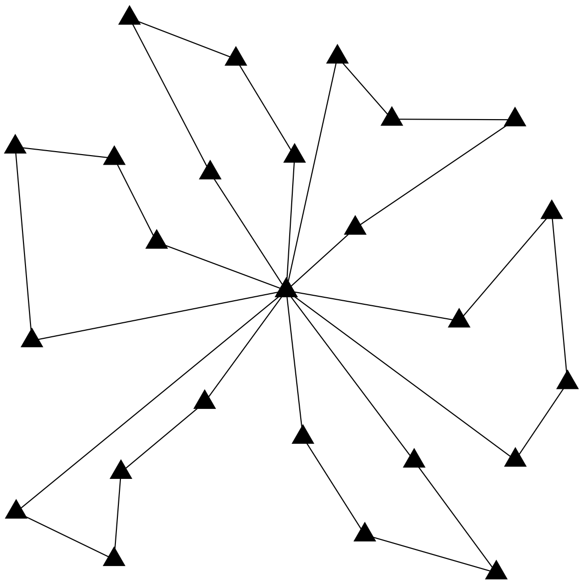


(k) p13-scenario10-HITL

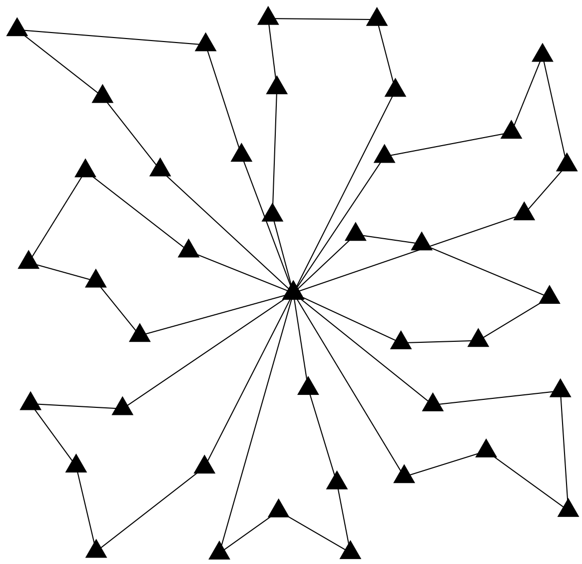


(l) p13-scenario11-HITL

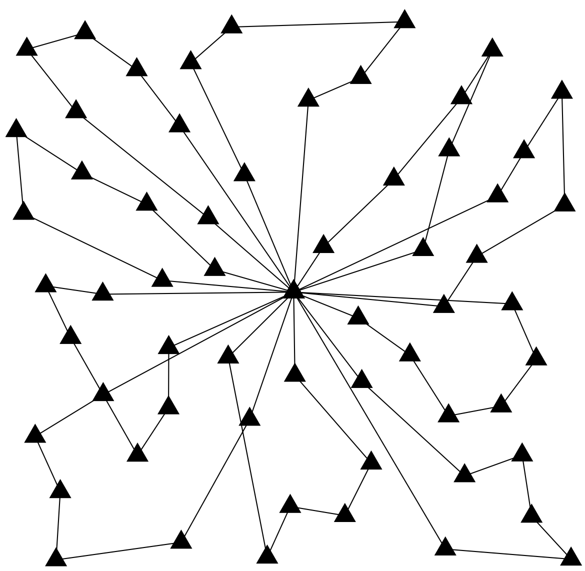
Figure H-14: Participant 13 solutions (continued)



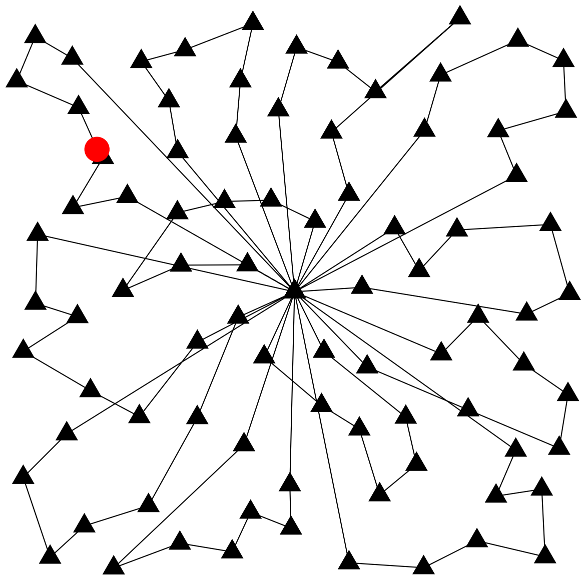
(m) p13-scenario12-HITL



(n) p13-scenario13-HITL



(o) p13-scenario14-HITL

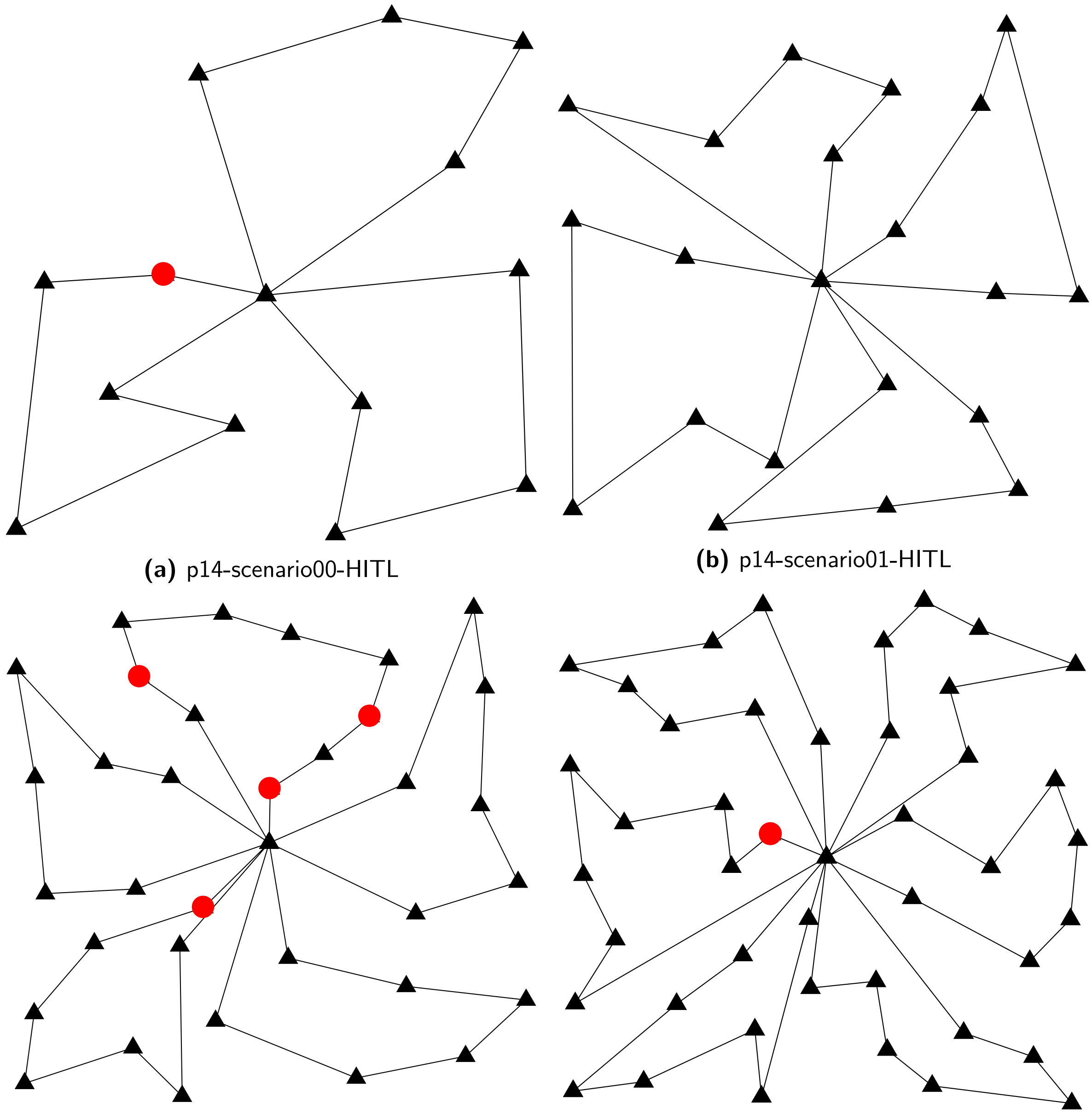


(p) p13-scenario15-HITL

Figure H-14: Participant 13 solutions (continued)

H-3-14 Participant 14

Participant 14 solutions.



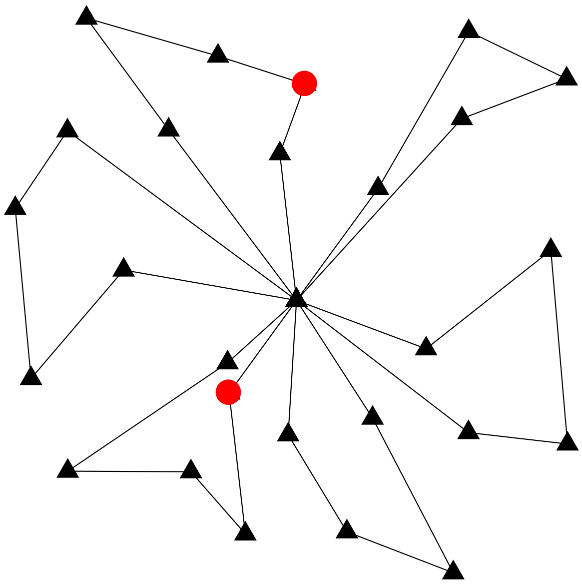
(a) p14-scenario00-HITL

(b) p14-scenario01-HITL

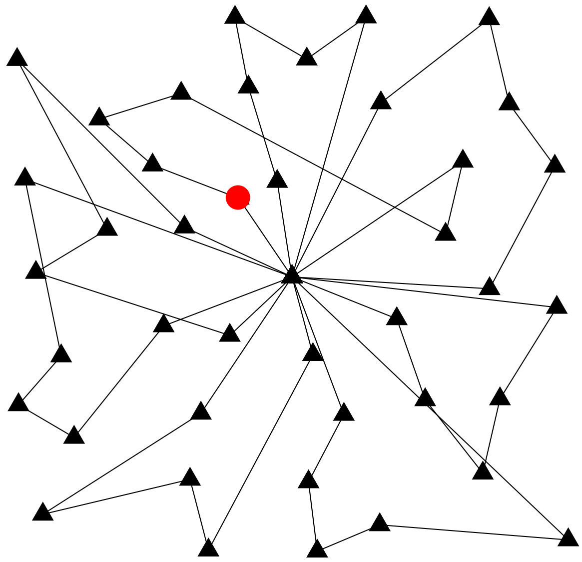
(c) p14-scenario02-HITL

(d) p14-scenario03-HITL

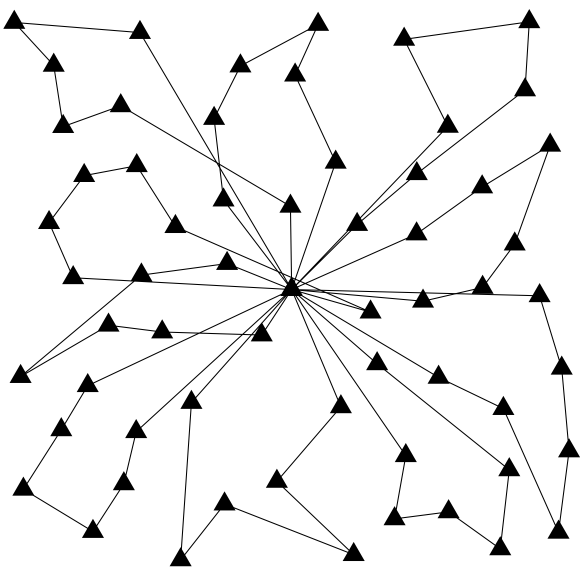
Figure H-15: Participant 14 solutions



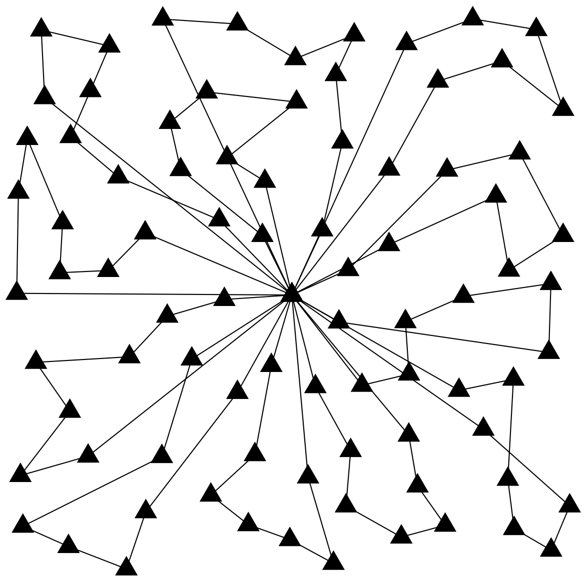
(e) p14-scenario04-HITL



(f) p14-scenario05-HITL

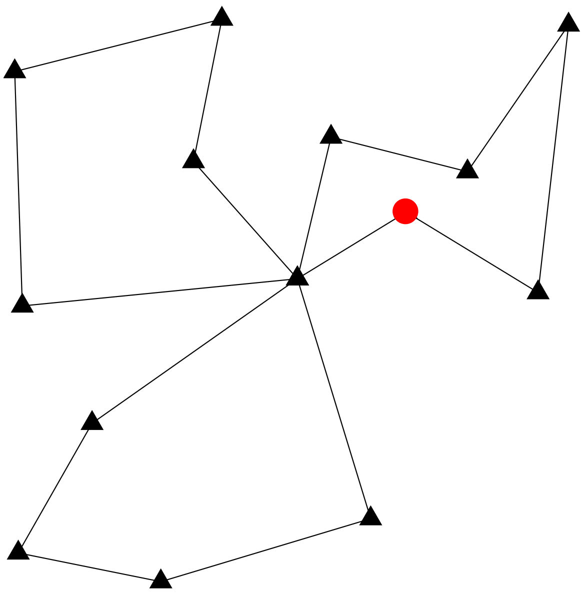


(g) p14-scenario06-HITL

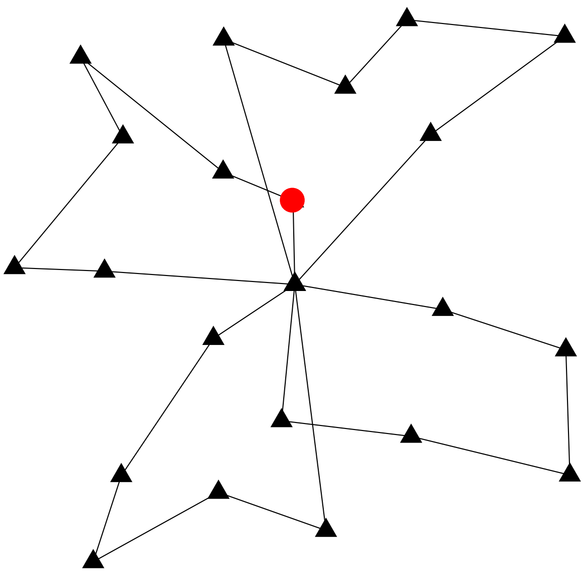


(h) p14-scenario07-HITL

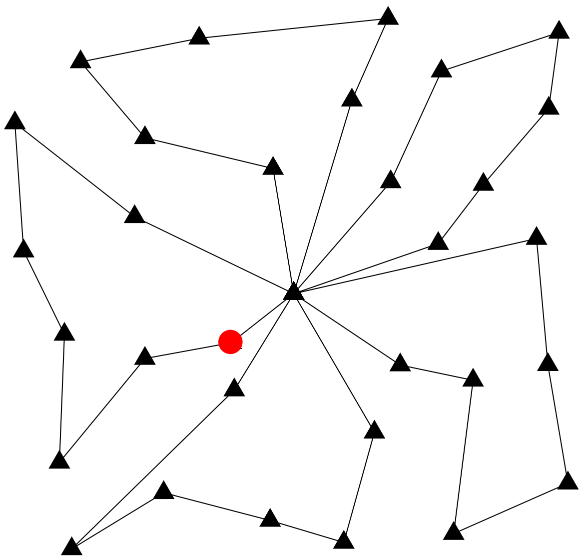
Figure H-15: Participant 14 solutions (continued)



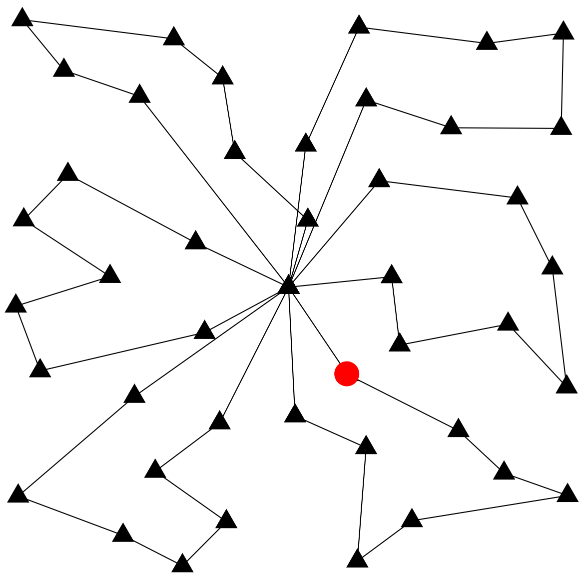
(i) p14-scenario08-HITL



(j) p14-scenario09-HITL

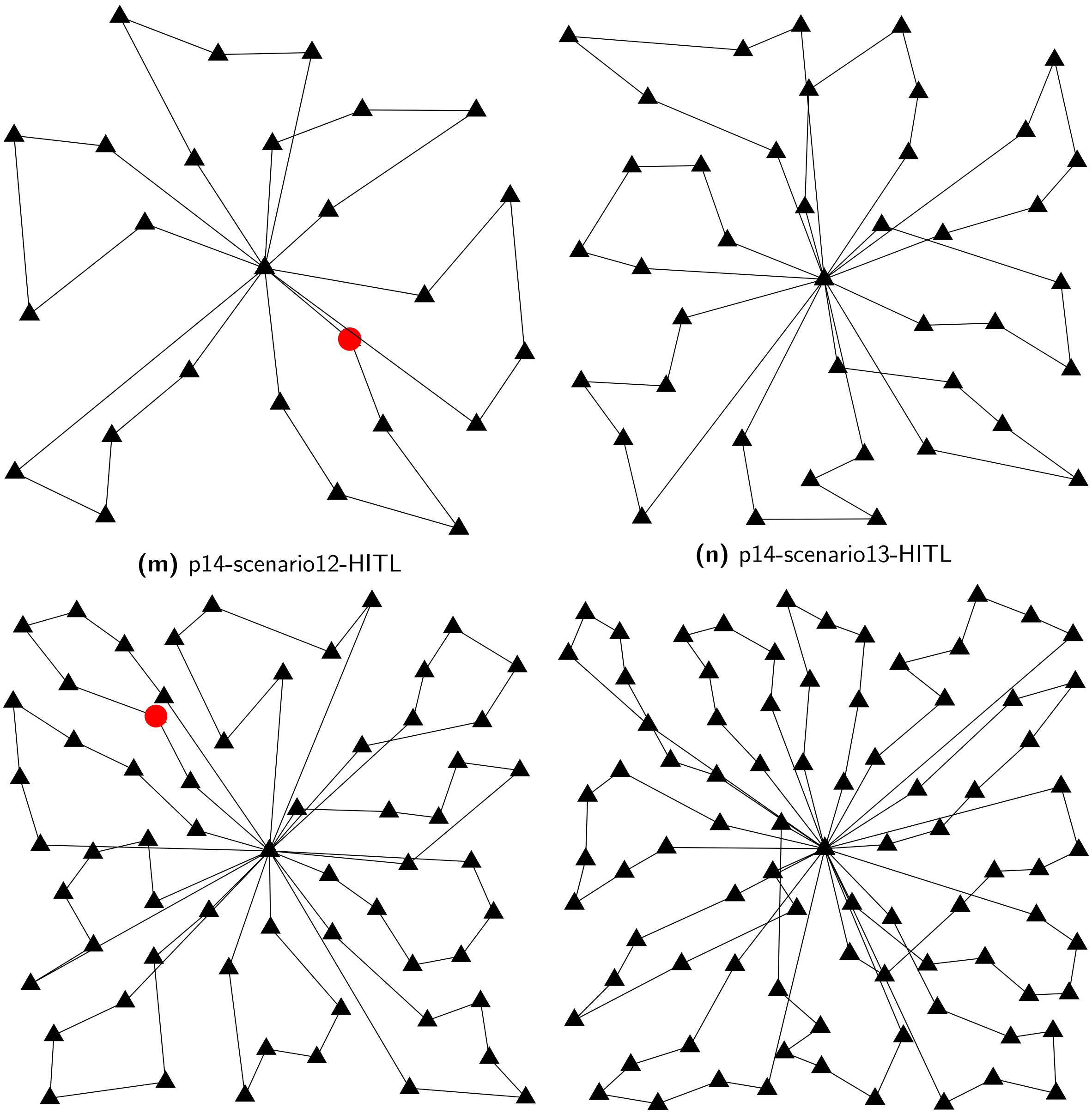


(k) p14-scenario10-HITL



(l) p14-scenario11-HITL

Figure H-15: Participant 14 solutions (continued)



(m) p14-scenario12-HITL

(n) p14-scenario13-HITL

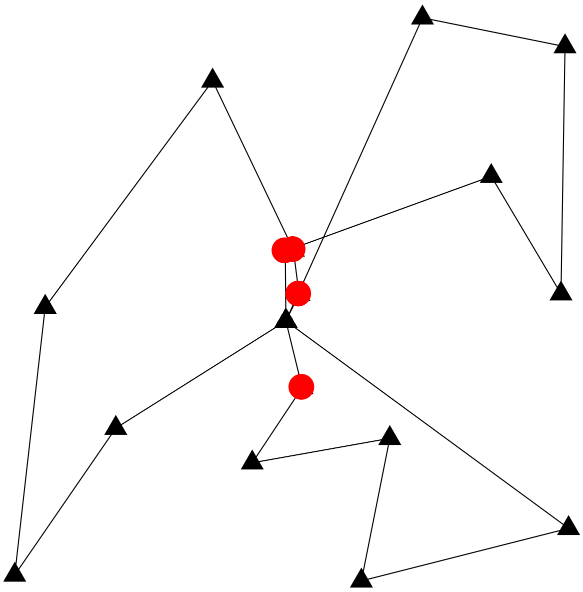
(o) p14-scenario14-HITL

(p) p14-scenario15-HITL

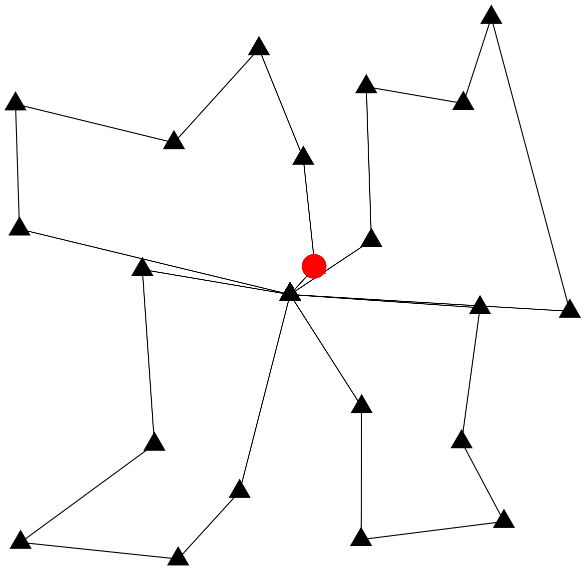
Figure H-15: Participant 14 solutions (continued)

H-3-15 Participant 15

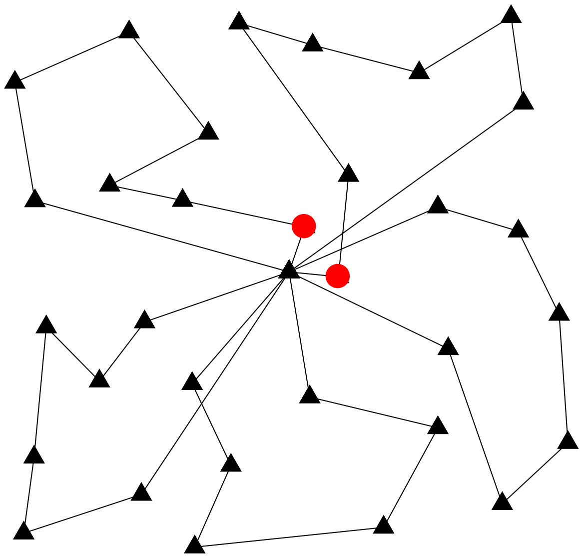
Participant 15 solutions.



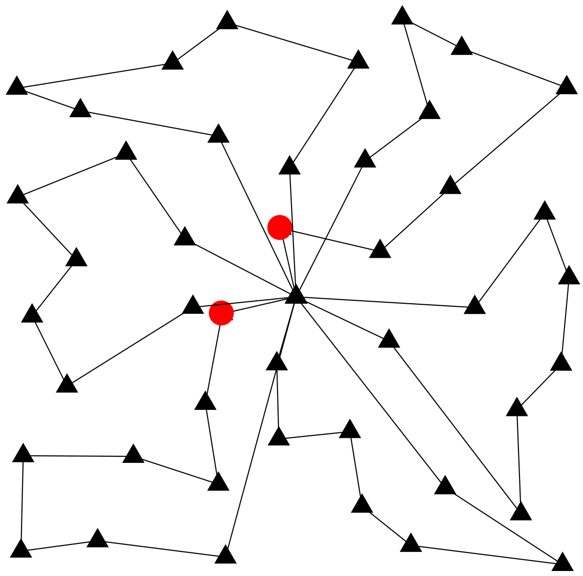
(a) p15-scenario00-HITL



(b) p15-scenario01-HITL

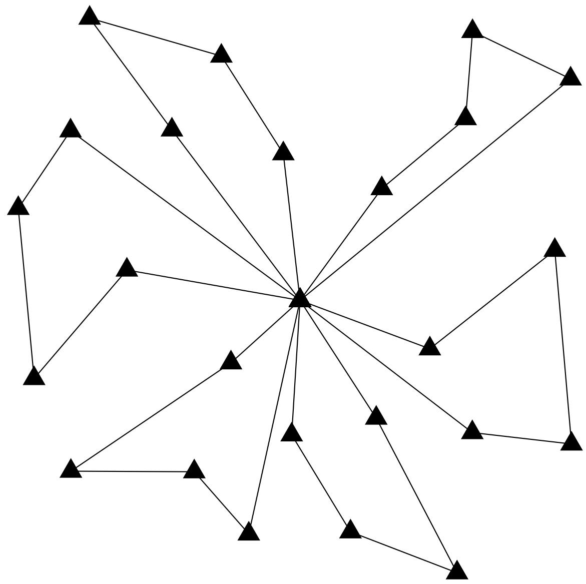


(c) p15-scenario02-HITL

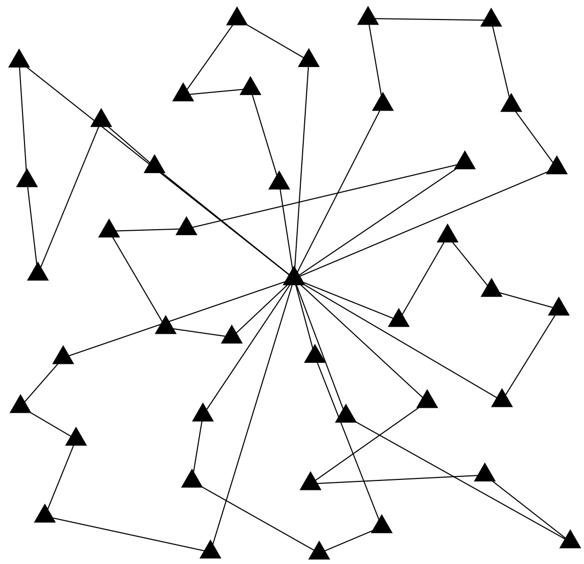


(d) p15-scenario03-HITL

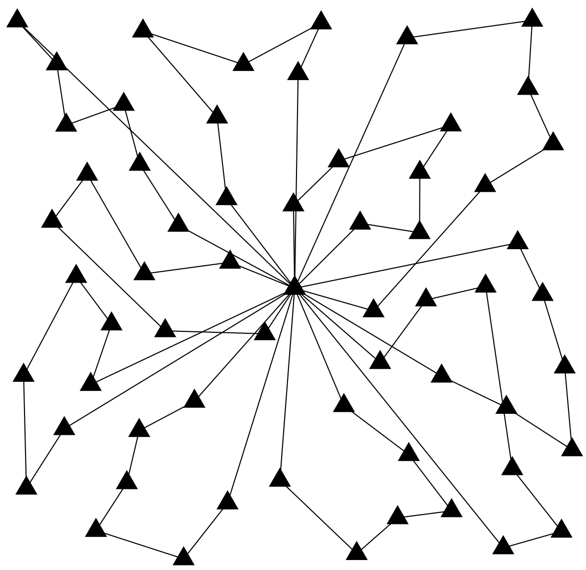
Figure H-16: Participant 15 solutions



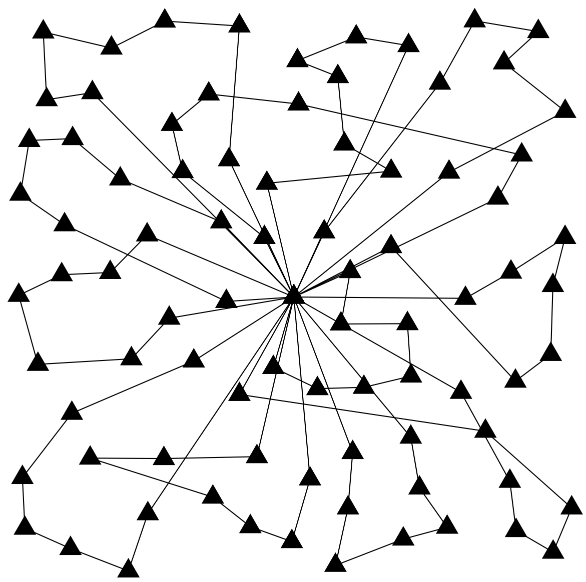
(e) p15-scenario04-HITL



(f) p15-scenario05-HITL

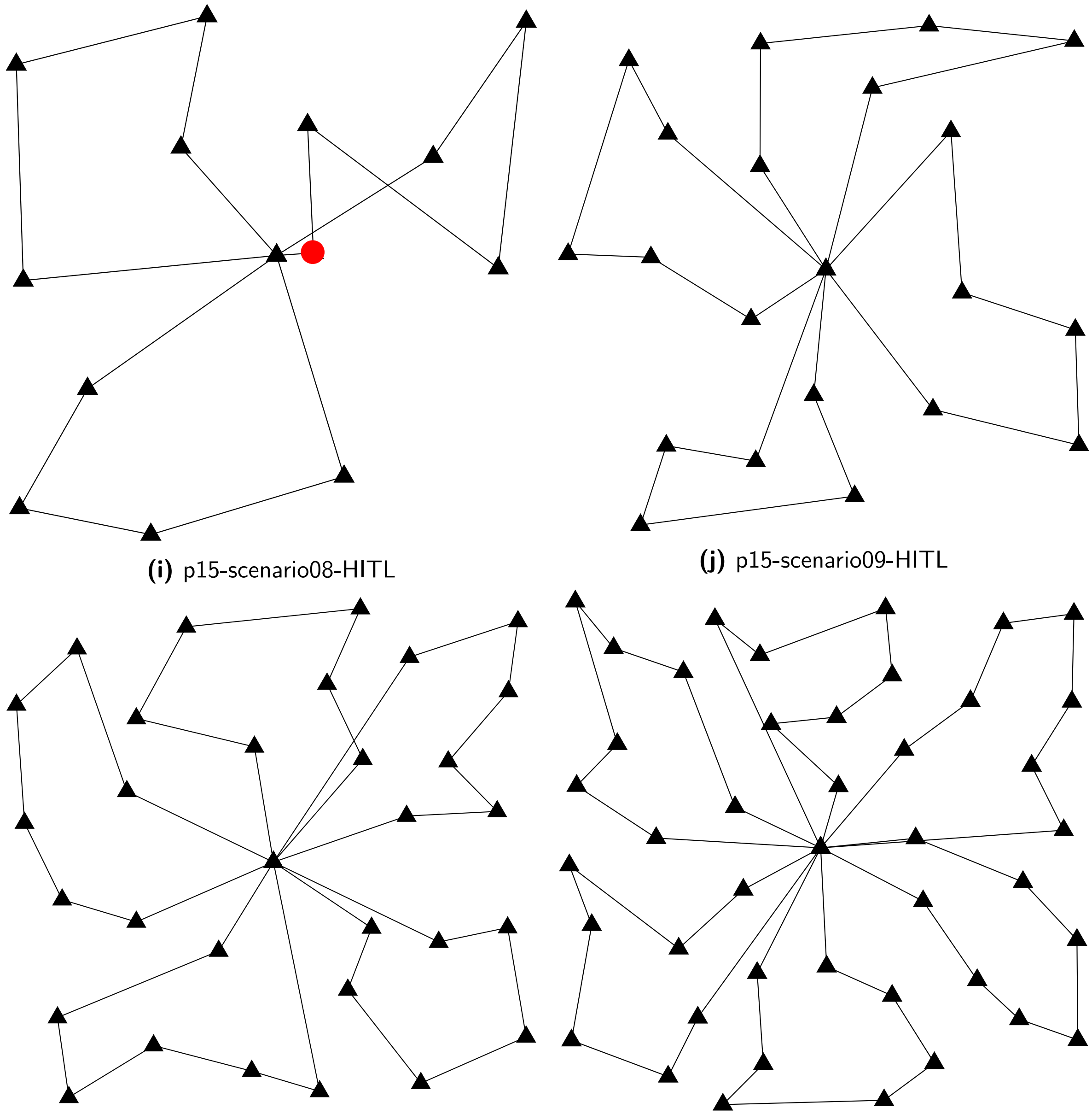


(g) p15-scenario06-HITL



(h) p15-scenario07-HITL

Figure H-16: Participant 15 solutions (continued)



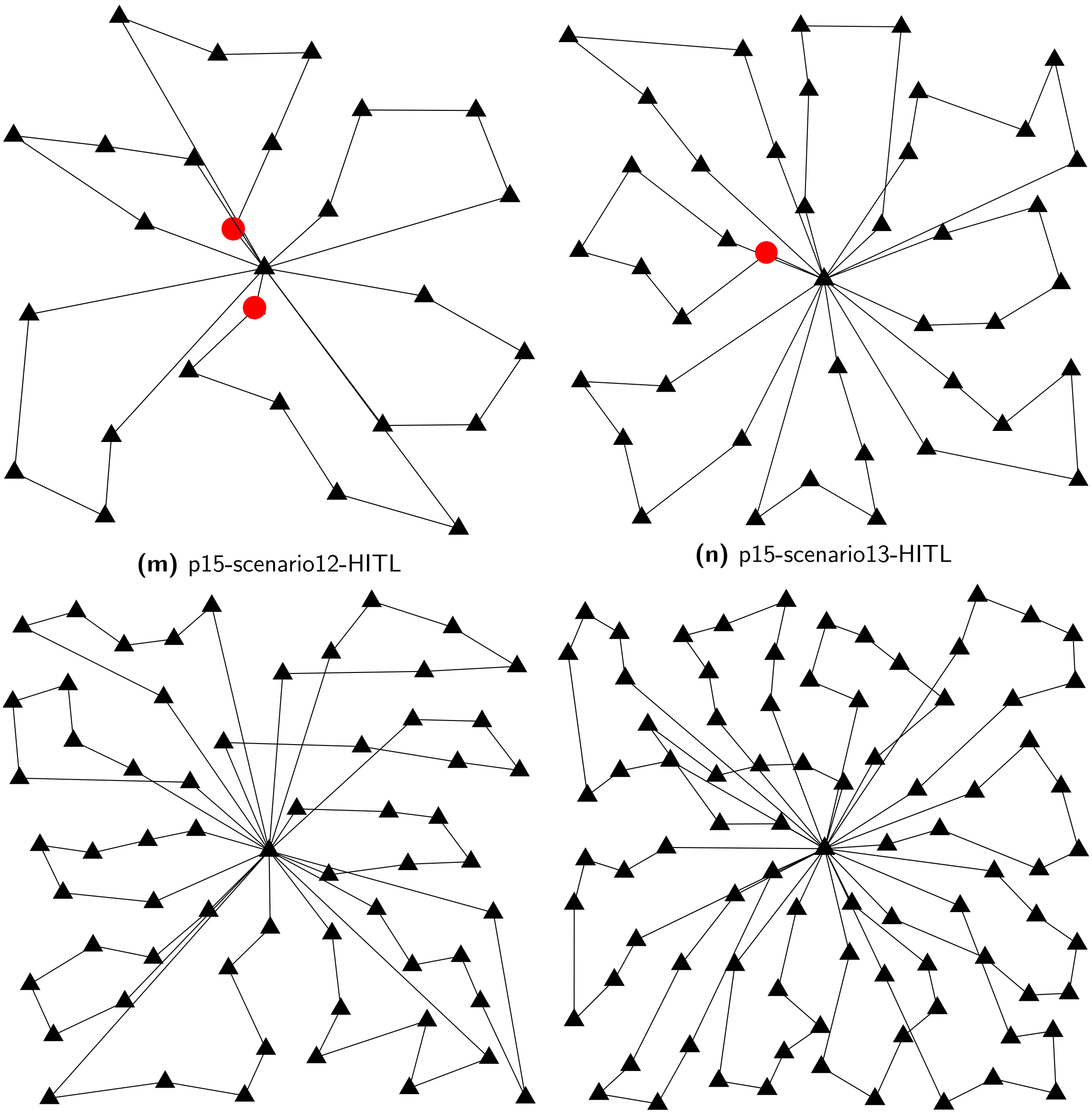
(i) p15-scenario08-HITL

(j) p15-scenario09-HITL

(k) p15-scenario10-HITL

(l) p15-scenario11-HITL

Figure H-16: Participant 15 solutions (continued)



(m) p15-scenario12-HITL

(n) p15-scenario13-HITL

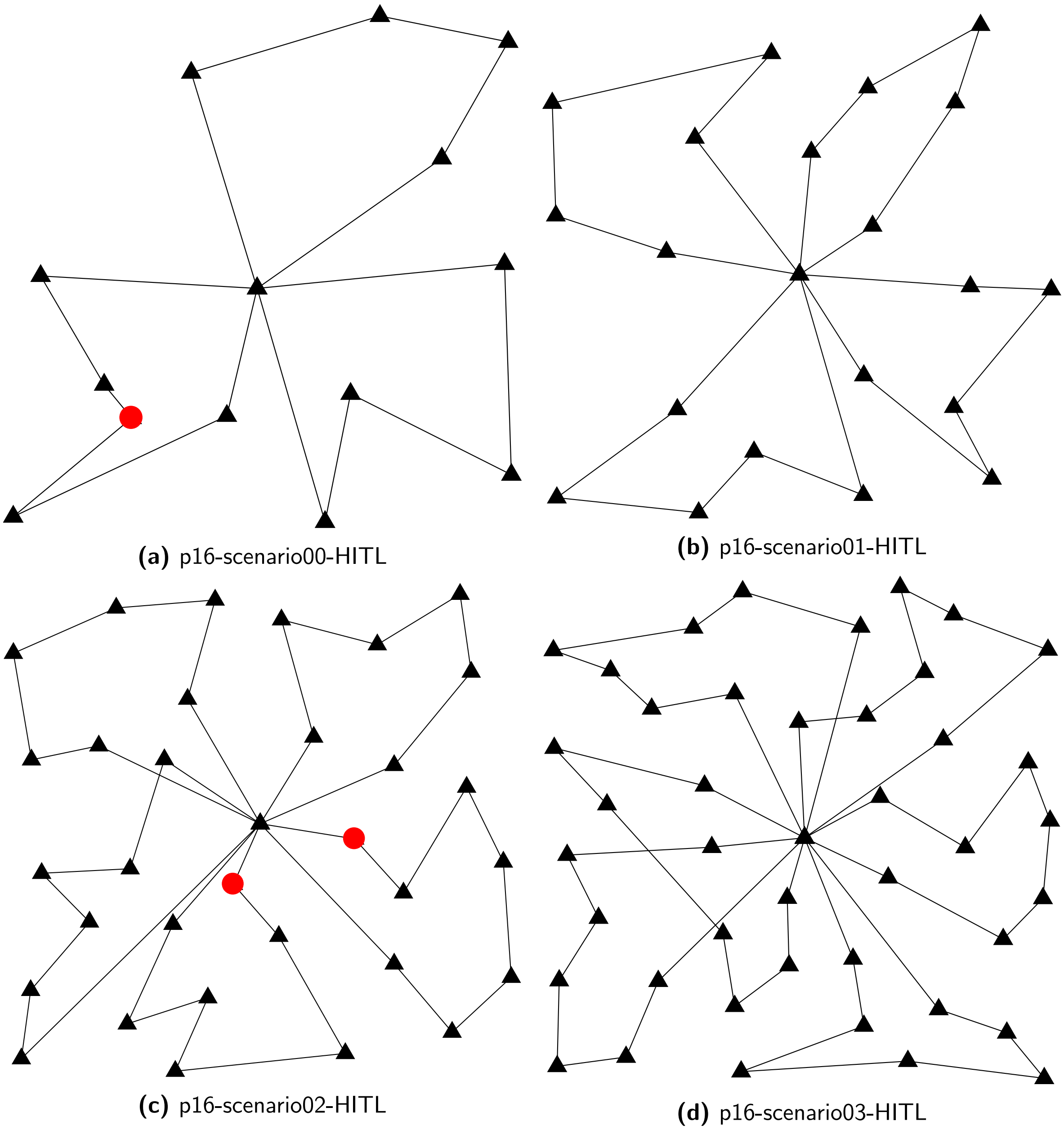
(o) p15-scenario14-HITL

(p) p15-scenario15-HITL

Figure H-16: Participant 15 solutions (continued)

H-3-16 Participant 16

Participant 16 solutions.



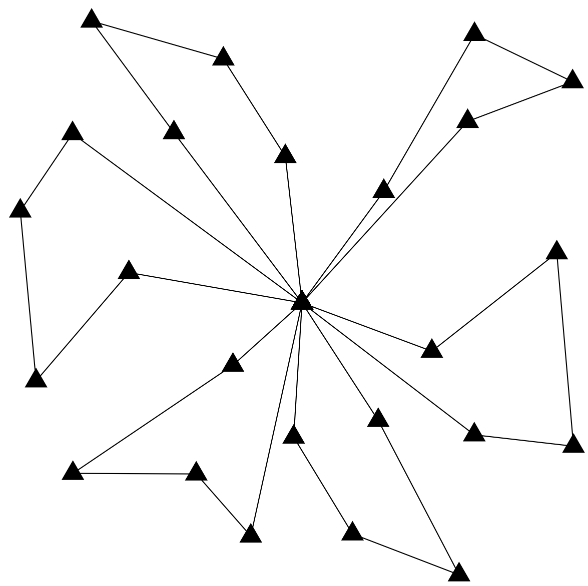
(a) p16-scenario00-HITL

(b) p16-scenario01-HITL

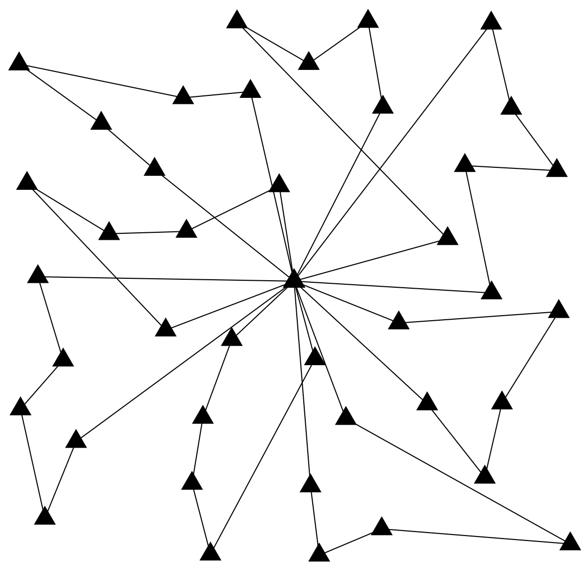
(c) p16-scenario02-HITL

(d) p16-scenario03-HITL

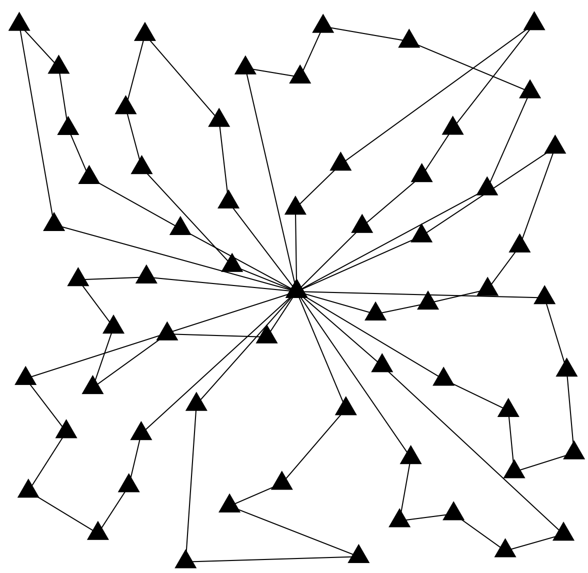
Figure H-17: Participant 16 solutions



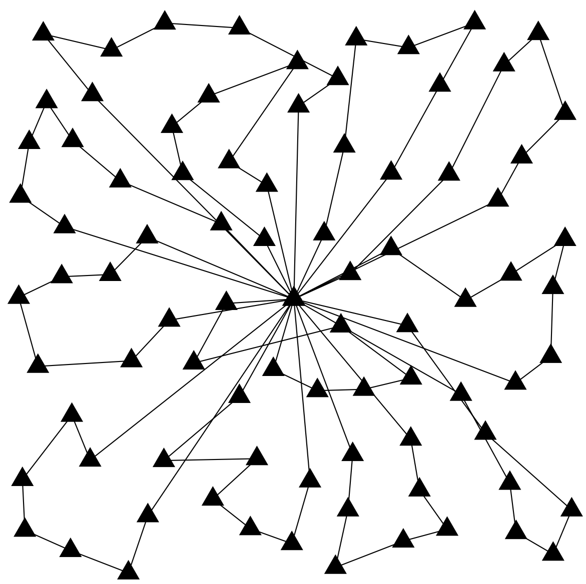
(e) p16-scenario04-HITL



(f) p16-scenario05-HITL

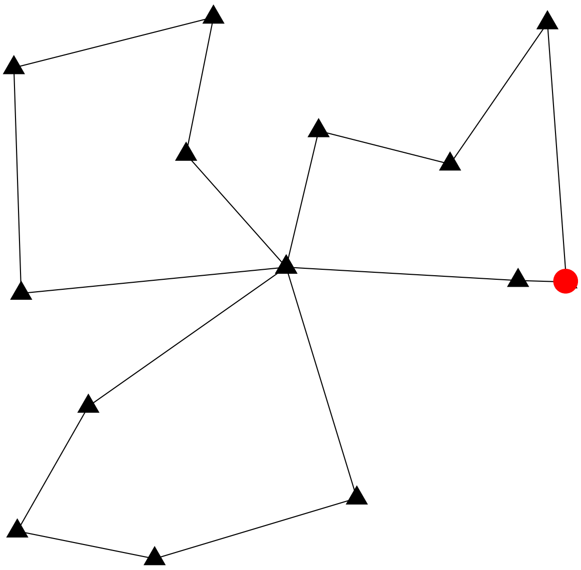


(g) p16-scenario06-HITL

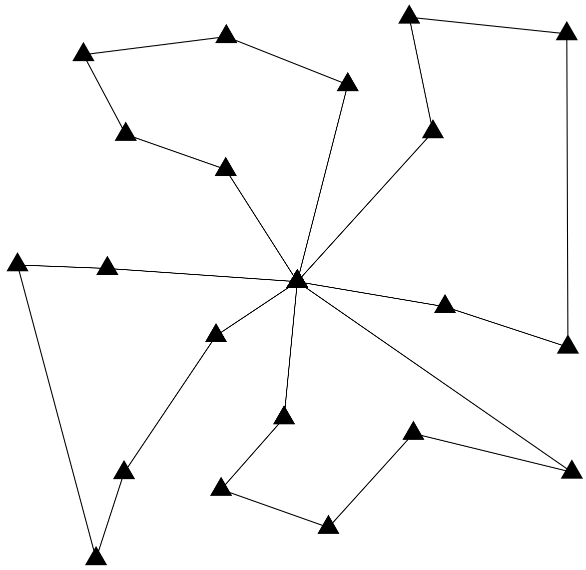


(h) p16-scenario07-HITL

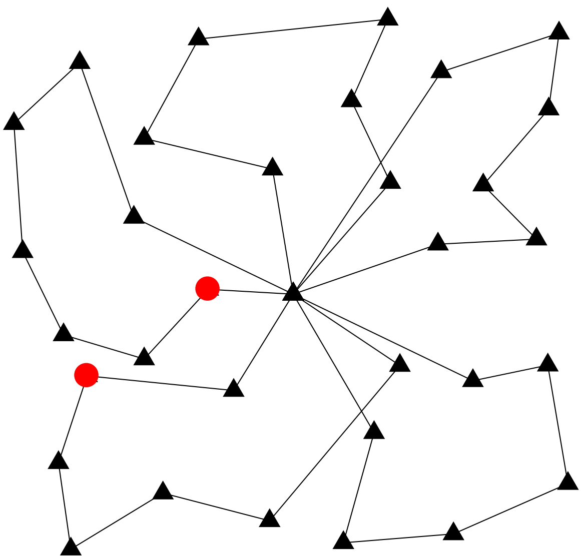
Figure H-17: Participant 16 solutions (continued)



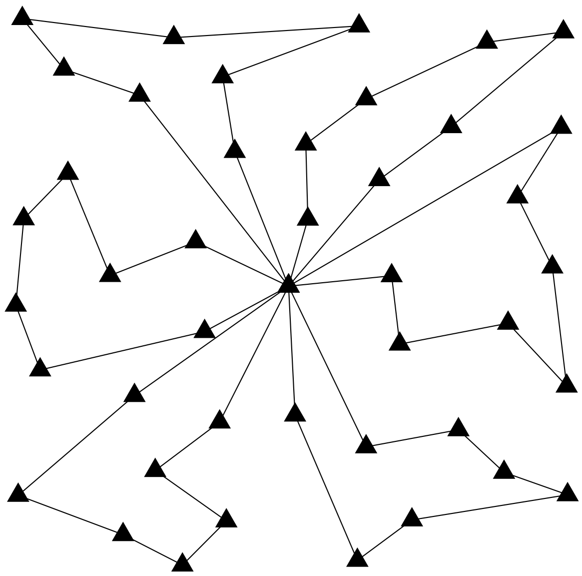
(i) p16-scenario08-HITL



(j) p16-scenario09-HITL

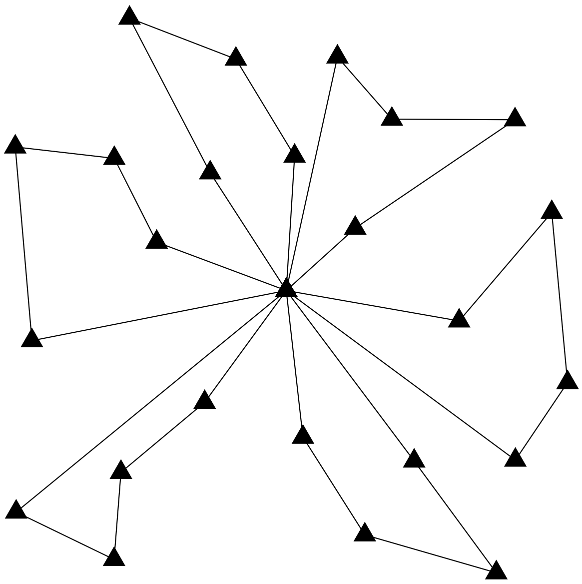


(k) p16-scenario10-HITL

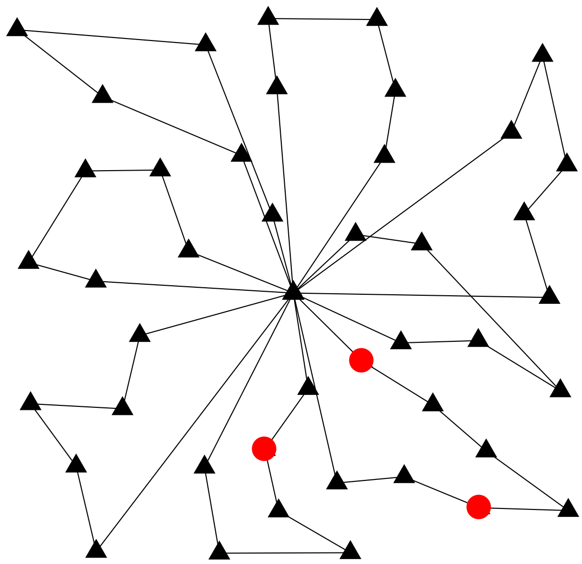


(l) p16-scenario11-HITL

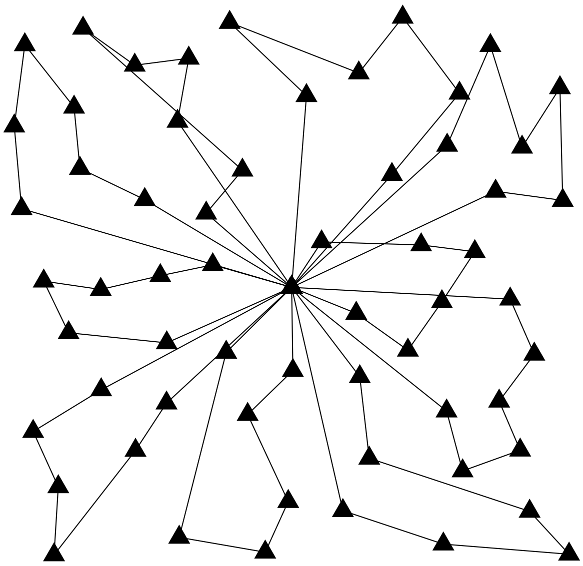
Figure H-17: Participant 16 solutions (continued)



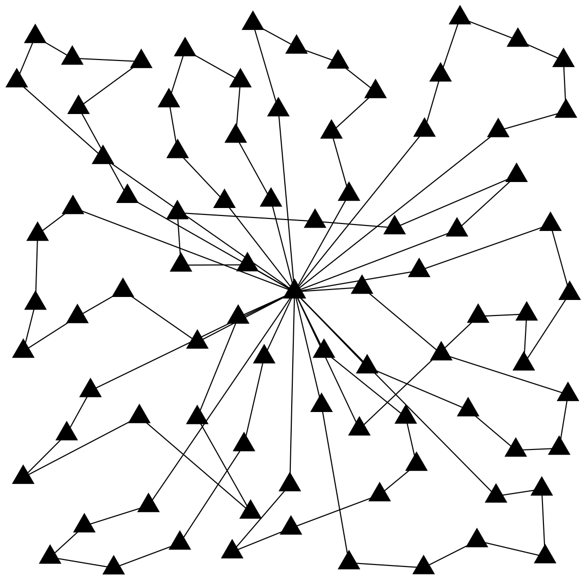
(m) p16-scenario12-HITL



(n) p16-scenario13-HITL



(o) p16-scenario14-HITL



(p) p16-scenario15-HITL

Figure H-17: Participant 16 solutions (continued)

H-4 Training Effects

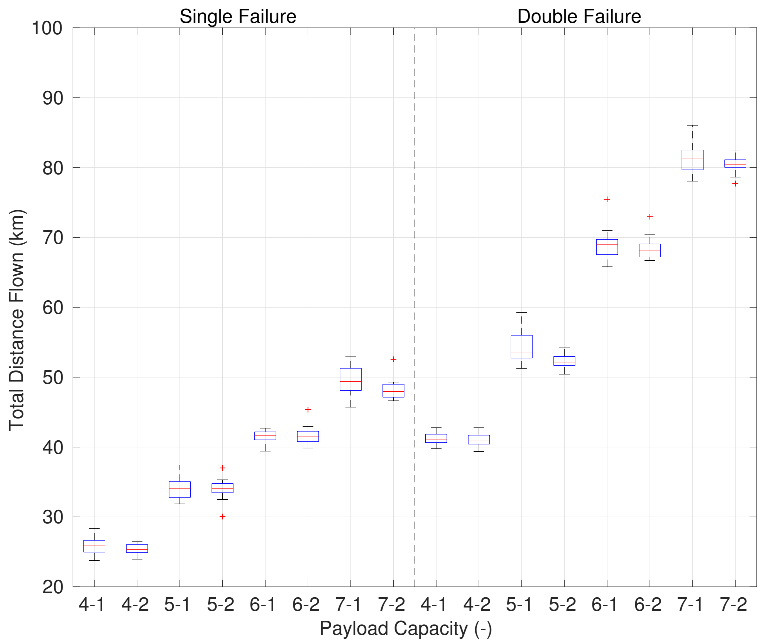


Figure H-18: Total distance flown, with first and second runs separated.

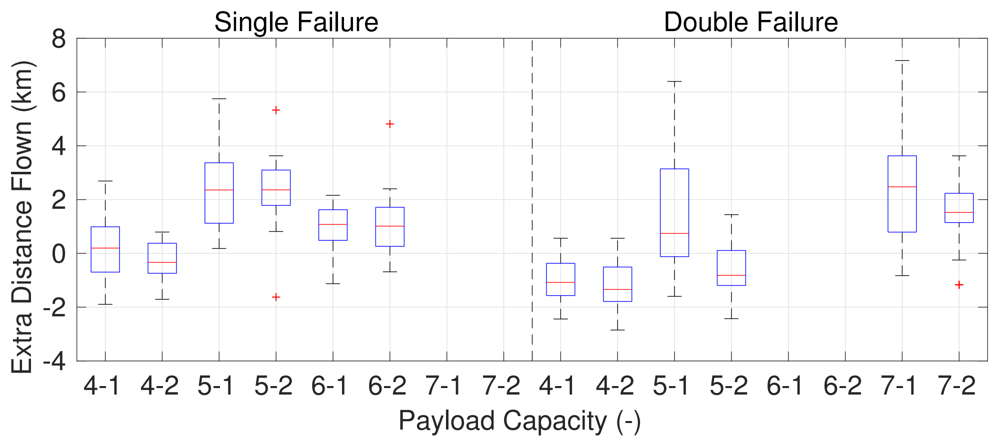


Figure H-19: Extra distance flown, with first and second runs separated.

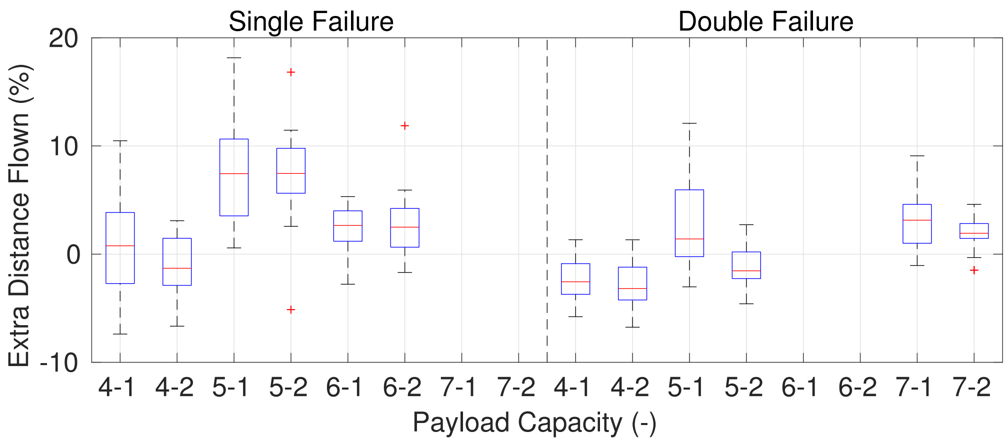


Figure H-20: Extra distance flown as a percentage of optimized solutions, with first and second runs separated.

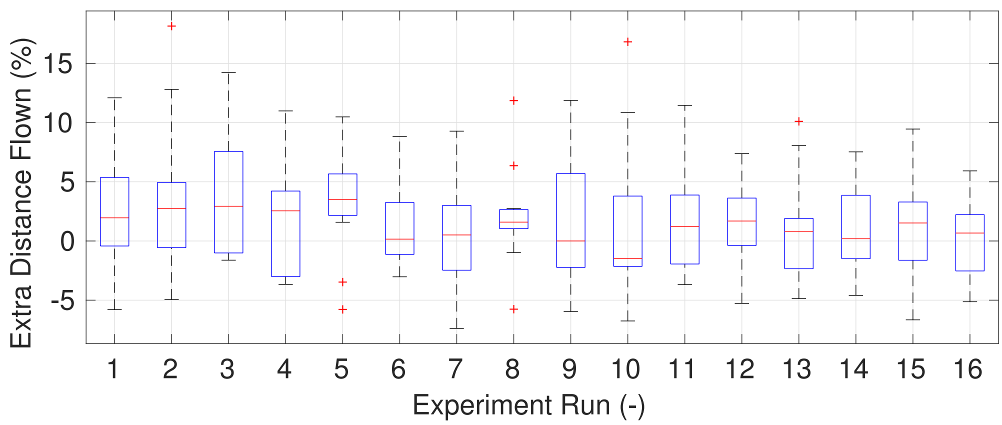


Figure H-21: Extra distance flown as a percentage of optimized solutions for each experiment run, with ordering according to experiment matrix (different experiment conditions are combined in runs, since each participant had a different scenario ordering (latin square)).

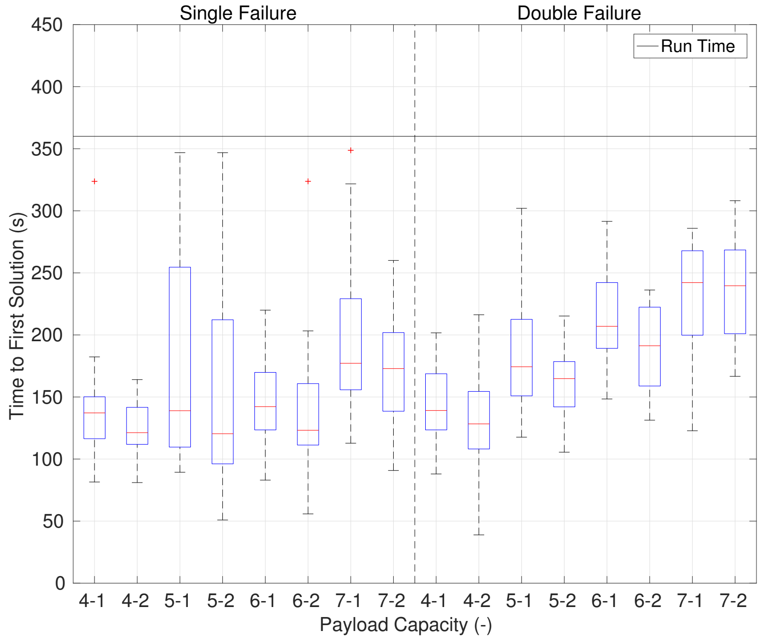


Figure H-22: Time to first solution, with first and second runs separated.

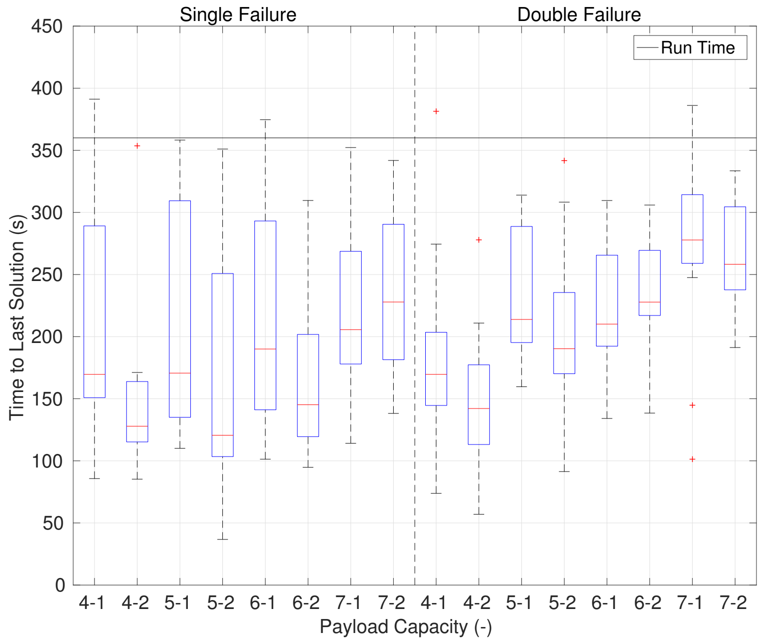
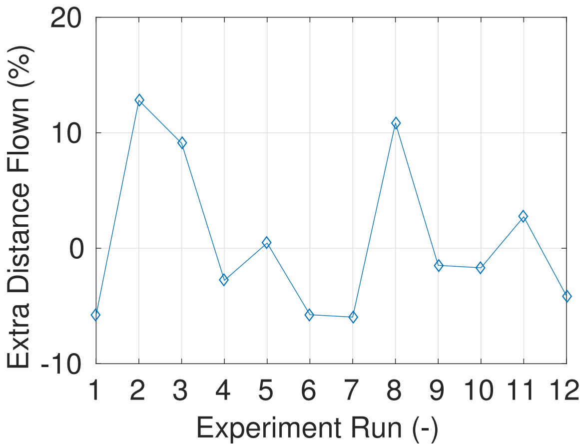
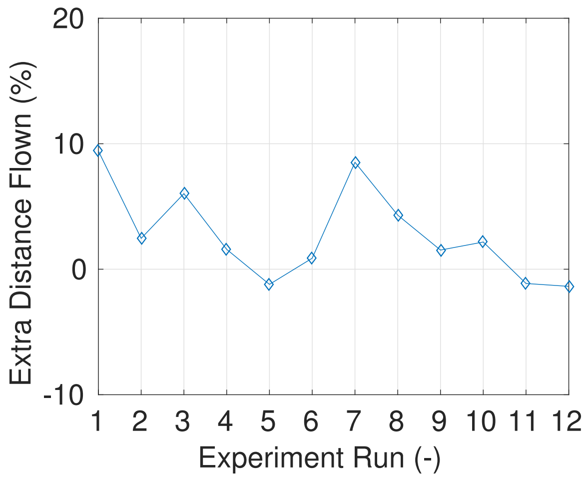


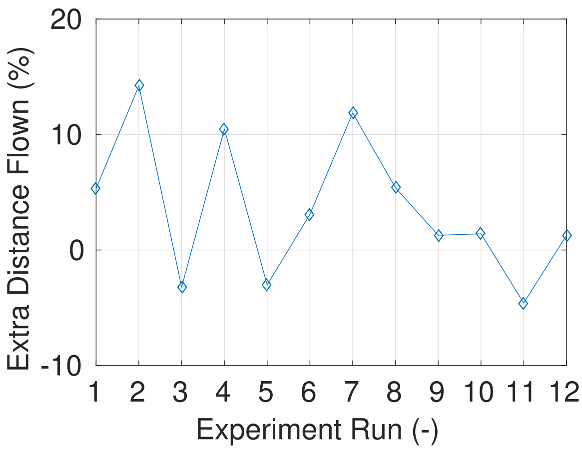
Figure H-23: Time to last solution, with first and second runs separated.



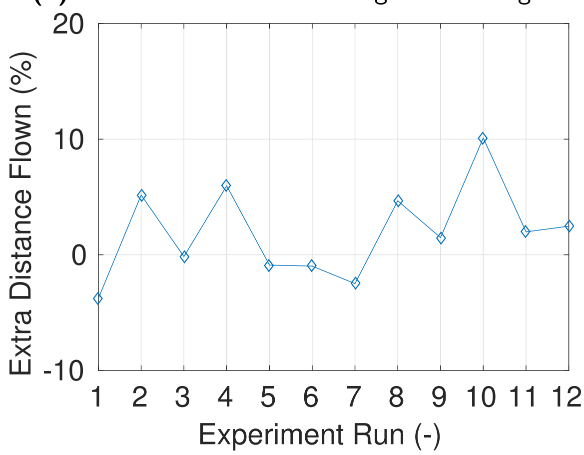
(a) distanceSumExtraPercentageFullTrainingP1



(b) distanceSumExtraPercentageFullTrainingP2

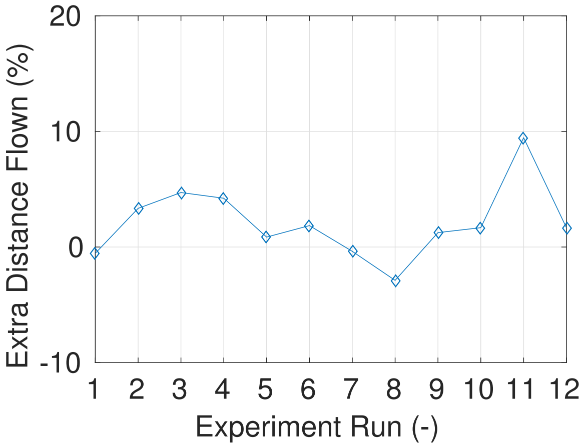


(c) distanceSumExtraPercentageFullTrainingP3

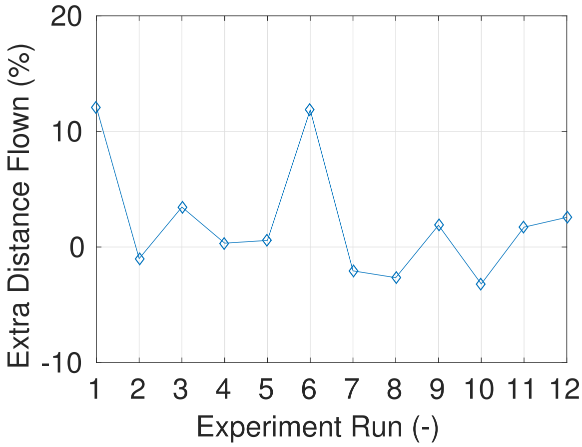


(d) distanceSumExtraPercentageFullTrainingP4

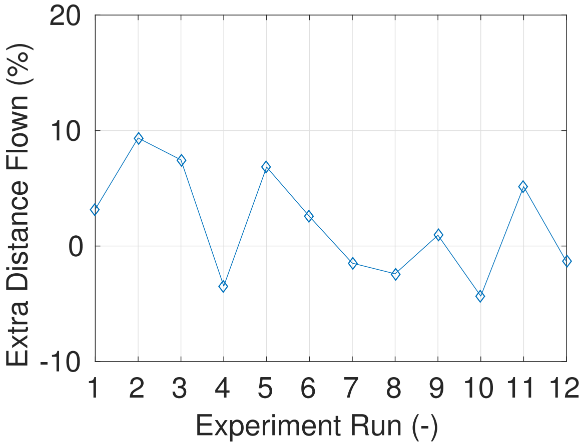
Figure H-24: Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6).



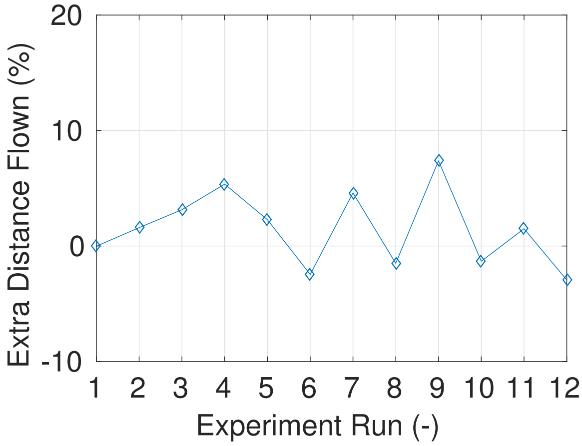
(e) distanceSumExtraPercentageFullTrainingP5



(f) distanceSumExtraPercentageFullTrainingP6

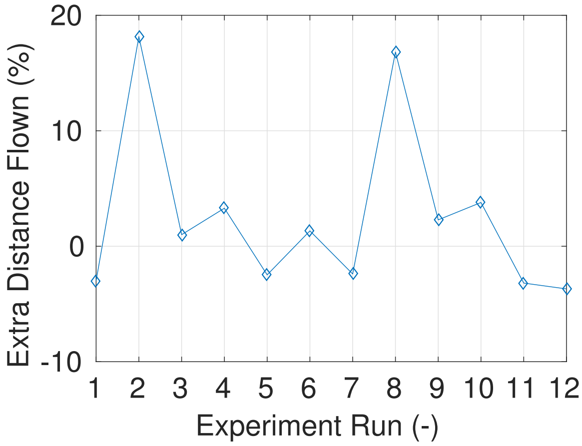


(g) distanceSumExtraPercentageFullTrainingP7

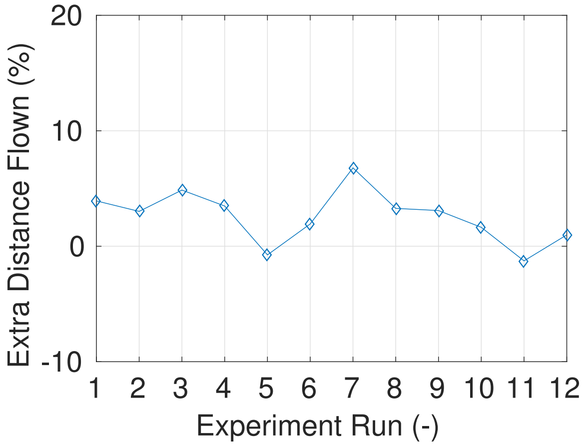


(h) distanceSumExtraPercentageFullTrainingP8

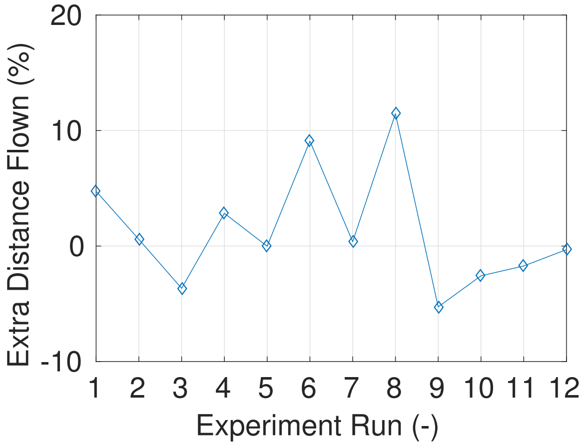
Figure H-24: Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). (continued)



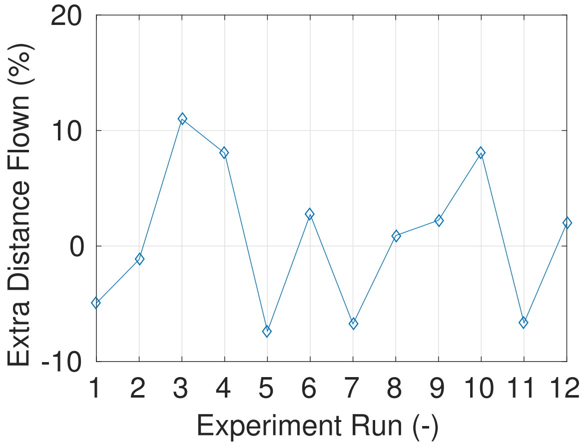
(i) distanceSumExtraPercentageFullTrainingP9



(j) distanceSumExtraPercentageFullTrainingP10

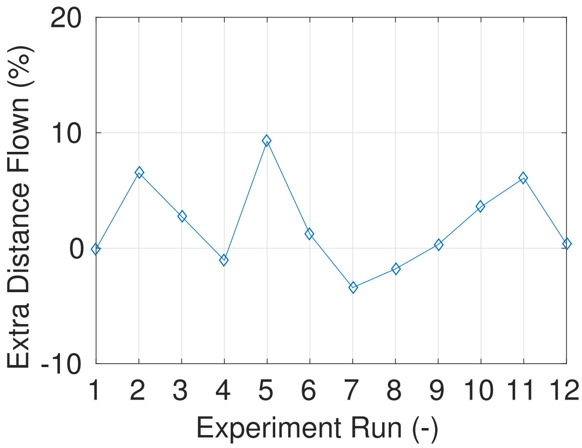


(k) distanceSumExtraPercentageFullTrainingP11

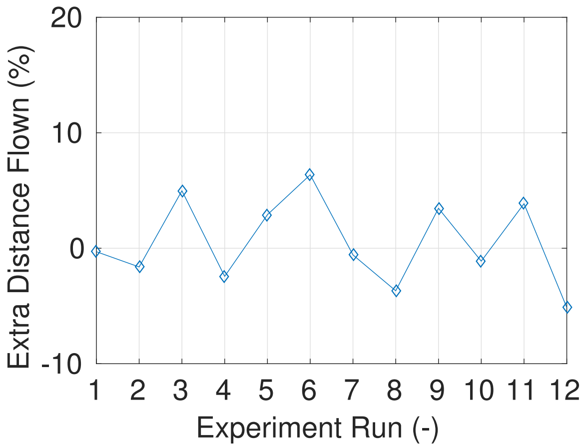


(l) distanceSumExtraPercentageFullTrainingP12

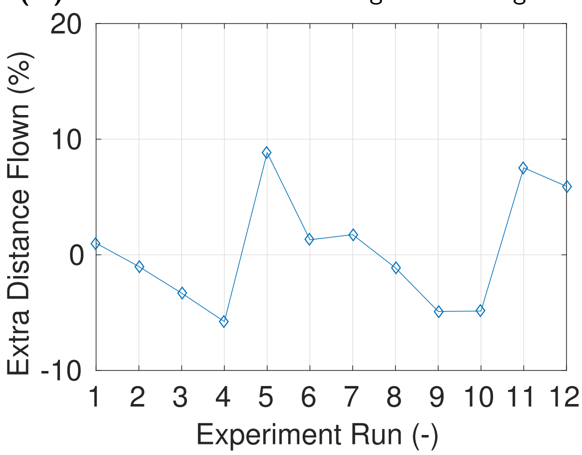
Figure H-24: Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). (continued)



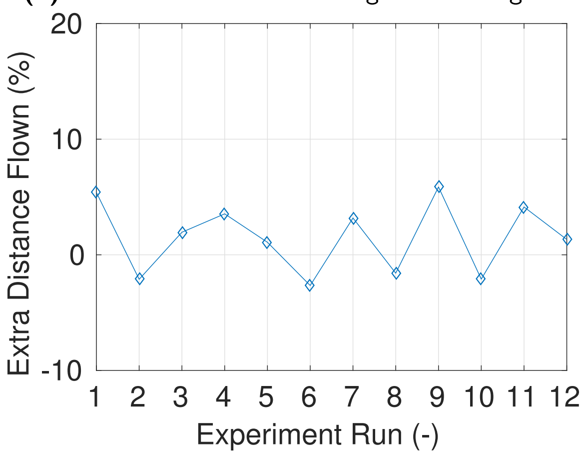
(m) distanceSumExtraPercentageFullTrainingP13



(n) distanceSumExtraPercentageFullTrainingP14



(o) distanceSumExtraPercentageFullTrainingP15



(p) distanceSumExtraPercentageFullTrainingP16

Figure H-24: Extra distance flown as a percentage of optimized solutions per participant for each experiment run, with ordering according to experiment matrix (only 12 runs, since no optimized solution was obtained for F1P7 and F2P6). (continued)

H-5 Post-Survey Participant Feedback

This section lists all comments obtained from participants in the post-survey. The following questions were asked:

1. How do you assess the usefulness and the functionality of the map view? Please provide examples in your elaboration.
2. How do you assess the usefulness and the functionality of the timeline view? Please provide examples in your elaboration.
3. How do you assess the usefulness and the functionality of the payload view? Please provide examples in your elaboration.
4. How do you assess the usefulness and clarity of the color use in the display? Please provide examples in your elaboration.
5. How would you solve the scenario depicted below (high-level description)? NOTE: each vehicle only has one spare payload item on board.
6. How would you solve the scenario depicted below (high-level description)? NOTE: each vehicle only has one spare payload item on board.
7. Do you have any other comments or suggestions with respect to the interface or the experiment?

The figures that relate to question 5 and 6, are depicted in Figure H-25 and Figure H-26, respectively. All participant comments to the aforementioned questions can be found in Table H-2, Table H-3, Table H-4, Table H-5, Table H-6, Table H-7 and Table H-8.

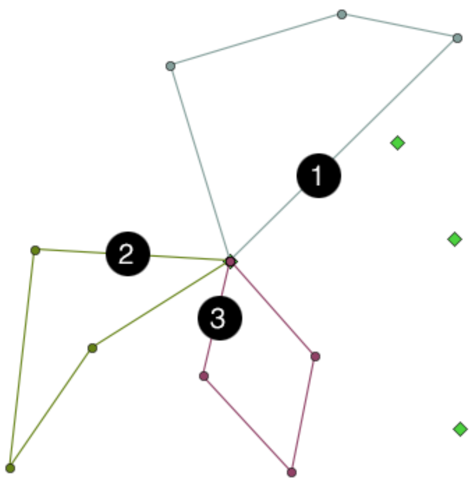


Figure H-25: Figure corresponding to post-survey question 5.

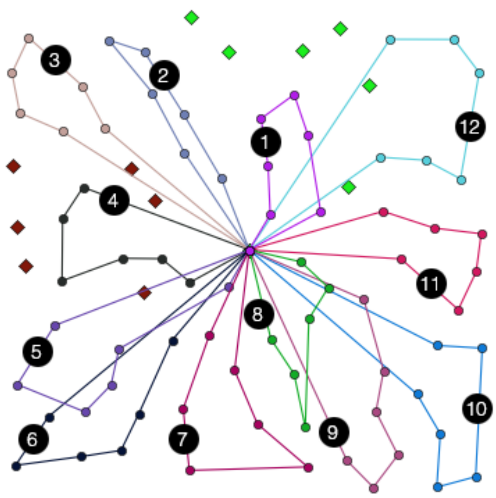


Figure H-26: Figure corresponding to post-survey question 6.

Table H-2: Q1: How do you assess the usefulness and the functionality of the map view? Please provide examples in your elaboration.

Participant	Comment
P1	Very useful, especially the green and red areas are useful during conflict solving. Also the connections with the other displays are useful, especially when there is a timing issue.
P2	The map view gives a simple but clear overview of the situation.
P3	UAV's, depot, waypoints etc. are clear. Misclicks occasionally happen because of the small (UAV) clicking areas.
P4	Was actually the most useful screen, 90% of the task was probably based on the information there. I barely looked at the payload view since the color of the aircraft already indicated its payload capacity.
P5	it was very useful as it projected the main information I used for the planning. The other (payload and timeline view) were only reviewed once the map showed red alerts.
P6	Really useful & functional. Especially the red ellipses for helping solving depot arrival time issues were quite useful. Had to get used to how to delete certain waypoints etc, but in the end it was quite clear
P7	Pretty useful I would say. Kind of hard to get an insight on the whereabouts of the drones without it. Serious though, the range indicator is very useful. I did not like the pre-programmed routes however. The red icons for drones were useful too and prevented the need to check the timeline if everything was ok.
P8	The map view was the most useful view. The use of colors to indicate different situations is helpful, especially when constraints are violated. It was hard to see in which direction the drones would start to fly.
P9	The map view is very useful especially with the green areas that you can reach with each UAV. If the depot constraint is violated you get notified on that via the map view, which I used quite a few times, mostly in the scenarios with a lot of vehicles. The map view has been my main tool for solving the problems in all scenarios. The interface contains a lot of useful information like the coloring of the aircraft. The check if they all are orange gives you a very quick check if all aircraft are delivering all their payload. This is a useful thing in scenarios like these where failure of some UAVs means that the others have to deliver all their payload. The controls are convenient after some time getting used to them.
P10	Most important part, main source of information. When the UAV(s) stopped working, it gave a quick overview of which other UAVs were passing by the now unused points. This allows for an easy starting point to plan the routes.

P11	The map view is the best part about the setup, and it is the main part I used throughout the test. It gives a very clear overview of what's going on, especially the added flight range, which is a very helpful tool in finding the available routes. The only comment I have on the user interface is the mandatory selection of a plane before adjusting the route, rather than instantly selecting a route and adjusting it. I lost some time respecting that rule, when I was completely sucked in and wanted to solve the routes as fast as possible. Though I understand this is done for structure and I can imagine that an interface where the flight routes can be directly adjusted could be confusing for some people.
P12	The map view was the display I primarily used to route the UAVs. The constraints are intuitively color coded and give an good overview of the range and amount of way-points it can visit. I used the display to look for different options taking neighboring UAVs and their planned trajectories into account.
P13	Good, sometimes you need to click very accurately leading to incorrect clicks. Colors are well chosen, but you had a very good advisor here ;) I got most of my information from the map view, for example payload full I looked at the red waypoints, not the payload view itself.
P14	Hard to define the usefulness of the display as a whole, as the experiment/simulation revolves around it. Considering the individual components: the color coding took some getting used to, and sometimes it was difficult to see which UAV still had some payload left. The visualizations of remaining flight time and conflicts in arrival time were very nice, this made judgment making somewhat easier.
P15	Very useful. It makes it clear where the uavs are going and what the possibilities are. The green background resonates well with the colors of the paths (active and inactive)
P16	Very useful, nearly all I used.

Table H-3: Q2: How do you assess the usefulness and the functionality of the timeline view? Please provide examples in your elaboration.

Participant	Comment
P1	The timeline is good to determine your overall performance and nice to have to determine which UAV should be delayed. Also the highlighting, i.e. connections to the other screen is very nice.
P2	The timeline view was clear and useful. You can easily see where arrival conflicts arise, and I always used it after planning all of my routes.
P3	Very useful to see the number of UAV's at depot. Not immediately clear what the colored area means (UAV at dept), also not which area corresponds to which UAV so not immediately visible on which UAV you should take action.
P4	Quite useful for resolving conflicts. However, information about how large was the overlap was available in the map view, so I only used the timeline view to decide which aircraft I should delay.
P5	I personally only reviewed the timeline view once I received an alert from the map view. Re-routing is an easier task then re-planning to reach all customers. Therefore the timeline view had the least priority

P6	Didn't use the timeline that much, except when there were conflicts regarding depot arrival times. In that case the hoovering over the timeline option to see which aircraft were in conflict was nice. (Especially since the map view did not always provide all conflicting aircraft)
P7	Useful if time constraints did form a problem. It was difficult however to see the solution space - i.e. I had to randomly play around with diversions before timing issues were solved.
P8	In case of multiple drones arriving at the depot at the same time (violating the maximum amount at the same time possible) it was hard to see which drones were involved in an 'arrival conflict'. It took some time to figure it out cause it was not directly visible on the map view which drones were involved. In case of no conflict, I didn't pay much attention to this view.
P9	The timeline view is something I only used when I had all dropoff points covered. After solving the routes I used it for monitoring if nothing weird happened, which did happen once. If there are depot overloads then it provided a quick check to see which vehicle needed a change in its route.
P10	Useful, shows when the problems occur. Does have some flaws. In one of the experiments, the timeline view suddenly gave problems after one of the UAVs passed a waypoint.
P11	I did not utilize the timeline that often, only in cases when the flight schedule contradicted one another. Depending on the position of the blocks I could easily figure out when I should delay a certain fight, or try to reduce its flight time.
P12	It was useful when depot conflicts occurred. As the UAVs are easily rerouted or delayed by adding or changing way-points, my primary focus was to visit all way-points with an efficient route. Afterwards I looked at the timeline view if there was a conflict and if it was easily solved by delaying a UAV, which could easily be solved.
P13	Fine, a bit confusing. Maybe each time blob could have gotten an outline, so that you could see the different UAVs
P14	It was okay. I only used it once I noticed there were arrival conflicts, and then only as an easy selection tool for finding the conflicting aircraft. To solve the issue, I used the map view. The preview of remaining flight time was useless in my view. Also, once 3 UAVs arrived at the same time, it got really crowded.
P15	Good be better. It is hard to distinguish the individual uavs. This makes it tricky in finding an optimal timeline.
P16	Useful when you placed all the routes and then had to assess whether the depot would be full or not. Didnt use it when placing the routes it self

Table H-4: Q3: How do you assess the usefulness and the functionality of the payload view? Please provide examples in your elaboration.

Participant	Comment
P1	The payload screen becomes not needed due to the color coding of the plane. They basically tell you everything you need to know. Only the amount of free spots is nice to know.

P2	I only used the payload view when I was unsure whether a UAV had more than one spare payload item. I think this view is redundant if the use of colors in the map view is optimized.
P3	Important view because it gives the max and available payload. It is tucked away in the corner of the big screen though making it less functional.
P4	Good to know what is the maximum payload capacity of every aircraft, but most of the time I took the information from the map view.
P5	The payload view gave a clear overview of the packages that each drone still had/or did not have available. I considered this view frequently for my routing planning.
P6	Didn't use it, liked the changing colors of the aircraft more. Easier to keep looking at the map view/aircraft, since the planning and rearranging was done on the map view as well.
P7	Very useful, allowed me to update routes with more planning by forming delivery clusters with corresponding size of the payload.
P8	The payload view was really handy to see how much payload there was still left. The coloring scheme of drones in case of different amount of payload in the map view was handy, but only to differentiate between: payload on board vs. no payload on board. For the amount of payload I always referred to this view.
P9	Because of the colors in the map display I hardly used the payload view for the final scenarios, once it sunk in that the color of the UAV is the same as the payload level the payload view became obsolete.
P10	Useful for quickly determining the amount of waypoints to be added or removed. For example, I knew I wanted a UAV to pass through 4 points, but there were only 2 loads left, this meant that I had to remove 2 more points.
P11	The payload view was key for my strategic planning. I tried to focus on freeing up as much payload as possible for the flights close to the vacant areas caused by the canceled flights. This was a vert effective strategy in combination with the map view.
P12	It had limited use for me. I used it at the begin of the run to see how much way-points each UAV should visit. During the run I did not use the view. After I rerouted the UAVs I would again use the view to see if I did not miss a way-point by selecting each UAV and checking if the payload view bar was empty.
P13	Useless, especially its positioning. The only information I got from it was at the start to check the capabilities of the UAVs.
P14	This was fine, it had one job and did it well. The color coding is a nice touch, but again: for me quite hard to distinguish from the other shades of yellow/orange/oker/whatever you call it.
P15	Useful. One can see in a split second how much room for payload is left. However, due to the position of the bar (out of view of the field of action), it takes the attention away from where one wants to have it. Namely at the map view. If it was positioned as a tiny bar or as individual blocks above the airplane, one would instantly know how much room for payload is left in a uav, without having to divert the attention from the map. It is different than for the timelime, because the timelime is more of a check afterwards.
P16	Good to have as a backup, but mainly reminded from my mind or colors how many payloads the uavs still could pick up.

Table H-5: Q4: How do you assess the usefulness and clarity of the color use in the display? Please provide examples in your elaboration.

Participant	Comment
P1	Very useful. They can tell in an instance is something is wrong and what is wrong.
P2	The different shades of yellow on the UAVs, indicating spare payload, where difficult to separate. I always had to check extra on the payload view; with a different color scheme, this would not be necessary. It was, however, immediately clear when a UAV had no spare or too little payload, which was nice.
P3	When nothing is selected the colors are quite clear, when selecting something some of the red notifications disappear which is a little bit inconvenient.
P4	Very useful. As mentioned earlier, I probably did 90% of the task based on the colors of the aircraft.
P5	Very clear. Alerts were displayed in red and gave an immediate warning if something in the planning was infeasable. The gray doted pathlines helped to come up with a strategy. The different flight paths could easilz be distinguished as well as the flight path of the selected item.
P6	Really useful and clear! Easy to figure out which aircraft still have payload left, or which aircraft cause conflicts regarding depot arrival times
P7	Color use felt natural, would've picked the same colors. Red for wrong, green for solution space.
P8	Use of color was really good. Color for different amount of payload on board was a bit hard to differentiate between (see question above).
P9	All colors have logical meanings, and after working with the interface for a while you can get all you need from just the map view with the respective colors. There have been scenarios that the map view was the only thing I used since it has all the information you need
P10	Generally useful, the red is a good indicator for any problems. One issue is that the plane, when red, does not show whether it is going to deliver all of its cargo. In one case this lead to confusion, because I didn't realize that it was not yet dropping all of its load.
P11	The color use was very helpful. It was instantly clear due to the bright red when I made a mistake w.r.t. flight time or payload. I also liked the dark and bright yellow colors of the planes in order to indicate,respectively, whether they still had some payload left or had all their payload spent.
P12	The color coding of the display was useful for me. Especially the green area indicating where drones could still go, and red areas indicating a depot conflict. The change of color of the trajectory helped in identifying which UAVs were active. The only color coding that did not provide much added value was the orange and yellow color of the UAV symbol, to indicate if the payload was used up or not. I still used the payload view to check instead of the color of the symbol.
P13	Amazing! Colors all feel very logically chosen.
P14	I elaborated on this above, colors for selection/active/conflicted/to be executed/etc. were good. I had some issues with payload levels.

P15	Colours are great. As mentioned, the contrast between the yellow path lines and the green background make it very clear where the uavs are going and how one should assess the problem.
P16	Colors are good. Except when the UAV becomes red because the depot is full while I am still planning the routes. The color of how many payloads the UAV has is much more useful then.

Table H-6: Q5: How would you solve the scenario depicted below (high-level description)? NOTE: each vehicle only has one spare payload item on board.

Participant	Comment
P1	Make space available for 1 and 3, by assigning way-points to 2.
P2	Have UAV 1 visit the top two extra locations and drop the one on his top left. UAV 2 will pick that one up. UAV 3 can visit the bottom extra location.
P3	2 picks left most customer from 3, 1 picks closest customer, 3 takes two remaining customers.
P4	Aircraft 1 takes the top one of the three nodes. Aircraft 3 is the closest one, so it takes the other two nodes, plus the other two from its initial flight plan (the ones closer to the right. Aircraft 2 takes the node left free by 3, the one at the left.
P5	a) I would add the item closest to 1 to the route of 1 b) I would remove the left most item of 3 and add it to 2 as a last item to be picked up c) Add the remaining 2 items that are unassigned to 3d) check the timeline view and review if the deposit is not overloaded.. optimize
P6	aircraft 1 takes the top green diamond. aircraft 3 loses it most left payload and takes the other two spare ones. And aircraft 2 picks up the point aircraft 3 dropped
P7	Vehicle one gets an additional delivery point enroute, i.e. the point next to its current intended track. Vehicle 3 deletes the leftmost destination and takes the remaining current vacant destinations on the right. Vehicle two takes the point that vehicle 3 was intended to take.
P8	vehicle 1: get rid of left node on the route. Now it can deliver payload at the upper 2 green nodes. vehicle 2: deliver payload at the node which is left out on vehicle 1s route. vehicle 3: deliver payload at lowest green node.
P9	UAV 1 takes the destination closest to it, UAV 2 takes the one from UAV 3 closest to it and UAV 3 goes to the two remaining destinations.
P10	I would use 1 to deliver to the upper point, by moving the closest line. I would then remove the leftmost point of 3, and use 3 to deliver to the 2 remaining points. Next, I would use 2 to deliver to the point vacated by 3.
P11	I would first delete the upper left drop-point of flight 1, then connect flight one to the top-and middle right vacant drop-points. Connect flight 3 to the bottom-right drop-point. Connect flight 2 to the now-vacant top left drop-point.

P12	I would start with one of the planned trajectories, e.g. 1 in this case, and add a open way-point which is not far off its current route. Route 1 is then complete. Next I 'turn' away from the open way-points and make the next trajectory loop complete by implementing way-points of neighboring trajectories. Filly I can use the last UAV to pick up the last open way-points taking the shortest route into account.
P13	UAV 1 receives the extra waypoint right next to it. UAV 2 receives the closes way-point from the route of UAV 3 and UAV 3 receives the remaining two waypoints.
P14	Top free point will be taken up by UAV 1. UAV 3 will cut out its left-most waypoint, and take the other two free points. UAV 2 will take the point that has been dropped by UAV 3.
P15	The three points are positioned right next to paths 3 and 1. These should pick up on the locations. UAV 1 should take the top green block. UAV 3 should take the other two and UAV 2 then takes over the very left spot of UAV 3.
P16	1 would pick up the upper green square. 2 would pick up the left payload from 3. 3 would pick up the other 2 green squares

Table H-7: Q6: How would you solve the scenario depicted below (high-level description)? NOTE: each vehicle only has one spare payload item on board.

Participant	Comment
P1	Free up space for 1 2 and 12. By assigning the way points to the neighboring UAVs.
P2	First, I add points that are (almost) flown over by nearby UAVs to their respective paths. Then I start snipping outliers that are near the edge / near other UAVs, and so on, spreading the workload from top left to bottom right.
P3	First let 4 fly to edge of space, serve as much customers as possible, then work around the clock. Making sure there aren't any payloads and customers left. UAV's should fly as much as possible along the spokes for efficiency.
P4	I take he closest UAVs (say 1 and 2), free up the nodes that can be overtaken by adjacent flight plans (those from 12 and 13), and replace them by the closest green nodes. Then 12 takes the two green nodes closest to its trajectory. I replace adjacent nodes between neighbor flight plans until all nodes are taken. Then I look for options for optimization if I have time left.
P5	First I would look at the loops that are closest to the unassigned elements and re structure them (remove a couple of elments and reshape the loop). And take over the new unassigned elements by the neighbouring loops.
P6	all the payload that needs to be picked up is on the top and left side. So start rearranging aircraft 6 - 11 such that the aircraft closer to the free diamonds can pick up more points. Then aircraft 2 & 3 pick up points that are closest. aircraft 1 gets a rearranged path, drop a couple of his current points and picks up 2 new green diamonds. Idem for aircraft 12, it gives a couple of its blue points to aircraft lower and takes the remaining green points. Idem strategy for the left. In this case, it takes too much time to completely think out the full strategy before hand, so start by the rearranging and see how that goes

P7	High-level description: There is a dense region on the lower half of the screen, i.e. all delivery points are taken. Removing the complete track of one the vehicles on the bottom (e.g. 8) allows for the new vacant points to be taken by other vehicles nearby. Vehicle 8 flies to the top part instead.
P8	Drones 1, 2 and 12 have to deliver at the green nodes on top. Make sure they can cover those nodes by assigning there initial nodes o their neighboring drones. In that way you create a shift of nodes assigned to drones. Every drone takes at least 1 node from their neighbor.
P9	routes far away from the failed routes take over one or two destinations from neighboring routes, so that the routes next to the failed ones only have one or two of their original destinations left. These can then take over the destinations of the failed routes. Multiple routes might take over a failed route if the destinations are located such that this becomes a feasible solution
P10	Use 1, 2 and 12 to reach the green points, removing points where necessary. Then use the neighboring UAVs to reach the vacated points. This will require vacating new points, which will be reached using the next neighbors. When doing this, the brown points will need to be connected as well, likely using a combination of 4, 5 & 6.
P11	Start freeing up as much payload as possible on flights 1, 12,2,3,4,5. Use the remaining flights to fill in the new gaps closest to each respective flight. Then connect the first=mentioned flights to the initial red and green gaps. All the while keeping a close eye on the amount of payloads which needs to be freed.
P12	Not all UAVs become active at once. I use the first UAV to complete a route which uses all payload and focus on the waypoints which are far away. I then turn clockwise or anti-clockwise o the next trajectory and adjust it to use all payload. I continue untill I made a full round. While doing this I try to avoid overlapping trajectories, as this decreased my overview.
P13	UAVs 2, 1 and 12 can only take one extra waypoint, so some other waypoints of their respective routes have to be removed, and everything shifted by one waypoint to adjacent UAVs.
P14	I would try to free up one of the UAVs that have a lot of overlap with others. Then these others can take over its waypoints using their one spare item. Furthermore, the empty (green) waypoints will be taken up as much as possible with the spares from the UAVs that are already in proximity. Then, the UAV that has been freed up will take the rest. Obviously, this will probably take some itereations as the exact payload numbers will likely not match up perfectly.
P15	Again the same strategy is enforced. The UAV paths right next to the open blocks take them over. The paths that come free from that action are taken by the UAV paths next to those, until no blocks are free anymore. In this case, UAV 1 takes over the 4 gren blocks on top and loses the three blocks on the right. UAV 12 takes over these three and loses 2. UAV 11 takes the 2 that come free from that action and loses one of its own blocks and finally UAV 10 takes over this block. For the brown blocks, the same approach is enforced. UAV 3 takes over 2 and loses 1. UAV 2 takes over the one block that comes from UAV3. UAV 4 takes over 4 blocks and loses 3. UAV 5 takes over 3 and loses 2 etc.

P16	free up space in the uavs that are most far away from the still to be picked up payloads. Thus the closer the uav to the free payloads is the more it can still pick up.

Table H-8: Q7: Do you have any other comments or suggestions with respect to the interface or the experiment?

Participant	Comment
P1	An overall performance score might be nice, to see if you can do better. Or maybe an optimize suggestion by automation?
P2	Nothing more to add.
P3	All the colors and button clicks are mostly not immediately intuitive and need lots of explanation. Can probably be made more intuitive by having drag and drop, icons instead of too many colors.
P4	It is difficult to swap one node from one aircraft to the other. If I want to assign to the currently selected UAV to a node that is taken in another flight plan, I have to select the other one, free out the node, select back the first one, take the node, etc. It would be way more efficient if I could free out nodes assigned to other airplanes while the current UAV is selected, without having to dis-select it (say, pressing CTRL+Left Click).
P5	If there would be some suggestions for optmizing current loops/routs, that would be easier for the air traffic controllers
P6	Only for the RSME, show latest value
P7	I disliked the pre-programmed routes, which could interfere with route modifications of existing vehicles if new vehicles appeared.
P8	Maybe also show performance (shortest route flown). It was hard to see if some actions would reduce the distance flown or increase it. Show (initial) direction of drones that are about to start flying (in grey). If you hover over the drone, maybe amount of payload on board shows up to the drone on the map view.
P9	It is a useful tool for DVRP and also fun to use in my opinion
P10	No.
P11	No I really liked the interface. It is quite basic, though I consider that a good thing due to the duration of the test. minimal distractions, which benefits the overall time-usage of the test.
P12	Add a check to see if all way-points are incorporated in the UAV trajectories. I sometimes missed one at first, and had to go over each UAV and check the payload view to be sure.
P13	Make clicking easier. Put the payload view at the top of the display, in your line of sight. Touchscreen? :P
P14	I needed a bit more training in the beginning to really understand what was happening, and how the controls worked. Only after that I could start developing a strategy.
P15	No
P16	no, looks good.

Appendix I

Code Architecture

I-1 Vehicle Routing Problem Optimization

The Vehicle Routing Problem optimization is used for generating the scenarios for the experiment as well as to solve these scenarios for post experiment comparison. The architecture of the tool is depicted in Figure I-1. The optimization tool primarily consists of Python code to setup the necessary variables for fast scenario generation and solving. All relevant settings can be managed from the `experimentDesign` function. This function creates a `VRProblemSet` consisting of all experimental conditions, which in turn consists of multiple `VRProblem` instances for each specific condition. Each `VRProblem` is then solved using `solveVRP`, which is a slightly modified Python script that is part of the Google Optimization Tools. The Google Optimization Tools is a library specifically designed for solving combinatorial optimization problems, which contains Traveling Salesman and Vehicle Routing Problems. The library is written in C++, but comes with a Python wrapper, which is what `solveVRP` interfaces with. After solving, `experimentDesign` can call `VRProblemSetPlotHelper` to generate some simple plots visualizing the original problem, the problem with the failure, and the optimized solution to the problem.

I-2 Multi-UAV Simulator

The multi-UAV simulator is JAVA based and the code base originated from the MUFASA editor (MEDIT). Use Eclipse Luna version 4.4.2 and JAVA version 1.6 for code development. The simulator uses the scenario files from the `scenarios` folder that are generated by the VRP optimization code. The `setup.txt` file in the `experiment-design` folder determines, based on the participant ID, the desired order of the scenarios. On launch, the simulator asks for the participant ID and determines, from the setup file and the amount of log files in the `logs` directory what the next scenario should be. There is also a `logs-click` folder, which contains the click logs, separate form the flightplan logs in the `logs` folder.

The architecture of the multi-UAV simulator is depicted in Figure I-2. The main objects are contained in the `+atclib.ssd.main` package. The `SSDFrame` is the main executable. A `Mission` object is defined that deals with the scenario on fleet level. The fleet of vehicles is then made up of UAV objects, which each contain a `Flightplan`, which consists of `Waypoint` objects. The complementary objects starting with GL define the visual representation (view) of the non-GL data objects (model). Both the data and click log objects are contained in the `+atclib.ssd.main` package. The map view display is contained in the `+atclib.ssd.display.elements` package, the timeline view in the `+atclib.ssd.display.timewindow` package (although the `GLTimeWindowContent` object still resides in `+atclib.ssd.display.elements`), and the payload view in the `+atclib.ssd.display.payloadwindow` package. The envelopes around the UAV guidance reference (guidref) are the `PerformanceEnvelope` and `PathStretchEnvelope` for visualizing flight time and delay to accommodate depot capacity, respectively. Both envelopes consist of `EnvelopeElement` objects for each flightplan leg.

For the experiment, and export of the code should be made with Eclipse, using File ¿ Export ¿ Runnable JAR file. This file should be located in a folder also containing the extra folders that are not included in the export: `command`, `config`, `experiment-design`, `icons`, `lib`, `logs`, `logs-click`, `scenarios`, `scenes`, `sectors`, `shaders` and `traffic`.

I-3 Experiment Survey

The web form that seres as the experiment survey is based on PHP and HTML, its architecture is depicted in Figure I-3. Use XAMPP on the host machine to locally serve the web form. The web form will automatically generate the resulting data files for whatever participant ID is filled out by the user, hence no further setup is required. A simple check for whether the participant ID already exists is implemented to prevent accidental data overwrites. Variables that allow the survey to know its state are visible in the URL, which also allows for manual intervention in case a problem arises when using the survey during an experiment.

I-4 Post Experiment Data Processing

The post experiment data processing scripts are MATLAB based and serve to, first, convert the log files to `.mat` format, second, analyze the data and create all figures. The architecture of the code is depicted in Figure I-4. With the experiment log files in the appropriate folder, run the three parse scripts to convert the data. Then, run the process script to calculate all data associated with the dependent variables and generate the figures. The figures will be automatically saved in the associated folder. Note that not only the experiment logs are required, but also the result logs containing the optimized solutions for the scenarios. Separately, plots for experiment scenarios, optimized solutions and participant solutions can be generated by running `plotScenarios`.

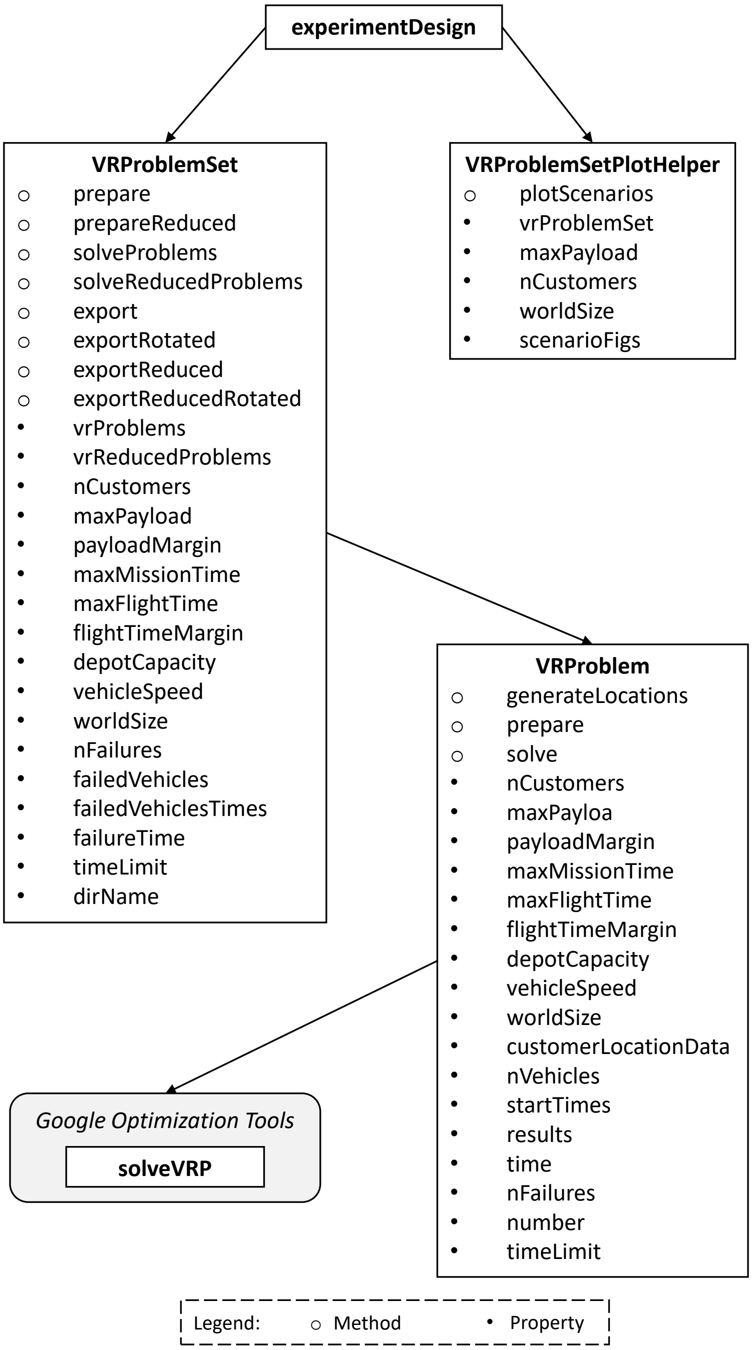


Figure I-1: Vehicle Routing Problem optimization code architecture.

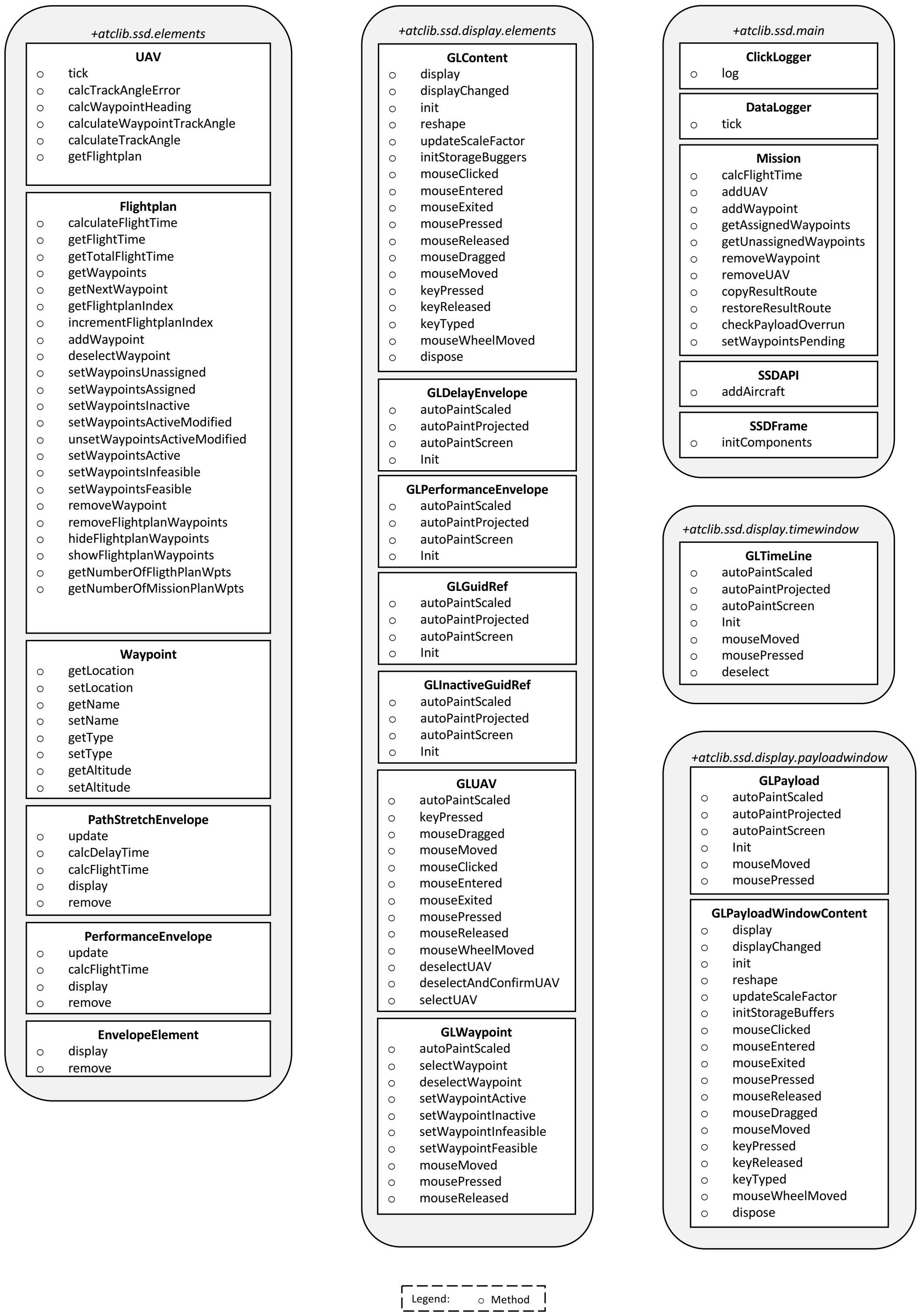


Figure I-2: Multi-UAV simulator code architecture.

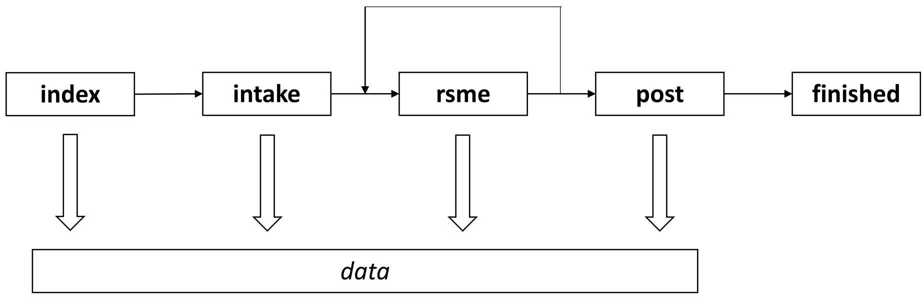


Figure I-3: Survey web form code architecture.

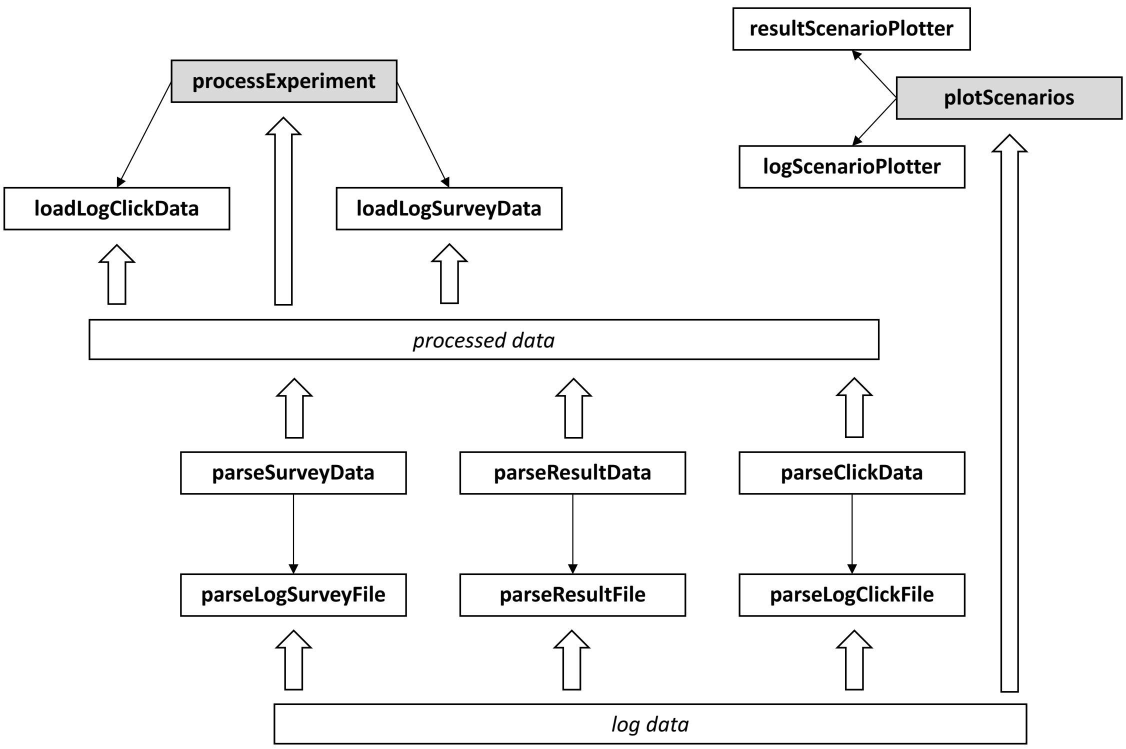


Figure I-4: Post processing code architecture.

Appendix J

Concluding Remarks and Recommendations

This chapter presents a more elaborate overview of recommendations in addition to the paper. Section J-1 discusses the interface design, Section J-2 the experiment design, and Section J-3 presents suggestions for future work.

J-1 Interface Design

The interface was generally effective, however, some possible improvements were identified during the experiment. The location of the payload view is too far from the main activity on the map view requiring participants to look up and down a lot. The payload view can potentially be moved to the top of the map view so it is in closer proximity or can be integrated in the map view by depicting the payload level in a small info window next to each vehicle.

The arrival blocks in the timeline view are not separated very well, which makes it difficult to distinguish between different vehicles when the arrival times are in close proximity. Hence, introducing differently colored borders around the arrival time blocks is recommended.

For the more complex scenarios, the control input method was considered to require too much effort. Customer locations first need to be unassigned before they can be assigned to another vehicle. It is recommended to investigate more efficient ways to maintain effective control in complex and visually cluttered scenarios, while not requiring an unnecessarily large amount of control inputs.

Some participants noted that it is difficult to compare optimality of possible solutions purely visually. Addition of some kind of route cost estimator, so that the visual observations can be more effectively compared is potentially useful.

Finally, it is recommended to extend the delay allocation visualizations (red ellipses) for solving depot congestion infeasibilities to also provide information pro-actively not merely reactively. For example, by also visualizing depot congestion infeasibilities associated with control actions in the solution space, thereby possibly avoiding self-induced infeasibilities.

J-2 Experiment Design

Since some training effects could still be observed in the experiment result data, it is recommended to increase the training volume for future experiments. Furthermore, since successful participant strategies were very similar, explicit information on how to effectively solve VRPs might be provided during training. Since the interface is not trivial to use, it is recommended to maintain sufficient interface training scenarios to ensure participants are comfortable with the controls.

Due to the heavy use of different colors in the interface, each carefully considered and associated with specific information, participants should be screened for color blindness.

Scenario difficulty was considered somewhat insufficient, since the limit of human control for VRPs was not reached during the experiment. More difficult scenarios should therefore be considered for future experiments.

J-3 Future Work

Several suggestions for future work have been identified. First, regarding the experiment scenarios, less predictable scenarios can be considered, which could consist of vehicle failures, pop-up customers or other dynamics that are not the same for every scenario (both in occurrence as well as timing). Constructing scenarios that contain unanticipated events allows for better EID evaluation, since supporting these tasks is a core focus of the framework. Also, overconstrained scenarios can be considered in which no solution is possible (and are hence scenarios in which an optimization algorithm would be unable to find a solution). In these scenarios, customers might be assigned different priority levels to assess operator strategies in which customers to serve and effects on overall mission success.

Second, asymmetric VRPs can be considered by, for example, introducing wind. This means the cost associated with moving from a to b is not equal anymore to moving from b to a. This also means the visual distance and proximity of customers to vehicles and guidrefs is not directly related anymore to the cost associated with visiting the customers. Since humans rely on visual perception to optimize routes, effects of asymmetry in the VRP are interesting to investigate. Maybe the customer locations, or vehicle routes can somehow be smartly distorted to result in correct visual cost representation...

Third, customers can be assigned time windows in which deliveries should take place. This is equivalent to waypoints with Required Time of Arrival (RTA) in 4D navigation in the Air Traffic Management (ATM) domain.

Fourth, mission integration into the ATM system could be of interest. For that purpose interfaces with Remotely Piloted Aircraft Systems (RPAS) Concept of Operations (CONOPS) could be identified and implemented in the interface and an experiment.

Fifth, only a single automation algorithm was used to obtain the current results. More methods can be investigated and compared to more elaborately assess benefits of having a human-in-the-loop. Also, to support this assessment more metrics can be designed, such as solution robustness.

Finally, a combination of human control and automation could be investigated to see if benefits from each can be used in a complementary way.

Bibliography

Ainsworth, L., & Marshall, E. (1998). Issues of quality and practicability in task analysis: preliminary results from two surveys. In *Ergonomics* (Vol. 41, pp. 1607–1617). doi: 10.1080/001401398186090

Amelink, M. H. J., Mulder, M., van Paassen, M. M. R., & Flach, J. (2005). Theoretical foundations for a total energy-based perspective flight-path display. *The International Journal of Aviation Psychology*, *15*(3), 205–231. doi: 10.1207/s15327108ijap1503

Andre, A. D., & Wickens, C. D. (1995). When users want what's not best for them. *Ergonomics in Design: The Quarterly of Human Factors Applications*, *3*(4), 10–14.

Annett, J. (2003). Hierarchical task analysis. In *Handbook of cognitive task design* (pp. 17–35).

BI Intelligence. (2016). *THE DRONES REPORT: Market forecasts, regulatory barriers, top vendors, and leading commercial applications.* Retrieved 2017-01-03, from `http://www.businessinsider.com/uav-or-commercial-drone-market-forecast-2015-2?international=true{&}r=US{&}IR=T`

Chen, J. Y. C., Barnes, M. J., & Harper-Sciarini, M. (2011). Supervisory control of multiple robots: Human-performance issues and user-interface design. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, *41*(4), 435–454. doi: 10.1109/TSMCC.2010.2056682

Cummings, M. L. (2004). Automation Bias in Intelligent Time Critical Decision Support Systems. *AIAA 1st Intelligent Systems Technical Conference*, *2*(2004), 557–562. doi: 10.2514/6.2004-6313

Cummings, M. L. (2006). Automation and Accountability in Decision Support System Interface Design.

Cummings, M. L., Bertucelli, L. F., Macbeth, J., & Surana, A. (2014). Task versus vehicle-based control paradigms in multiple unmanned vehicle supervision by a single operator. *IEEE Transactions on Human-Machine Systems*, *44*(3), 353–361. doi: 10.1109/THMS.2014.2304962

Cummings, M. L., Bruni, S., Mercier, S., & Mitchell, P. J. (2007). Automation Architecture for Single Operator , Multiple UAV Command. *The International C2 Journal*, *Vol 1*(No 2).

Cummings, M. L., Brzezinski, A. S., & Lee, J. D. (2007). Operator performance and intelligent aiding in unmanned aerial vehicle scheduling. *IEEE Intelligent Systems*, *22*(2), 52–59. doi: 10.1109/MIS.2007.39

Cummings, M. L., Clare, A., & Hart, C. (2010). The Role of Human-Automation Consensus in Multiple Unmanned Vehicle Scheduling. *The Journal of the Human Factors and Ergonomics Society*, *52*(1), 17–27. doi: 10.1177/0018720810368674

Dinadis, N., & Vicente, K. J. (1999). Designing functional visualizations for aircraft systems status displays. *The international journal of aviation psychology*, *9*(3), 241–269. doi: 10.1207/s15327108ijap0903

Dry, M., Lee, M. D., Vickers, D., & Hughes, P. (2006). Human Performance on Visually Presented Traveling Salesperson Problems with Varying Numbers of Nodes. *The Journal of Problem Solving*, *1*(1), 20–32.

Ellerbroek, J., Visser, M., Van Dam, S. B. J., Mulder, M., & van Paassen, M. M. (2011). Design of an airborne three-dimensional separation assistance display. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *41*(5), 863–875. doi: 10.1109/TSMCA.2010.2093890

Endsley, M. R. (1988). Situation Awareness Global Assessment Technique (SAGAT). In *Aerospace and electronics conference, 1988. naecon 1988., proceedings of the ieee 1988 national* (pp. 789–795).

Endsley, M. R. (1990). Objective evaluation of situation awareness for dynamic decision makers in teleoperations. In *Presented at the engineering foundation conference on human-machine interfaces for teleoperators and virtual environments, santa barbara, ca.*

Endsley, M. R. (1995). Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *37*(1), 32–64. doi: 10.1518/001872095779049543

Fuchs, C., Borst, C., de Croon, G. C. H. E., van Paassen, M. M., & Mulder, M. (2014). An Ecological Approach to the Supervisory Control of UAV Swarms. *International Journal of Micro Air Vehicles*, *6*(4), 211–229. doi: 10.1260/1756-8293.6.4.211

Guerlain, S. A., Smith, P. J., Obradovich, J. H., Rudmann, S., Strohm, P., Smith, J. W., & Svirbely, J. (1995). Dealing with brittleness in the design of expert systems for immunohematology. *Immunohematology/American Red Cross*(12.3), 101–107.

Hall, D. S., Shattuck, L. G., & Bennett, K. B. (2012). Evaluation of an Ecological Interface Design for Military Command and Control. *Journal of Cognitive Engineering and Decision Making*, *6*(2), 165–193. doi: 10.1177/1555343412440696.

Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in psychology*(52), 139–183.

Jahns, D. (1973). A Concept of Operator Workload in Manual Vehicle Operations. *Forschungsinstitut Anthropotechnik, Meckenheim*(Bericht Nr. 14).

Jamieson, G. A., Miller, C. A., Ho, W. H., & Vicente, K. J. (2007). Integrating task- and work domain-based work analyses in ecological interface design: A process control case study. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, *37*(6), 887–905. doi: 10.1109/TSMCA.2007.904736

Johannsen, G. (1976). Preview of Man-Vehicle Control Session. In *Monitoring behavior and supervisory control* (pp. 3–12). Springer US.

Johannsen, G. (1977). Workload and Workload Measurement. In *Proceedings of the nato symposium on theory and measurement of mental workload* (pp. 3–11). Springer Science+Business Media New York.

Kilgore, R., & St-Cyr, O. (2006). *The SRK inventory: a tool for structuring and capturing a worker competencies analysis* (Vol. 50) (No. 3). Sage CA: Los Angeles, CA: SAGE Publications.

Kirwan, B., & Ainsworth, L. K. (1992). *A guide to task analysis: the task analysis working group.* CRC press.

Koldaev, A. V. (2007). Non-Military UAV Applications. *Aero India International Seminar*(2007 Edition).

Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European journal of operational research*, *59*(3), 345–358.

Lazarski, A. J. (2002). Legal Implications of the Uninhabited Combat Aerial Vehicle. *Air & Space Power Journal*(Summer 2002), 74–83.

Macgregor, J. N., & Chu, Y. (2011). Human Performance on the Traveling Salesman and Related Problems. *The Journal of Problem Solving*, *3*(2), 1–29.

Mcewen, T. R., Flach, J. M., & Elder, N. C. (2014). Interfaces to Medical Information Systems : Supporting Evidenced Based Practice. In *2014 ieee international conference on systems, man, and cybernetics (smc)* (pp. 335–340). IEEE.

Miller, C. A., & Vicente, K. J. (1998). Toward an integration of task-and work domain analysis techniques for human-computer interface design. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *42*(3), 336–340.

Mouloua, M., Gilson, R., & Hancock, P. (2003). Human-Centered Design Of Unmanned Aerial Vehicles. *Ergonomics in Design*(Winter 2003), 6–11.

Mouloua, M., Gilson, R., Kring, J., & Hancock, P. (2001). Workload, situation awareness, and teaming issues for UAV/UCAV operations. In *Proceedings of the human factors and ergonomics society* (pp. 162–165).

Prinet, J. C., Terhune, A., & Sarter, N. B. (2012). Supporting Dynamic Re-Planning In Multiple UAV Control: A Comparison of 3 Levels of Automation. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *56*, 423–427. doi: 10.1177/1071181312561095

Rasmussen, J. (1976). Outlines of a hybrid model of the process plant operator. In *Monitoring behavior and supervisory control* (pp. 371–383). Springer US.

Rasmussen, J. (1980). The human as a systems component. In *Human interaction with computers.* Academic Press, Incorporated.

Rasmussen, J. (1983). Skills Rules and Knowledge, Other Distinctions in Human Performance Models. *IEEE Transactions on Systems, Man, and Cybernetics*, *13*(3), 257–266.

Rasmussen, J. (1985). The role of hierarchical knowledge representation in decisionmaking and system management. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-15*(2), 234–243. doi: 10.1109/TSMC.1985.6313353

Reising, D. V. C., & Sanderson, P. M. (2002). Work domain analysis and sensors I: principles and simple example. *International Journal of Human-Computer Studies*, *56*(6), 569–596. doi: 10.1006/ijhc.1006

Rodgers, M. D., Mogford, R. H., & Strauch, B. (2000). Post hoc assessment of situation awareness in air traffic control incidents and major aircraft accidents. In *Situation awareness analysis and measurement* (pp. 73–112).

Ruff, H., Narayanan, S., & Draper, M. H. (2002). Human Interaction with Levels of Automation and Decision-Aid Fidelity in the Supervisory Control of Multiple Simulated Unmanned Air Vehicles. *Presence: Teleoperators and Virtual Environments*, *11*(4), 335–351. doi: 10.1162/105474602760204264

Sarter, N. B., & Woods, D. D. (1992). Pilot interaction with cockpit automation: Operational experiences with the flight management system. *The International Journal of Aviation Psychology*, *2*(4), 303–321. doi: 10.1207/s15327108ijap0204

Sarter, N. B., & Woods, D. D. (1994a). Decomposing automation: Autonomy, authority, observability and perceived animacy. In *First automation technology and human performance conference* (pp. 22–26).

Sarter, N. B., & Woods, D. D. (1994b). Pilot interaction with cockpit automation II: An experimental study of pilots' model and awareness of the flight management system. *The International Journal of Aviation Psychology*, *4*(1), 1–28. doi: 10.1207/s15327108ijap0401

Shepherd, A. (2001). *Hierarchical task analysis.* London: Taylor & Francis.

Sheridan, T. B. (1992). *Telerobotics, Automation, and Human Supervisory Control.* MIT press.

Sheridan, T. B., & Verplank, W. L. (1978). *Human and computer control of undersea teleoperators* (Tech. Rep.). Massachusetts Institute of Technology, Cambridge, Man-Machine Systems Lab.

Smeur, E. J. J., Chu, Q., & de Croon, G. C. H. E. (2016). Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Aerial Vehicles. *AIAA Guidance, Navigation, and Control Conference*.

Smith, P. J., McCoy, E. C., & Layton, C. (1997). Brittleness in the Design of Cooperative Problem-Solving Systems: The Effects on User Performance. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *27*(3), 360–371.

Tijmons, S., de Croon, G., Remes, B., de Wagter, C., & Mulder, M. (2016). Obstacle Avoidance Strategy using Onboard Stereo Vision on a Flapping Wing MAV. *arXiv preprint*(arXiv:1604.00833).

Toth, P., & Daniele, V. (2014). *Vehicle routing: problems, methods, and applications.* Society for Industrial and Applied Mathematics.

van Lochem, S., Borst, C., de Croon, G. C. H. E., van Paassen, R., & Mulder, M. (2015). Ecological Interface Design for Collaboration of Multiple UAVs in Remote Areas.

Vicente, K. J. (1999). *Cognitive work analysis: Toward safe, productive, and healthy computer-based work.* CRC Press.

Vicente, K. J., & Rasmussen, J. (1992). Ecological Interface Design : Theoretical Foundations. *IEEE Transactions on Systems, Man, and Cybernetics*, *22*(4), 589–606.

Wickens, C. D., Mavor, A. S., & James, P. (1997). *Flight to the Future: Human Factors in air Traffic Control.* National Academies Press.

Wickens, C. D., Mavor, A. S., Parasuraman, R., & Mcgee, J. P. (1998). *The future of air traffic control: Human operators and automation.* National Academies Press.

Wiener, E. L. (1988). Cockpit automation.