Multi-goal Pathfinding in Ubiquitous Environments: Modeling and Exploiting Knowledge to Satisfy Goals

Oudom Kem
Univ Lyon, MINES Saint-Étienne,
CNRS, Laboratoire Hubert Curien,
UMR 5516
F-42023, Saint-Étienne, France
oudom.kem@emse.fr

Flavien Balbo
Univ Lyon, MINES Saint-Étienne,
CNRS, Laboratoire Hubert Curien,
UMR 5516
F-42023, Saint-Étienne, France
flavien.balbo@emse.fr

Antoine Zimmermann
Univ Lyon, MINES Saint-Étienne,
CNRS, Laboratoire Hubert Curien,
UMR 5516
F-42023, Saint-Étienne, France
antoine.zimmermann@emse.fr

ABSTRACT

Multi-goal pathfinding (MGPF) is a problem of searching for a path between a start and a destination allowing a set of goals to be satisfied. We address MGPF in ubiquitous environments that accommodate cyber, physical and social (CPS) entities from smart objects to sensors and to humans. Given a MGPF problem in a pervasive environment, our approach aims at exploiting data from various resources including CPS entities located in the environment and external resources such as the Web to solve the problem. In this paper, we present a knowledge model for describing a ubiquitous environment integrating its spatial dimension, CPS entities it contains and its relevant resources. A global view of the approach is provided. We address particularly one of the challenges in MGPF, namely goal satisfaction problem, which consists of identifying through which entities a goal can be satisfied. Towards this aim, we design an ontology to formally model CPS entities, goals and their relations. We describe a method to exploit modeled knowledge in order to solve the goal satisfaction problem.

KEYWORDS

Multi-goal pathfinding, Ontology, Ubiquitous environment

ACM Reference format:
Oudom Kem, Flavien Balbo, and Antoine Zimmermann. 2017. Multi-goal Pathfinding in Ubiquitous Environments: Modeling and Exploiting Knowledge to Satisfy Goals. In *Proceedings of WI '17, Leipzig, Germany, August 23-26, 2017,* 4 pages.
https://doi.org/10.1145/3106426.3109054

1 INTRODUCTION

In literature, there are two common definitions of multi-goal pathfinding (MGPF). First, given a start and multiple goals, MGPF is defined as a problem of searching for a path from the start to each separate goal, resulting in multiple paths [5]. The second is defined as a traveling salesman problem in which the objective is to find a path from a start to a set of goals before reaching a destination, resulting in a single path [7]. The problem addressed in this paper is close to the second definition. The specificity of our problem is that there are constraints on the order of goals to satisfy, and a goal may be satisfied at multiple locations. For the purpose of clarity, consider the following example. A traveler, Alice, arrives at an airport. Alice wants to find a path to her departure gate. On the way, Alice has a number of goals she wants to achieve in the following order: transport her luggage, check-in and purchase a takeout for lunch.

We address MGPF in the context of ubiquitous environments accommodating cyber, physical and social (CPS) entities. A CPS entity can be, for instance, a place (e.g. café, restroom), a smart object (e.g. connected trolley), a sensor or even a human. Our approach to solving MGPF aims at exploiting data from CPS entities located in an environment as well as from external resources such as the Web. To understand the underlying motivation for such an approach, refer back to the example about Alice. Spatial information about the airport enables us to navigate in the airport and to find the gate. However, it is not sufficient. Knowledge about locations and entities of the airport is needed to determine where each of Alice's goal can be satisfied. As an example, information pertaining to each restaurant (e.g. availability) acquired from its website or reviews by other travelers (e.g. quality) on restaurants enable us to choose a restaurant best suited for Alice. Moreover, to find an optimal path, dynamic and up-to-date information is also required. For example, to get a trolley to transport Alice's luggage, instead of suggesting her to go to a trolley area that is at the opposite direction of the gate, it is possible to locate an available trolley nearby that was left by another person, thanks to the data from connected trolleys.

Goals can be satisfied by using entities. Referring to the example concerning Alice, her goals may be satisfied by using the following entities: transport luggage - *trolley*, check-in - *check-in booth or kiosk* and takeout lunch - *restaurant*. The challenge is how to determine through which entities a given goal can be satisfied. We address this goal satisfaction problem by providing an ontology to model the relations between CPS entities and goals such that we can exploit the structured knowledge to solve the problem.

Our contributions in this paper include a model for capturing the knowledge about spatial dimension, CPS entities and goals in a ubiquitous environment, an ontology for formally modeling the knowledge, and a knowledge-based method to solve the goal satisfaction problem.

The rest of the paper is structured as follows. First, we present a knowledge model for describing a ubiquitous environment along with the necessary ontology. Second, we provide an overview of

---

WI '17, August 23-26, 2017, Leipzig, Germany

ACM ISBN 978-1-4503-4951-2/17/08…$15.00
https://doi.org/10.1145/3106426.3109054

the approach and describe in details our method to solve the goal satisfaction problem. Third, we conclude the paper.

2 ENVIRONMENT DESCRIPTION MODEL

Spatial information is necessary but not sufficient to determine at which location a goal can be satisfied. Up-to-date and qualitative information is also required to find an optimal path. Theses reasons motivate our approach to exploit data from various resources. In this section, we present a model for describing an environment incorporating the concept of resources and goals.

2.1 The knowledge model

In this model, we represent the spatial dimension of a real-world ubiquitous environment as a graph of locations, denoted by SG, by defining each location as a node. A directed edge between two adjacent nodes n and n' is defined if the location represented by n' is *directly* accessible from that by n.

Definition 2.1. An *environment* at a given time is defined as a tuple $E_t = (SG, HE, CPSE, R)$ where:

- SG is a search graph defined as $SG = \langle L, C \rangle$ where L is a finite set of nodes representing locations in E and $C \subseteq L \times L$ is a set of edges representing connections between locations
- HE represents an organizational hierarchy of E. It is a tree whose elements correspond to hierarchy entities (e.g. terminals) of E and the child relation indicates sub-hierarchy entities (e.g. zones within a terminal). Locations are grouped under hierarchy entities, so the leafs of HE are directly connected to locations
- $CPSE$ is a finite set of CPS entities located in E
- $R = (r_n)_{n \in CPSE \cup C}$ is a finite set of resources providing information about a CPS entity or giving information on how to move between locations. An example of a resource can be a website or an API to sources of data collected from CPS entities.

In this work, we associate a goal g to an activity a^g. We say that an entity $cpse \in CPSE$ located in $l \in L$ *satisfies* g if a^g can be carried out through $cpse$. To determine if an entity can satisfy a goal, we need information about the entity, specifically, the activities that it supports. To capture such knowledge, we incorporate also the concept of activities and its relations with CPS entities into our model.

2.2 Space-Goal Ubiquitous Environment Ontology

To formally describe an environment, we propose an ontology, entitled Space-Goal Ubiquitous Environment Ontology[1], abbreviated to `sgue`, using OWL (Web Ontology Language). Fig. 1 shows the schema of the ontology. The main reason for using an ontology is that we aim at exploiting the semantics of modeled knowledge to solve the goal satisfaction problem. It is worth mentioning that one of our objectives is to allow descriptions of different environments to be compatibly integrated together. Such integrated description can be used to solve MGPF problems of different scales. The shared models and vocabularies offered by ontologies are one of the keys to achieve these objectives.

We define `sgue:CPSEntity` as the class of *CPSE*. A CPS entity has a set of resources providing information about that entity. For example, a resource relevant to a restaurant can be the restaurant's website; a resource of an elevator sensor can be an API to a sensor data store. We use the property `sgue:hasRelevantResource` to relate a CPS entity to a resource.

The class `sgue:Location` represents L. A location $l \in L$ contains a set of CPS entities. We use `sgue:containsEntity` to capture the relations between l and an entity located in l at a given point in time. As an example, l contains five sensors, two gates and five trolleys (`sgue:CPSEntity`) at time t. Each location is directly under a hierarchy entity (`sgue:HierarchyEntity`). This type of relationship is described by the `sgue:isUnder` property. A location can be directly connected to other locations. The class `sgue:Connection` represents a direct connection from a location to another. Each connection is attributed with a finite set of resources providing information about the path between the two locations. The property `sgue:hasConnectingResource` relates a connection to such a resource. For instance, a direct connection from l to $l' \in L$ is attributed with a set of resources $R^{l-l'} \subset R$ providing information about the path from l to l'.

The class `sgue:HierarchyEntity` represents *HE*. A hierarchy entity may directly contain a finite set of locations. The relationship between a hierarchy entity and a location is described by the `sgue:containsLocation` property. A hierarchy entity may be further divided into a finite set of hierarchy entities of a lower level. An example can be Terminal 1 consisting of Zone 1, 2 and 3. To express these connections, we use the property `sgue:containsHierarchy`. Furthermore, a hierarchy may also be directly under another hierarchy of a higher level (parent hierarchy). Taking the preceding example, Terminal 1 can be under an Airport. To relate a hierarchy entity to its parent hierarchy entity, the property `sgue:belongsTo` can be used. Such relationship is non-existent if the hierarchy entity is of the highest level (i.e. root). A hierarchy entity may also cover a range of activities. These are the activities supported by CPS entities under its coverage. The rationale behind describing such activities in the description of each hierarchy entity is to facilitate the subgraph extraction in pathfinding process (Section 3.1) and avoid having to explore every entity in *SG*. These relations can be described using the property `sgue:coversActivity`. A hierarchy entity $he \in HE$ may be connected to other hierarchy entities. To describe how he is directly connected to $he' \in HE$ where he and he' are at the same level, we use the class `sgue:ConnectingPoint`. A connecting point consists of: (1) a connected hierarchy entity he', (2) a connection c, instance of the class `sgue:Connection` where $l^{exit} \in L$ the origin of c is an exit point of he and l^{exit} is under the coverage of he; the destination of c is an entry point $l^{entry} \in L$ of he' where l^{entry} is under the coverage of he'; a finite set of resources $R^{he-he'} \subset R$ for the path from l^{exit} to l^{entry}.

The class `sgue:PotentialActivity` represents activities that *can* be carried out in a given environment, independent of whether they are or will be carried out. A CPSE entity *cpse* may support a set of potential activities. In other words, through *cpse*, a set

---

[1] https://partage.mines-telecom.fr/index.php/s/SlhfVYkLsdLA2QR

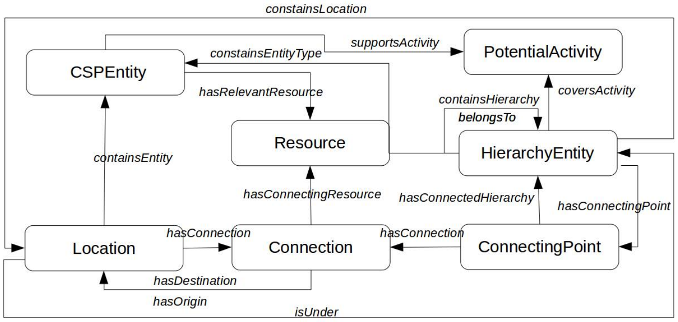

Figure 1: Space-Goal Ubiquitous Environment ontology

of potential activities may be carried out. This is captured by the `sgue:supportsActivity` property.

The concept of *Activity* is widely used in various domains and fields of research. Existing activity ontologies such as [2] and [3] model activities for their respective domains, while [1] attempts to provide a generic ontology that captures the common core of activities. The concept *PotentialActivity* defined in our ontology does not represent the actual activities as in [1], [2] or [3]. Potential activities stress on the *ability* to be performed, independent of whether they are actually carried out or not. For example, an instance of `sgue:PotentialActivity` is *eating a pizza*, while the actual activity could be *eating a pizza in a restaurant X at Y time*. The relationship between an actual activity a and its potential activity a' can be viewed as a *realizes* a' at a given point in time.

To demonstrate an instance of `sgue`, we specialize it to model CPS entities and potential activities in a smart airport. The Smart Airport Activity ontology[2] (`saa`) was defined following an analysis of airport travelers' activities described in [6]. This analysis provides a classification of the common activities that travelers do at an airport. Based on this classification, we determined the CPS entities that may support each activity. It is essential to note that the classes representing CPS entities and activities in this ontology are by no means exhaustive nor compatible with all kinds of airports. It was, however, defined in a manner such that it can be reused and extended to capture the specifics of a particular airport as well as the desired level of granularity of potential activities. Fig. 2 illustrates an extracted view of the ontology. A CPS entity can be a physical, social and/or cyber entity. These variations are captured by the four direct and indirect subclasses of `sgue:CPSEntity` as follows:

- `saa:PhysicalEntity` represents physical entities - any entity that occupies a physical space.
- `saa:CyberPhysicalEntity` covers physical entities that are equipped with cyber abilities.
- `saa:SocialEntity` is the class of entities that possess social capabilities - the ability to interact with other entities.
- `saa:CyberPhysicalSocialEntity` is the class of physical entities that have both cyber and social abilities.

The `saa` allows different types of CPS entity to be associated with their relevant classes of potential activity at the ontology level. This association is captured by using existential restrictions

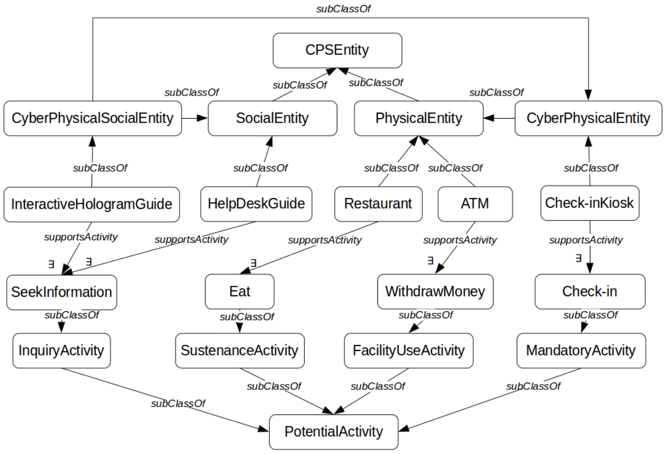

Figure 2: A partial view of Smart Airport Activity Ontology

on the property `sgue:supportsActivity`. For example, to state that instances of the class `saa:HelpDeskGuide` are the those that support instances of the class `saa:SeekInformation`, we make use of the following axiom:

$$HelpDeskGuide \equiv \exists supportsActivity.SeekInformation$$

In the next section, we present how modeled knowledge of an environment is used to solve MGPF.

3 GOAL SATISFACTION

By modeling an environment as previously described, we can define a MGPF as follows:

Definition 3.1. MGPF $= (E_t, n_o, n_d, G, CR)$ where E_t is a representation of an environment at time t, $n_o \in L$ is a node representing a start location, $n_d \in L$ is a node representing a destination, G is an ordered list of goals to satisfy, and CR is a set of criteria for evaluating a path. Criteria are problem-specific. For instance, a criterion can be distance, price or duration. A problem is solved when an optimal path is found. A path is a list of locations through which every goal can be satisfied in the given order. A path is optimal if it has a minimum cost evaluated based on CR.

3.1 Approach overview

Our approach to pathfinding is divided into two main steps: (1) description generation and (2) search. The first step of the approach is to generate a description of an environment of interest. This step is conducted once at beginning, and the description can be used to solve various MGPF problems in the described environment. In a concrete implementation, the description is constructed by using `sgue`. The description, being based on RDF and IRI (Internationalized Resource Identifier), can be stored in a centralized manner or distributed in an environment or on the Web. In addition, it is also possible to enable different actors of the system that are responsible for different CPS entities to provide their own description of the entities. This allows people in charge of the entities to provide information specific to each entity. Furthermore, `sgue` models generic relationships between CPS entities and potential

---

[2] https://partage.mines-telecom.fr/index.php/s/jUfcd4dS8yRXhQX

activities. However, CPS entities and potential activities differ from one environment to another. Thus, to describe such relationships of a specific environment, it is necessary to extend `sgue` to capture the knowledge of the environment (e.g. Smart Airport Activity Ontology).

Having generated the description, the next step, the search process uses knowledge available in an environment's description to solve MGPF in the environment. This process is executed for each MGPF problem. It is divided into three main steps: subgraph extraction, partial plan generation and search.

Subgraph extraction. Taking a MGPF problem as an input, this first step identifies relevant locations in *SG* and extracts a subgraph containing such locations. A location is relevant if it contains at least a CPS entity through which one or more goals may be satisfied. The rationale behind subgraph extraction is to reduce the size of the graph over which we search for an optimal path to that of a graph containing only relevant nodes.

Partial plan generation. A goal may be satisfied at multiple locations. To determine which location is optimal for satisfying each goal, we need to take into account the following factors: the cost for moving from the current goal's location to the next goal's location and the quality of CPS entities in the next goal's location. The quality of a CPS entity is evaluated using qualitative information from resources. To represent the space of this problem, we construct a partial plan, denoted by π, by using G and the relevant locations extracted from the preceding step. We use the definition of a partial plan in [4].

Search. A partial plan π is an abstract graph that is constructed based on a *SG*. To compute the path, a search algorithm can be applied to perform the search over π. Many search algorithms exist such as Dijkstra's algorithm, uniform-cost search and A*. The separation between *SG* and *HE* in our model makes it possible to apply different search algorithms such as heuristic ones that search using domain knowledge (i.e. *HE*) and the ones using only information in *SG*.

3.2 Exploiting knowledge to satisfy goals

When describing a hierarchy entity, we need to include the types of activities supported the hierarchy entity. To acquire this information, each entity under the hierarchy is explored. The issue here is how to determine the types of activities supported by an entity, the goal satisfaction problem. To determine activities an entity *cpse* can support, we use the knowledge in the description of *cpse*, as demonstrated in Algorithm 1. This algorithm takes an inputs the IRI of *cpse*. In the case where the potential activities supported by *cpse* are provided in its description, we can use that knowledge directly to determine the types of activities (Algorithm 1: line 2-5). In addition, we also make use of the ontology to identify the types of activites *cpse* supports (Algorithm 1: line 6-12). The associations between subclasses of `sgue:CPSEntity` and subclasses of `sgue:PotentialActivity` are defined at the ontology level through existential restrictions on the property `sgue:supportsActivity`. We exploit such connections to identify the classes of activities *cpse* supports.

Algorithm 1 getClassesOfSupportedActivities(*cpse*)

```
1:  retrieve the description of $cpse$
2:  $supportedActivities \leftarrow$ get the objects of property sgue : supportsActivity where the subject is $cpse$
3:  for each $activity$ in $supportedActivities$ do
4:      add the class of $activity$ to classesOfSupportedActivities
5:  end for
6:  $cpseClass \leftarrow$ get the class of $cpse$
7:  retrieve the description of $cpseClass$
8:  for each $superClass$ of $cpseClass$ do
9:      if $superClass$ is an anonymous class that supports instances of a subclass of sgue : PotentialActivity then
10:         add subclass of sgue : PotentialActivity to classesOfSupportedActivities
11:     end if
12: end for
13: return classesOfSupportedActivities
```

4 CONCLUSION

This paper presents an overview of an approach to address MGPF in a ubiquitous environment. The approach exploits data from various resources from entities located in a given environment to the Web. We provided a knowledge model and an ontology to describe an environment integrating different aspects, allowing the description to be used for solving MGPF. We tackled one of the core issues we encountered in MGPF, the goal satisfaction problem. To address this issue, we proposed an ontology to formally model the connections between CPS entities and activities supported in an environment. Our method solves the goal satisfaction problem by using the described knowledge of CPS entities and the ontology.

In practice, we could allow people to provide a description for each of their entities. People can incorporate additional knowledge specific to their entities using other ontologies. Such knowledge may allow us to associate entities to more specific activities. However, this would introduce heterogeneity in granularity of knowledge (e.g. activities) and in ontology, which requires extension of the reasoning process used in goal satisfaction and/or employment of existing semantic reasoners.

REFERENCES

[1] Amin Abdalla, Yingjie Hu, David Carral, Naicong Li, and Krzysztof Janowicz. 2014. An ontology design pattern for activity reasoning. In *Proceedings of the 5th International Conference on Ontology and Semantic Web Patterns-Volume 1302*. CEUR-WS. org, 78–81.

[2] Tiziana Catarci, Benjamin Habegger, and Antonella Poggi. 2006. Intelligent user task oriented systems. *Personal Information Management: Now That WeâĂŹre Talking, What Are We Learning?* (2006), 20.

[3] Mark S Fox, John F Chionglo, and Fadi G Fadel. 1993. A common-sense model of the enterprise. In *Proceedings of the 2nd Industrial Engineering Research Conference*, Vol. 1. 425–429.

[4] Malik Ghallab, Dana Nau, and Paolo Traverso. 2016. *Automated Planning and Acting*. Cambridge University Press.

[5] K. L. Lim, L. S. Yeong, S. I. Ch'ng, K. P. Seng, and L. M. Ang. 2014. Uninformed multigoal pathfinding on grid maps. In *2014 International Conference on Information Science, Electronics and Electrical Engineering*, Vol. 3. 1552–1556. https://doi.org/10.1109/InfoSEEE.2014.6946181

[6] Xuan Liu, John M Usher, and Lesley Strawderman. 2014. An analysis of activity scheduling behavior of airport travelers. *Computers & Industrial Engineering* 74 (2014), 208–218.

[7] Rajesh Matai, Surya Prakash Singh, and Murari Lal Mittal. 2010. Traveling salesman problem: An overview of applications, formulations, and solution approaches. *Traveling Salesman Problem, Theory and Applications* (2010), 1–24.