Ontological Decision-Making for Disaster Response Operation Planning

Aaron Wheeler, Jim Dike, and Michael Winburn
3 Sigma Research, Indialantic, Florida, USA

Abstract – Disaster response operation planners could benefit from a software tool to assist them in extracting individual mission information from various operation tasking orders, determining resources able to carry out the operation, construct a number of alternative operation plans, alert them to mission-critical changes to these plans, and allow them to quickly replan when problems occur. Within the context of aircraft operation planning for disaster relief, we use an ontological approach to solve these problems that includes modeling the air coordination plan (ACP) knowledge domain with OWL and SWRL, making inferences about the capabilities of real-world resources using an automated reasoner, retrieving relevant operation planning resources with SPARQL, then building multiple plans from the results of the queries. We also use SPARQL queries of the operation planning ontology database to provide alerts to operation planners of potential mission-critical changes to resources. Our results show that this approach allows operation planners to generate and monitor aircraft operation plans having dependencies on aircraft and pilot availability, weather effects on payload effectiveness, flight route distance, and fuel availability. We conclude that an ontological approach to aircraft operation planning does work and also supports broader goals of data transparency, information sharing, and domain knowledge persistence.

Keywords: ontology, decision-making, operation planning, disaster response

1 Introduction

Large-scale disasters come from many sources, ranging from natural events like hurricanes, tornadoes, earthquakes, fires, famines, volcanic eruptions, or tsunamis; to human-made crises like nuclear accidents or political unrest. Aircraft provide a critical component to first response and longer term disaster recovery.

This research focuses on operation planning of air support for disaster recovery from natural and human-made disasters and political unrest. Specifically, we introduce a methodology for automated reasoning about operation plans to support disaster response decision-making.

Because of their ability to avoid ground obstacles, aircraft provided critical components to disaster recovery, including communication, aerial reconnaissance, supplies, and evacuation. Consider the following disaster recovery scenario.

Suppose that a category five hurricane strikes a major population center on the coastal United States. Winds and flooding destroy communication and transportation networks, while people lack food, water, shelter, and medical care. Disaster response operation planners use aircraft to provide a number of basic needs. They deploy a drone aircraft to circle the affected area with a payload to provide an ad hoc private cellular network to first responders. Helicopters bring smart phones to emergency responders already on the ground so they can collect and share situational awareness data (GPS linked text, audio, still pictures, and video) with each other and with the command center located outside the affected area. Other drone aircraft provide aerial reconnaissance to assess damage and look for victims. Operation planners send helicopters to evacuate victims. Both helicopters and fixed-wing aircraft drop supplies to those stranded but not in immediate danger.

Unplanned changes to aircraft status and pilot availability, supply shortages, weather conditions, and scheduling conflicts, to name but a few complications, require operation planners to have a number of contingency plans and to know quickly when the current plan cannot succeed and replanning must occur.

Section 2 of this paper provides background information about our proposed system architecture, how ontologies can support emergency operation planning, and specific ontologies for aircraft operation planning. Section 3 describe the steps for generating primary and contingency operation plans using our ontological approach. Section 4 discusses our main findings, and section 5 concludes with a brief description of a current project that we intend to apply these transformations, and also a discussion of future research directions.

2 Background

This section introduces the Semantic Emergency operation planning with Ontological Reasoning (SEMPOR) system architecture and the Semantic Web technologies used by SEMPOR to reason about aircraft operation plans.

Aircraft operation planning for disaster response and for military operations share a number of similarities. Therefore,
we leverage the same terminology when appropriate. This has the advantage of supporting cross-domain information exchange and coordination, especially important during emergency situations.

2.1 System Architecture

Figure 1 below shows the SEMPOR system architecture. The left set of components represent the various data sources from which SEMPOR will extract relevant data in order to construct and update operation plans. SEMPOR itself comprises the components to the right.

![SEMPOR System Architecture](image)

Figure 1. SEMPOR System Architecture.

The Air Coordination Plan (ACP) database stores the current air coordination plans. The Operation Planning System (OPS) provides flight route details to the reasoner. Other data sources provide up-to-the-minute weather status, as well as details about mission-critical assets like aircraft, pilots, equipment, and supplies. SEMPOR may interactively query these data sources for specific information instead of using all possible information available about each operation asset.

Disaster response operation planners interact with SEMPOR and the ACP Message Service (ACPMS) through a graphical user interface. Both SEMPOR and ACPMS retrieve information from an ontology database. The ontology database contains specially formatted data collected from a variety of data sources that SEMPOR needs to generate operation plans.

2.2 Ontologies

The Extensible Markup Language (XML) provides language for describing the structure of information to support automated processing [8]. The XML Schema Definition (XSD) language contains type and element definitions that describe characteristics of well-formed elements and attributes of XML documents [9]. However, the XSD language does not express semantics [14] and so creates difficulties for Semantic Web technologies.

An ontology provides a formal description of a domain of discourse [1] and represents a core component of the Semantic Web. The World Wide Web Consortium (W3C) has recommendations for expressing ontologies using RDF [10], RDFS [11], and OWL [2].

The Resource Description Framework (RDF) represents a W3C standard for the Semantic Web that makes statements using subject-predicate-object triples [10]. RDF Schema (RDFS) extends RDF by allowing for the definition of classes and properties that describe other classes and properties [11].

Both RDF and the OWL 2 Web Ontology Language provide a formal semantics for interpreting ontology structures. OWL extends RDF and RDFS with terminology to express ontologies using description logic, a decidable fragment of first order logic [12]. OWL DL ontologies belongs to subset of OWL ontologies that satisfy the expressive requirements of a description logic. The expressiveness, completeness, and decidability of OWL DL makes it possible for automated reasoning engines to discover new information implied by ontology structures [13].

SEMPOR uses several Semantic Web languages to represent, reason about, and query for data needed to construct aircraft operation plans.

SEMPOR uses the Web Ontology Language (OWL) to describe our aircraft operation planning domain and the real-world resources needed for operation plans. OWL-DL, a subset of the full OWL specification, allows us to describe aircraft operation planning concepts and resources using a decidable and complete subset of first-order logic called description logic (DL). OWL-DL semantics allow for automated reasoning about classes and properties of things [2].

OWL allows us to make inferences about additional class memberships and properties of entities. However, we cannot automatically create relations between individuals based on comparisons of their explicit or inferred properties. In order to accomplish this type of reasoning, we use the Semantic Web Rule Language (SWRL) [3]. Rules in SWRL resemble production rules (e.g. “if X and Y then Z”) in that individuals that match a conjunction of conditions will trigger or fire another set of conditions from which we can infer still more class membership and property relations for the individuals.

SEMPOR uses the SPARQL Protocol and RDF Query Language (SPARQL) to retrieve data from the inferred data model. The Resource Description Framework (RDF) provides a formalism for making statements about resources in terms of subject-predicate-object triples. SPARQL resembles
Structured Query language (SQL) used for querying relational databases, but includes Uniform Resource Identifiers (URI) for globally unambiguous queries [4].

An ontological representation of aircraft operation planning supports the W3C Decisions and Decision-Making Community Group objective of developing a semantic representation of decision-making concepts to improve sharing and use of decision information across diverse systems [7].

2.3 Ontologies for Operation Planning

SEMPOR uses several domain ontologies expressed using OWL and SWRL. Some of these ontologies we constructed specifically for SEMPOR, while others represent OWL versions of XML schema already developed to express ACP and related resources. We use a semi-automated approach to make the transformation from XML to OWL.

The Air Operations Community of Interest (AOCOI) has developed a number of XML schemas to describe air operations resources and concepts. They developed these schema with the intent of “enabling data transparency across the Air Operations domain, enable standard data-driven services for managers at all levels and reduce data entry, homegrown systems, and the need for record re-creation” [5]. We have chosen to use their vocabulary, taxonomy, and semantics in SEMPOR to support this effort and provide a foundation for better coordination between civilian and military responders during times of crisis.

The AOCOI has XML schemas (version 1.0.15.2) for Airspace, Common Mission Description (CMD), Friendly Order of Battle (FrOB), Mission Task Request (MTR), and Reference [5]. We have transformed to OWL the CMD and FrOB XML schemas to use in our prototype some of the terms these schemas contain. The CMD ontology provides use with concepts like ConfigurationItem and RampFuel, while the FrOB ontology gives us terms like BurnRate.

The Operation Planning Ontology contains concepts and properties specific to operation planning and also a representation of operation planner knowledge and reasoning. The Operation Asset Ontology contains representations of all operation assets not already represented by other ontologies.

3 Operation Planning

Now we enumerate the steps that operation planners and SEMPOR will follow during the course of generating air operation plans.

1. Collect information relevant to the operation from the ACP and other data sources.

2. Create in the ontology database a prototype operation plan.

3. Apply a reasoner to the ontology to infer candidate operation plan elements.

4. Build operation plan alternatives from these candidate elements.

5. Rank the operation plan alternatives from most to least preferred.

We will describe each of these steps in more detail in the sections to follow.

3.1 Data Acquisition

Collect information relevant to the operation from the ACP. Data relevant to the operation gets retrieved from their respective data sources and stored in the ontology database using OWL classes, relations, and values. This data acquisition strategy allows us to use heterogeneous data sources.

3.2 Operation Plan Prototype

SEMPOR assumes that the operation planner will create a high level description of an operation plan as part of the operation planning process. This high level specification will describe abstract planning elements from which SEMPOR can infer real elements using OWL and SWRL.

Figure 2 below shows an example of the information contained in the operation plan prototype. The operation planner provides an abstract description of each class of thing that could serve in the role of a particular operation plan element. In this example, the operation plan needs only one element. Classes of things that could perform the role required of this element appear in option1. This entity has relations to two configurations, one specifying a hoist for lifting people and equipment and the other a cellular base station payload. Anything that SEMPOR can infer has one or both of these configurations can become an element in this operation plan.

Figure 2. Operation plan prototype data.

Note that SEMPOR interprets multiple needsElement and hasConfiguration differently. SEMPOR interprets multiple needsElement relations using a logical AND, meaning the operation plan must have “this element AND this
element AND ...”. However, SEMPOR interprets multiple hasConfiguration relations using a logical XOR, meaning the aircraft operation plan must have only one of “this configuration OR this configuration OR ...”. SPARQL queries and post-processing that we describe later will exploit these logical constructs to generate final operation plan alternatives.

### 3.3 Ontological Reasoning with OWL and SWRL

Describing operation planning resources and concepts with OWL-DL and SWRL allows us to use automated reasoners to make inferences over the logical structure of the knowledge representation. OWL-DL reasoners make it possible to discover implied relationships between real-world entities and use these to infer the suitability of aircraft and pilot for the air operation. We use the Pellet OWL 2 reasoner because it has a Java API, supports the latest version of OWL 2, and allows for most SWRL builtins [6].

We create in the ontology a class called Conf_Hoist to represent the set of things having a particular hoist configuration. We define the class by stating that individual entities belong to the set if they have both a has_ConfigurationItemObj relation to an individual that belongs to the set of Hoist things and a has_capacity property value of at least 500 pounds. Figure 3 below shows this graphically as a Venn diagram.

Figure 3. Logical AND with ontology concepts.

```
has_ConfigurationItemObj some Hoist
AND
has_capacity some [500 or more]
```

Figure 4 below shows a similar diagram, but for inference by logical OR. In this example, we define a class called AirAmbulance to represent the set of things with equipment capable of providing emergency medical support. Air ambulances can come equipped with a variety of medical payloads so we use a logical OR to capture them all in the definition of the set.

Figure 4. Logical OR with ontology concepts.

```
AirAmbulance
AND
has_ConfigurationItemObj some CardiacMonitor
has_ConfigurationItemObj some Defibrillator
has_ConfigurationItemObj some Oxygen
has_ConfigurationItemObj some Respirator
has_ConfigurationItemObj some Sterilizer

OR

some_aircraft has_ConfigurationItemObjs
```

In both examples above, we could further specify that things that belong to one of the sets, either Conf_Hoist_500 or AirAmbulance, also have additional property values or relations to other individuals or types of things. In this way we can extend the number of inference steps that the reasoner can make, which could result in long and elaborate chains of reasoning that automatically find not so obvious classes, relations, and property values for individual entities needed for aircraft operation planning.

While OWL allows us to infer set memberships and additional property relations for individual entities, SWRL allows us to compare set memberships, relations, and property values of individuals then create relationships or assign property values. Figure 5 below shows a sequence of rules in SWRL that we use to assign additional characteristics to operation planning resources. The figure shows in words what the left-hand side (LHS) of each rule does and then shows on the right-hand side (RHS) in SWRL syntax the binding that results if the conditions on the LHS get triggered.

Figure 5. Sequence of rules for inferring additional properties.

```
Available(?x)

hasPilot(?a, ?p)

Payload effective given weather?
Effective(?x)

Does aircraft have the payload configuration needed to fill the role of a particular operation plan element?
configuredFor(?a, ?o)

What additional fueling demand does the aircraft have?
hasRefuelingDemand(?a, ?demand)

Does the aircraft need refueling?
needsRefueling(?a, true)
```

Figure 6 shows the LHS conditions in the SWRL syntax. For example, Available(?x) means that an individual instantiated on the LHS with variable ?x belongs to the set of Available things. For another example, hasPilot(?a, ?p) means that when the conditions on the LHS of the rule fire, the aircraft entity instantiated as the variable ?a has a hasPilot relation to a pilot entity instantiated on the LHS with the variable ?p.
3.4 Ontology Database Queries with SPARQL

Automated reasoner, like Pellet load the explicit data model described with OWL and SWRL then makes an inferred model that contains statements implied by the logical structure of the explicit data model. SEMPOR uses database queries express with SPARQL to retrieve information from this inferred model.

Specifically, SEMPOR uses a SPARQL query to select for each inferred operation element the following values:

1. Operation element (e.g. option1)
2. Aircraft
3. Pilot
4. Configuration item (e.g. hoist_1)
5. Refueling demand

Figure 7 below shows the SPARQL query used by SEMPOR to retrieve these records from the inferred data model.

Figure 7. SPARQL query to retrieve operation elements.

```sparql
where {
  ?a a sp:OperationElement .
  ?a sp:configuredFor ?o .
  ?o sp:hasConfiguration ?c .
  ?o omcm:has_ConfigurationItemObj ?c1 .
  ?a sp:hasFile ?f .
  ?p a sp:available .
  ?a sp:hasRefuelingDemand ?rd .
}
```

SEMPOR uses the information obtained from the SPARQL query to construct aircraft operation plans, which we describe in the next section.

3.5 Aircraft Operation Plan Construction

SEMPOR now has a set of records from the SPARQL query from which it can construct aircraft operation plans with real resources that fit the requirements for operation elements requested by the operation planner in the prototype plan. The algorithm proceeds as follows.

First, divide the query results into separate lists based on the option. Each option represents a choice SEMPOR makes for a particular operation plan element. A prototype operation plan with two elements should have two options and two lists containing at least one record. This assumes at least one real resource exists that meets the requirements for the operation plan element. Next, pick one record from each list and generate an aircraft operation plan. Finally, repeat for every combination of records from each list.

For example, suppose we have a prototype operation plan that specifies two elements. The results of the SPARQL query give us aircraft1 and aircraft2 for option1; and aircraft3 and aircraft4 for option2. The algorithm creates operation plans from all combinations taken from these two lists. Specifically, SEMPOR creates the following combinations of operation plan elements:

- aircraft1 and aircraft3
- aircraft1 and aircraft4
- aircraft2 and aircraft3
- aircraft2 and aircraft4

Each of the combinations of elements above becomes part of an operation plan specification that gets returned by SEMPOR to the operation planner.

3.6 Real-Time Alerts

The ACP Message Service (ACPMS) provides real-time alerts to operation planners to keep them informed of changes in the status of both resources belonging to existing primary or contingency operation plans and to resources that could allow for new aircraft operation plans.

ACPMS monitors for changes to the ontology database. When the ontology database gets updated, ACPMS performs a number of SPARQL queries using the named entities in the primary and contingency operation plans.

Currently, ACPMS applies SPARQL queries to the ontology database to verify the following remain true;
• pilot available
• aircraft available
• aircraft has the required equipment configuration
• equipment remains effective for the weather conditions at the target
• aircraft has the same refueling demand

ACPMS alerts the operation planner to any deviations in the above conditions, as well as notifying the operation planner that more resources might have become available. The operation planner can then replan or select a different contingency plan.

4 Discussion

Our research efforts thus far show that we can

1. Represent mission-critical real-world resources with OWL and SWRL Semantic Web languages.

2. Capture operation planning knowledge and reasoning using OWL, SWRL, and SPARQL.

3. Use automated reasoning to infer implied relationships in an explicit data model of operation planning concepts and resources.

4. Query the inferred data model to obtain candidate operation plan elements.

5. Construct a primary operation plan and several contingency plans.

6. Recognize changes to operation plan elements and to the operation planning resource database that could affect existing operation plans or provide additional operation plan alternatives.

7. Alert operation planners to changes in near real-time.

5 Conclusions

We conducted research using an automated reasoner on an ontological data model of operation planning resources and concepts to determine the feasibility of the approach for ultimately developing a software tool able to meet the needs of operation planners. Among these needs include the ability to extract specific operation information for Air Coordination Plan (ACP), collect and organize data required for each mission, and verify the suitability and availability of all resources, both people and equipment, needed to carry out the air operation.

The results of our investigation show that our approach can indeed meet the needs of operation planners in their task of building and maintaining multiple operation plans. Through our research we have demonstrated that our ontological approach allows operation planners to generate and monitor aircraft operation plans having dependencies on aircraft and pilot availability, weather effects on aircraft and equipment, flight route distance, and fuel available.

In addition to satisfying these requirements, our approach has a number of additional advantages. First, the ontologies we use derive from AOCOI XML schema. This means our work supports AOCOI efforts to provide data transparency and sharing across both civilian and military Air Operations domains. Second, the ontological reasoning uses set theory in ways similar to how operation planners might combine pieces of information to construct operation plans. Third, the ontological models we develop capture the operation planning thought process in a form readable by both people and computers. Finally, having captured operation planning knowledge in an open data standard, our approach allows for the preservation and conveyance of operation planning decision-making in support of the W3C Decisions and Decision-Making Community Group [7].

6 References


