A Realistic Multi-Modal Cargo Routing Benchmark

Tony Allard[1] and **Charles Gretton**[2*]

[1]Joint and Operations Analysis Division
Defence Science and Technology Organisation, Adelaide, Australia
tony.allard@DSTO.defence.gov.au

[2]Optimisation Research Group
NICTA, Canberra, Australia
charles.gretton@nicta.com.au

Abstract

We describe a multi-modal cargo routing (MMCR) domain for modelling military logistics planning problems. These are transport optimisation problems that feature timing constraints, concurrency, capacitated resources, and action costs. We have developed a PDDL domain model, and have released a collection of problem instances along with a software tool to aid in the design and generation of new problem instances. Small instances of this domain stretch the capabilities of existing automated planning procedures, and larger realistic instances are beyond the capabilities of existing automated planning systems. We anticipate that scalable solution procedures for this domain will follow in the footsteps of systems, such as OPTIC and TIMIPLAN, which combine heuristic search concepts with mathematical programming optimisation tools.

Introduction

We developed a new realistic benchmark for the problem of multi-modal cargo routing. Our benchmark is based on transport problems that occur in defense operations planning. Many of the challenging aspects of such problems are also relevant in commercial civilian settings. Such aspects include time-dependent goal achievement, and actions which execute over time, are costly, and are performed concurrently. Timing constraints on goal achievement and resource availabilities mean that problems frequently exhibit *required concurrency*[1] (Cushing et al. 2007) in order to achieve feasible solutions. Cargoes reside in transport assets and at places, which are both capacitated mediums. They are transported between places concurrently using a variety of assets which are available for operations during specific time intervals. The initial collection and delivery must occur during specific time intervals. In a typical plan we have that multiple distinct assets—terrestrial, nautical and aeronautical—must be used to transport cargoes to their destinations. Lastly, the loading, un-loading and re-loading of assets must be scheduled during time intervals when the relevant assets are available.

We have developed a Planning Domain Description Language (PDDL) domain model for the MMCR domain. We have also developed problem modeller in the Java programming language, and we have made that available on GitHub (https://github.com/Optimised/MMCRDomainPDDLGenerator). That software includes a convenient graphical user interface for examining and modifying our test scenarios, and writing routing problems to PDDL. We have also released a number of example scenarios. These include a highly simplified toy problem, and then problems that range in complexity up to large sophisticated scenarios of the sort one encounters in real-world operations planning.

The paper is organised as follows. We first review logistical benchmarks which have featured in the AI planning literature, and discuss our benchmark in the context of the transportation sciences and operations research literature. We then describe the MMCR domain, and provide the details of a small example problem with accompanying PDDL. We briefly report on features of MMCR which make it challenging. Finally, toward future work we discuss important features of real-world scenarios which we have yet to include in the PDDL model.

Background and Related Work

Logistical domains have for some time been a focus of the automated planning community. The first two International Planning Competitions, held in 1998 and then 2000, featured the classical LOGISTICS benchmark (McDermott 2000). This models an uncapacitated transport problem, without any temporal aspects, and where all actions have unit cost by convention. The problem posed by a typical LOGISTICS instance is to relocate "packages" from source locations to their destinations. Trucks can relocate packages to and from airports and other inner city locations. Airplanes can relocate packages between airports. All vehicles and locations have infinite capacity, actions are supposed to execute instantaneously, and all vehicles are unconstrained in their availability. This classical LOGISTICS benchmark appeared again in the IPC of 2011, and on that occasion featured non-uniform action costs.

Other early transportation domains, dating from 2002 in-

*NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

[1]i.e. it can be that no plan is forthcoming unless actions are executed concurrently.

clude (i) ZENO: transporting human cargo by air while trading distance for fuel consumption in a highly-discrete setting, (ii) DRIVERLOG: a simple transport domain that also models some aspects of crew availability; and (iii) DEPOTS: a bespoke product of the classical AI benchmark BLOCKSWORLD and the above LOGISTICS domain. Overall, none of these domains feature some key aspects of important transportation planning scenarios, such as concurrency, action durations and rich numeric action costs.

More recently the automated planning literature has described transport domains that are more familiar to computer science and operations research. The Travelling Purchase Problem (Ramesh 1981) TPP appeared as a planning benchmark in 2006, in this case featuring capacitated multi-depot and multi-vehicle varieties of this problem with delivery deadlines. The same year also marked the introduction of the TRUCKS domain (Dimopoulos et al. 2006), which is a version of LOGISTICS with durative actions, non-uniform action costs and delivery deadlines. Most recently, in 2014 a temporal version of DRIVERLOG was introduced, which poses a makespan minimisation problem where the objective is to schedule crews in order to route deliveries. The DRIVERLOG SHIFT temporal variant from Coles et al. (2009) features required concurrency, where durative actions are used to model each driver's status – i.e. *resting* and *working*. Another temporal version of DRIVERLOG with *deadlines*—e.g. goal facts must be achieved within a time limit—is described by Marzal, Sebastia, and Onaindia (2014).

The development of progressively more realistic logistical benchmarks in planning has followed trends and augmentations of PDDL. From 1998 onwards that formalism has became the *de facto* language in which to describe planning problems and domains. At its inception PDDL was quite limited, admitting compact descriptions of domain and problem theories which could be compiled straightforwardly to the propositional STRIPS representation of Fikes and Nilsson (1971). Since then, by a process of progressive enhancement—e.g. (Fox and Long 2003; Edelkamp and Hoffmann 2004; Gerevini and Long 2006)—PDDL has at version 3.1 become the somewhat formidable and powerful language that we find ourselves using today (Kovacs 2011). PDDL is now sufficiently expressive to model realistic defense transport scenarios. The benchmark that we developed in the following sections uses key features from recent versions of the PDDL language, such as *durative actions* (Fox and Long 2003), *timed initial literals* (Edelkamp and Hoffmann 2004), *numeric fluents* and *action costs*.

From the perspectives of *transportation sciences* and *operations research*, our benchmark can be classified as a rich *vehicle routing problem* (Golden, Raghavan, and Wasil 2010), many varieties of which are treated in excellent detail by Goel and Gruhn (2008). Our benchmark also express similarities with operational freight logistics problems (Crainic and Laporte 1997). More specifically, our benchmark looks most like a pickup-and-delivery problem, which in its canonical form can be solved at scale using an *adaptive large neighbourhood search* (ALNS) heuristic (Ropke and Pisinger 2006). However, our benchmark exhibits a number of nonstandard constraints, such as vehicle-dependent travel times/costs, quantity-dependent load and unload times, concurrency constraints, and asynchronous exchanges at capacitated places. We are not aware of a single monolithic vehicle routing platform that addresses all of the above, however an implementation of ALNS which should be readily adapted to treat our non-standard constraints is provided by INDIGO, a hybrid solver which integrates constraint propagation and local search (Kilby and Verden 2011). Finally, it is worth noting that we are not the first to consider leveraging AI planning technology in a transportation science setting. García et al. (2013) have developed a bespoke hybrid of classical optimisation, AI planning, and in earlier work *ant colony optimisation* to solve intermodal transportation problems.

The Planning Domain

MMCR is characterised by the movement of cargo between locations via a transport network. An example network is displayed in Figure 1, and is comprised of locations (shown as circles), and routes (shown as the edges between locations). The figure also shows a truck and an aircraft capable of traversing the routes as indicated by solid and dashed lines between locations. Each vehicle is capable of travelling to locations as defined by the respective sub-graph of the problems transport network.

Figure 1 also shows cargoes which are moved through the route network between marked source and destination locations. In practice, operation planners rarely reason about a network in its entirety, but rather some fragments of the network which hopefully facilitate meeting localized cargo movement goals. Part of the difficultly of planning stems from the fact that the networks of multiple individual vehicles must be reasoned about in order to effect cargo deliveries. Cargoes can represent multiple products, including food, potable water and mechanical components. The chief elements in a cargo routing scenario are thus the transportation assets, the model of their respective route networks, and the cargoes to be delivered. Network locations include cities, supply depots, warehouses, defense establishments, etc.

Transporting Cargo

In order to reach its destination, cargo must usually be relocated between origin, intermediate and destination locations using different modes of transportation (i.e. air, land, sea).

Each cargo item is associated with a time window, specifying the period within which it can be moved. The bounds of this time define when the cargo is initially available to be moved, through to when it is required to be at the destination.

We have a rich heterogeneous routing problem, in which different vehicles may be capable of reaching distinct subsets of locations. Distinct vehicles have different costs, distances and speeds when travelling between locations. In a valid plan, vehicles are often required to make multiple trips along the same network arc. We have modelled the capabilities of different vehicles to transit routes of various lengths, and would like to emphasize that in our target problems it

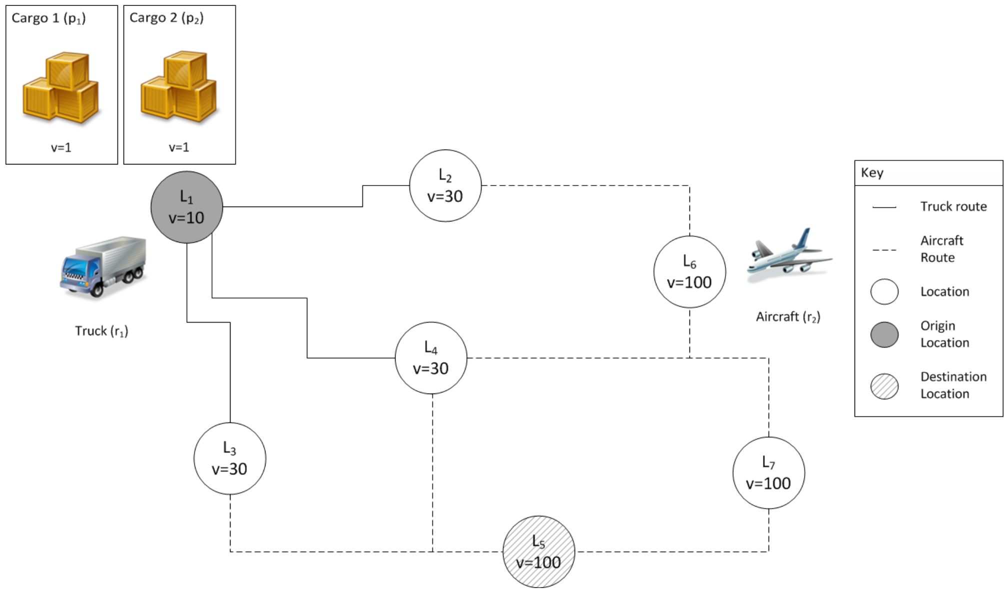

Figure 1: Example of a small multi-modal cargo routing problem.

is vital that we have modelled actions as durative and not instantaneous events.

In typical scenarios there is no single continuous path for a vehicle to effect the movement of a cargo from its origin to its destination. A plan will have the cargoes transit, and transportation asset movements must be carefully scheduled to make such transits possible. Planning must identify mutually reachable locations to conduct exchanges of cargoes between assets, and ensure that where exchange is asynchronous that the transit location has sufficient storage capacity.

For example, a deployment scenario requires cargo to be moved from unit stores within a military base to a consolidation point, before on-forwarding into a theatre of operations.[2] The principal move from the consolidation point is performed by a large strategic asset – e.g. aircraft or ship. Movement from unit stores however is typically achieved using available land, rail and air assets. Transportation between unit stores (origin) and the theatre of operations (destination) requires the use of multiple vehicles, and an intermediate point for cargo exchange. Indeed, in some problem instances multiple exchanges are required.

When transporting cargo there are associated times for loading and unloading onto a particular vehicle, further emphasising the need to model action executing times. Usually the time taken to load/unload cargo is a function of: (i) the type of vehicle, (ii) location, and (iii) quantities of the cargo. Dependence on (i) and (iii) is relatively self-evident. Dependence on (ii) is highly complex, relating to the type and availability of material handling equipment (MHE) at the location, as well as other factors such as weather, terrain, available personnel, etc. Our current modelling greatly simplifies these aspects of the underlying scenarios. We define a fixed amount of time for loading and unloading one unit of cargo on the vehicle for each location.

Cargoes and Capacities

In a valid plan cargoes can be stored at locations or temporarily in vehicles. A cargo can represent multiple types of products, each of differing weights and dimensions. For the scenarios we consider, we model these capacitive aspects of cargoes in terms of the number of pallets they can hold. This unit of measurement is referred to as *pallet equivalent units* (PEU). Vehicles and locations have a finite capability to hold a fixed number of PEUs. Each item of cargo is an indivisible quantity specified by the number of PEUs the item will consume.

The above capacity modelling constrains what locations can feasibly be used to transit cargoes, and what assets can be used. Real-world deployments often face problems where intermediate points become overladen with cargo, and where high-capacity vehicles are overused. In the case of the latter, there is little margin for error, and the consequences of a single vehicle failure/delay can be amplified where high-capacity vehicles are completely filled and other available lower-capacity assets are underfilled. Continuing our earlier deployment scenario, we can provide further intuition for the types of constraints that occur in practice by focusing on the consolidation point. If sufficiently many assets are scheduled to make overlapping deliveries at that point, then the cargo holding capacity can be exceeded and such a schedule is thus infeasible. Two solutions might be used to address that infeasability: (i) schedule the movement and on-forwarding of cargo so that units are staggered, and the amount of cargo at the location is kept below the maximum; or (ii) use multiple locations as intermediate points and em-

[2] A *theatre* is a geographic territory where the defence forces are operating, external to the national territory of that force.

ploy more vehicles for the principal movement. The benefits and trade-offs of these proposals would need to be considered within the wider context of the goals of the problem and cost of the solutions.

Asset Availability

Just as we give a time window for the period in which a cargo can be transported, time windows also describe each asset's availability to move cargo. Asset availability windows are used to model scenarios in which commercial transport contractors are employed to make deliveries during a specified interval of time. Typically a portion of assets are available more-or-less indefinitely, and are therefore highly unconstrained in this regard.

Costs and Optimisation Criteria

Each movement of cargo requires the utilisation of one or more transportation asset for the actions of loading, moving and unloading. Such utilisation incurs a cost dependent on the type of vehicle, and is usually linear in proportion to the duration of its employment. The total cost of a plan is the sum of all vehicle utilisation costs. Our optimisation criteria are therefore to minimise costs, while respecting all capacity and time window constraints.

Required Concurrency

Rich temporal actions are required to model the real-world interactions of vehicles within the MMCR domain. Moreover, the interactions between capacities and time windows can force all plans to exhibit concurrent execution of actions. For example, two or more vehicles can be required to present simultaneously to unload cargo at a single location. Where at least two such deliveries are on the critical path,[3] concurrent executions are required for legal transits.

Indeed due to capacitive and durational constraints, problems from our benchmark can require concurrency in the sense discussed by Cushing et al. (2007). Describing the rich set of constraints require a temporally expressive language, such as PDDL version 2.2. We cannot use earlier problem description formalisms, such as those used by decision epoch planners including ZENO (Penberthy and Weld 1994), and systems which support fragments of PDDL 2.1, such as TGP (Smith and Weld 1999) and also the system VHPOP (Younes and Simmons 2003), which relies on atemporal causally valid planning to obtain incumbent solutions which may admit schedules. Such systems do however exhibit a general planning approach which may be fruitful for MMCR planning. Particularly appealing is the idea of using a classical-epoch/causal planning approach interleaved with temporal reasoning, as was also exhibited recently by the temporal SAT-based PDDL 2.1 system of Rankooh and Ghassem-Sani (2013).

In additional to accurately capturing the actual scheduling constraints in MMCR, temporal modelling also enables the type of coordination discussed by Coles et al. (2008), where concurrent executions of actions further assists in the development of more efficient plans. For example, the movement of a vehicle to a particular location may execute concurrently with the movement of another. Where cargoes are to be exchanged between multiple vehicles, it is preferable to move those towards the transit location concurrently, and therefore not incur an overly inflated makespan.

[3] The critical path is defined as a sequence of events with no slack time – i.e. critical-path actions required to achieve a sub-goal can neither be delayed nor expedited within the schedule.

An Instance

We provide the details on an MMCR problem that models a cost-sensitive, constrained, resource allocation problem (Allard and Shekh 2012). Below we provide PDDL extracts for the domain and problem descriptions associated with this problem. An admissible solution should allocate vehicles to the movement of cargoes while respecting defined constraints. A solution could therefore be described as a schedule for each vehicle, that encapsulates the loading, movement and unloading of cargoes, to meet problem specifications. It is worth noting that a solution schedule can be either fixed, or flexible in its action timings. A schedule which commits executions to time windows rather than specific times would indeed be preferable, as it would usually be inherently more flexible. In this section we define an exemplar problem with accompanying PDDL for the MMCR domain and that problem. The PDDL is shortened for space considerations, a full specification accompanies the problem generator on GitHub.

Example

Our exemplar takes the network of Figure 1 and adds further details about action costs, durations and asset capacities, in Tables 1 and 2 and then below in PDDL extracts. The route network is characterised by an undirected graph, $G = \langle L, E \rangle$. The graph contains a set of nodes or vertices, L, connected by edges as defined by an adjacency matrix E. The set of nodes, $L = \{l_1, \ldots, l_{|L|}\}$, represent the geographical locations, l_1 to l_7. Each location l_i has a corresponding capacity v, which specifies the maximum number of PEUs that can be stored at that location (shown for each location in Figure 1). The connectivity between locations in the graph is different for each vehicle. The problem description captures a set $E = \{E_{r_1}, \ldots, E_{r_{|R|}}\}$ of adjacency matrices, where each matrix E_{r_i} gives reachability and travel times between each location,

$$\begin{bmatrix} 0 & \cdots & d_{l_1, l_{|L|}} \\ \vdots & \ddots & \vdots \\ d_{l_{|L|}, l_1} & \cdots & 0 \end{bmatrix}$$

for a particular resource, r_i (see Table 3). An infinite value indicates that no path exists between the two locations for the vehicle i. Given the default closed world assumption used in PDDL, unconnected locations can be omitted from the problem specification. They are shown in Table 3 for the purposes of completeness.

Cargo, $P = \{p_1, \ldots, p_{|p|}\}$, defines the set of cargoes within the problem. Cargo p_i is defined by its size, v (measured in PEUs), origin location o, demand location d, and a time window given by the starting s and ending e

Table 1: Vehicle specification for MMCR exemplar.

Property		Truck (r_1)	Aircraft (r_2)
v		2	2
o		l_1	l_6
L/U	l_1	1	n/a
	l_2	1	1
	l_3	1	1
	l_4	2	1
	l_5	n/a	1
	l_6	n/a	1
	l_7	n/a	1
c		10	100

Table 2: Cargo specification for MMCR exemplar.

Property	Cargo 1 (p_1)	Cargo 2 (p_2)
v	1	1
o	l_1	l_1
d	l_5	l_5
s	1	1
e	11	11

times. Therefore an item of cargo is described by the tuple, $\langle v, o, d, s, e \rangle$ (see Table 2).

An MMCR problem models a heterogeneous set of transport resources, $R = \{r_1, \ldots, r_{|R|}\}$, vehicles that are initially located at specified transport network nodes. A vehicle can be categorised by its capabilities and modelled by the tuple, $r_i = \langle v, o, L, U, c_{r_i} \rangle$ (see Table 1). Again v defines the capacity of r_i, which determines the maximum number of PEUs r_i could carry. Term o is the initial location of the vehicle. Both L and U specify the set of load and unload times required for vehicle r_i to load and unload one PEU of cargo at each location. Finally c specifies the cost associated with utilising r_i for one unit of time.

Table 3: Reachability matrices for resources r_1 and r_2 (any omitted routes are deemed unreachable). An overview of reachability for each resource is also shown in Figure 1.

Route	Travel Time (hours)	
	E_{r_1}	E_{r_2}
$L_1 \longleftrightarrow L_2$	1	∞
$L_1 \longleftrightarrow L_3$	1	∞
$L_1 \longleftrightarrow L_4$	2	∞
$L_2 \longleftrightarrow L_6$	∞	1
$L_3 \longleftrightarrow L_4$	∞	2
$L_3 \longleftrightarrow L_5$	∞	2
$L_4 \longleftrightarrow L_5$	∞	1
$L_4 \longleftrightarrow L_6$	∞	2
$L_5 \longleftrightarrow L_7$	∞	2
$L_6 \longleftrightarrow L_7$	∞	1

This problem specifies that there are two units of cargo, p_1 and p_2, that are required to be transported between locations l_1 and l_5 – i.e. depicted in Figure 1 by the shaded and striped circles respectively. Measuring time in hours, both units of cargo can be moved from time $t = 1$, and must arrive at the destination no later than time $t = 11$ (see Table 2). Table 3 (and Figure 1) shows that neither vehicle can travel between l_1 and l_5. Therefore cargoes must be exchanged at a mutually reachable location. Possible intermediate locations are l_2, l_3 and l_4. The journeys between origin and destination through these locations take 5, 3 and 3 hours respectively. At first glance it might appear that using either l_3 or l_4 to exchange cargoes would seem appropriate. Given that r_2 needs to travel either 4 or 2 hours to get from its starting location of l_6 to either l_3 or l_4 respectively, l_4 seems like the right choice. This adds an extra hour to the journey of r_1 (for moving to l_4 as opposed to l_3). Considering r_2 possesses a higher utilisation costs, favouring a reduction in its travel time (over r_1) seems sensible.

It can be seen from Table 1 that r_1 requires two hours at location l_4 to unload one PEU of cargo. This brings the total time to concurrently exchange all cargoes between vehicles to 5 hours. Adding transit time, and a 2 hour load and unload time at the origin and destination, brings the total plan duration to 12 hours (with a cost of 780). This means cargo will not be delivered within the specified time window. Using l_3 as a cargo exchange point, can reduce the duration of the exchange to 3 hours. This is because cargo can be unloaded from r_1 in only one hour at l_3. This could be due to additional MHE, or personnel at that location, as compared with l_4. The total duration of such a plan is 10 hours including 3 hours for transit and time for loading and unloading at origin and destination. This plan meets the required goals, however at a higher total cost of 1050. This is an example of how concurrency can be required in the plan to enable goal satisfaction.

PDDL Domain Model

```
(define (domain multi-modal-cargo-routing)
  (:requirements :typing :equality
    :fluents :action-costs
    :durative-actions
    :timed-initial-literals
    :duration-inequalities
  )
  (:types
    CONTAINER CARGO - object
    VEHICLE LOCATION - CONTAINER
  )
  (:predicates
    (at ?x - (either VEHICLE CARGO)
      ?y - LOCATION)
    (in ?x - CARGO ?y - VEHICLE)
    (ready-loading ?x - VEHICLE)
    (available ?x - (either VEHICLE CARGO))
  )
  (:functions
    (remaining-capacity ?x - CONTAINER)
    (travel-time ?x - VEHICLE
      ?y ?z - LOCATION)
    (size ?x - CARGO)
```

```
    (load-time ?x - VEHICLE ?y - Location)
    (unload-time ?x - VEHICLE
      ?y - Location)
    (cost ?x - VEHICLE)
    (total-cost)
  )
  (:durative-action load
    :parameters (?x - VEHICLE ?y - CARGO
      ?z - LOCATION)
    :duration (= ?duration (*
      (load-time ?x ?z) (size ?y)))
    :condition
      (and
        (over all (at ?x ?z))
        (at start (ready-loading ?x))
        (at start (at ?y ?z))
        (at start (<= (size ?y)
          (remaining-capacity ?x)))
        (at start (available ?y))
        (over all (available ?x))
        (over all (available ?y))
      )
    :effect
      (and
        (at start (not (at ?y ?z)))
        (at start (decrease
          (remaining-capacity ?x)
          (size ?y)))
        (at start (not (ready-loading ?x)))
        (at end (increase
          (remaining-capacity ?z)
          (size ?y)))
        (at end (in ?y ?x))
        (at end (ready-loading ?x))
        (at end (increase (total-cost)
          (* ?duration (cost ?x))))
      )
  )
  (:durative-action unload
    ...
```

Reverse of load action shown above.

```
    ...
  )
  (:durative-action move
    :parameters (?x - VEHICLE
      ?y ?z - LOCATION)
    :duration (= ?duration
      (travel-time ?x ?y ?z))
    :condition
      (and
        (at start (at ?x ?y))
        (at start (>=
          (travel-time ?x ?y ?z) 0))
        (at start (not (= ?y ?z)))
        (over all (available ?x))
      )
    :effect
      (and
        (at start (not (at ?x ?y)))
        (at end (at ?x ?z))
        (at end (increase (total-cost)
          (* ?duration (cost ?x))))
      )
  )
)
```

PDDL Problem Instance

```
(define (problem p01)
  (:domain multi-modal-cargo-routing)
  (:objects
    R1 R2 - VEHICLE
    L1 L2 L3 L4 L5 L6 L7 - LOCATION
    C1 C2 - CARGO
  )
  (:init
    ; Vehicle and Cargo Initial locations
    (at R1 L1)
    (at R2 L6)
    (at C1 L1)
    (at C2 L1)

    ; Vehicles ready to load cargoes
    (ready-loading R1)
    (ready-loading R2)

    ; Vehicle load/unload time
    (= (load-time R1 L1) 1)
    (= (unload-time R1 L1) 1)
    ...
```

Vehicle load/unload times as defined in Table 1.

```
    ...

    ; Vehicle availability
    (available R1)
    (available R2)

    ; Vehicle utilisation cost
    (= (cost R1) 10)
    (= (cost R2) 100)

    ; Capacity Constraints
    (= (remaining-capacity R1) 2)
    ...
```

Location capacity as defined in Figure 1.

```
    ...

    ; R1 reachability matrix
    (= (travel-time R1 L1 L2) 1)
    ...
```

Vehicle reachability as defined in Table 3.

```
    ...

    ; Size of cargo
    (= (size C1) 1)
    (= (size C2) 1)

    ; Cargo Time Window
    (not (available C1))
    (not (available C2))
    (at 1 (available C1))
    (at 1 (available C2))
    (at 11 (not (available C1)))
    (at 11 (not (available C2)))
  )
  (:goal (and (at C1 L5) (at C2 L5)))
  (:metric minimize (total-cost))
)
```

Example Solution

The plan for the example problem (the PDDL for which is shown above) was found by COLIN (Coles et al. 2012) in just over 45 seconds, and is shown in Figure 2. It should be noted that due to decimal separation in the action timings, time window deadlines for both items of cargo needed to be increased to 12 hours. The plan found follows the solution discussed previously, where l_3 is used as a cargo exchange point. The cost of the plan is correctly reported to be 1050.

```
0.003: (move v2 l6 l4)  [2.000]
1.001: (load v1 c2 l1)  [1.000]
2.002: (load v1 c1 l1)  [1.000]
3.003: (move v2 l4 l3)  [2.000]
3.004: (move v1 l1 l3)  [1.000]
4.005: (unload v1 c2 l3)  [1.000]
5.006: (unload v1 c1 l3)  [1.000]
5.007: (load v2 c2 l3)  [1.000]
6.008: (load v2 c1 l3)  [1.000]
7.009: (move v2 l3 l5)  [2.000]
9.010: (unload v2 c2 l5)  [1.000]
10.011: (unload v2 c1 l5)  [1.000]
```

Figure 2: Extract of plan computed using COLIN.

Domain Features

The MMCR domain provides a number of features that are of interest to the current state-of-the-art planners and the wider planning community.

Firstly, the realistic nature of this domain can produce problems with quite large state and action spaces. Problem instances we have tried comprise between 4-to-30 locations, 4-to-25 vehicles, and 61-to-6000 units of cargo to be transported. Most of our instances have to date proved to be unsolvable using current state-of-the-art planners COLIN (Coles et al. 2012), OPTIC (Benton, Coles, and Coles 2012), and CPT-YAHSP[4] (Vidal and Geffner 2004; Vidal and others 2004). Simpler problems, such as the exemplar described early in this paper, still take a seemingly long time to solve. The scalability of existing systems can be examined as we scale the example problem by increasing the number of cargo elements to be moved. If we reduce the exemplar to have a single cargo the problem can be solved in 0.01 seconds. Adding an additional cargo creates a problem that appears to be unsolvable within a reasonable time limit. This new item of cargo originates at l_2, with the demand location equal to that of other cargoes, and other constraints adjusted appropriately. Our preliminary testing seems to indicate that planning system scalability is difficult in our setting. We postulate the decrease in performance is related to the timed initial literals modelling delivery time windows. These constraints cause many dead end states within the search space, and require a state-based planner to continually back track.

Secondly, the domain presents a temporally interesting and complex problem set. Typical instances we encounter require multiple modes of transportation to be used. Just like the classical LOGISTICS benchmark, for a given delivery problem the paths between origin and destination are frequently disjoint, and therefore require the use of multiple vehicles to effect cargo delivery. The interactions of durative actions, action-coordination, and capacity and timing constraints provide a complex mix of new features for a planner to consider in that setting. Good planning systems must carefully consider pruning of the search and action spaces to enable timely problem satisfaction without affecting solution quality.

Finally, the problem is specified to enable the modelling of real-world logistic scenarios. Improving the state-of-the-art in automated planning to provide good quality solutions to MMCR problems should provide advances to other problems which exhibit cost-sensitivity, time and space constrainedness, and limitations of transport resources. Our benchmarking also further seeks to reduce the gap between automated planning research and practical applications.

[4]Simplified problem instances were trialled with CPT-YAHSP due to lack of support for timed initial literals.

Planned Extensions

We plan to extend the current domain to include more aspects from the underlying real-world problems. We also plan to consider giving more flexibility to planners by softening delivery deadlines. This section describes the two proposed domain extensions in more detail.

Real World Behaviour

In the current domain model total plan cost is a factor of vehicle utilisation alone. While this is the primary contributor to costs in logistics applications (Rushton, Croucher, and Baker 2006; Ghiani, Laporte, and Musmanno 2004), warehousing costs are also a significant factor. The model should therefore be extended to incorporate the time which cargo is stored in warehousing. This would effect solutions that require the use of intermediate cargo exchange points. For short term scenarios such costs should incentivise synchronization of vehicles involved in exchanges at warehouses in order to minimise the time cargo waits at these points.

We also aim to investigate extending the model to include production and consumption actions at origin and destination locations. Currently cargo is located at the origin until it is available to be moved, and (once successfully moved) remains at the destination until the problem is completed. While this behaviour replicates the nature of some types of cargo (e.g. vehicles), other types (such as food, etc.) follow more of a production/consumption life-cycle. It is our intent to investigate the possibility of including these actions once we have completed our initial study of the MMCR benchmark.

Relaxed Delivery Deadlines

Currently, an admissible solution must deliver all cargoes within their respective time window deadlines. We plan to investigate relaxing these constraints and employing the use of preferences. This will give the planner more flexibility to make cost driven decisions. The total-cost metric can then be utilised to balance the cost of transportation vs cargo delay.

Conclusion

This paper presented a multi-modal cargo routing problem as a new planning domain benchmark. Preliminary tests of problem instances on domain independent planners such as COLIN and CPT-YAHSP show that small problems push the limits of these technologies, while real-world sized instances are beyond current capabilities.

We have discussed the interesting and complex features of this domain. Future work will focus on better understanding how this complexity affects the planning performance of the current state-of-the art, and building upon these technologies to provide high quality, efficient solutions to this benchmark. We are especially interested in how sampling could be used to balance solution quality and computational complexity.

References

Allard, T., and Shekh, S. 2012. Hierarchical multi-agent distribution planning. In Thielscher, M., and Zhang, D., eds., *AI 2012: Advances in Artificial Intelligence*, number 7691 in Lecture Notes in Computer Science. Springer Berlin Heidelberg. 755–766.

Benton, J.; Coles, A.; and Coles, A. 2012. Temporal planning with preferences and time-dependent continuous costs. In *Twenty-Second International Conference on Automated Planning and Scheduling*.

Coles, A.; Fox, M.; Long, D.; and Smith, A. 2008. Planning with problems requiring temporal coordination. In *AAAI*, AAAI'08, 892–897. Chicago, Illinois: AAAI Press.

Coles, A. I.; Fox, M.; Halsey, K.; Long, D.; and Smith, A. J. 2009. Managing concurrency in temporal planning using planner-scheduler interaction. *Artificial Intelligence* 173(1):1–44.

Coles, A.; Coles, A.; Fox, M.; and Long, D. 2012. COLIN: Planning with continuous linear numeric change. *Journal of Artificial Intelligence Research (JAIR)* 44:1–96.

Crainic, T. G., and Laporte, G. 1997. Planning models for freight transportation. *European Journal of Operational Research* 97(3):409 – 438.

Cushing, W.; Kambhampati, S.; Weld, D. S.; et al. 2007. When is temporal planning really temporal? In *Proceedings of the 20th international joint conference on Artifical intelligence*, 1852–1859. Morgan Kaufmann Publishers Inc.

Dimopoulos, Y.; Gerevini, A.; Haslum, P.; and Saetti, A. 2006. The benchmark domains of the deterministic part of ipc-5. *Abstract Booklet of the competing planners of ICAPS-06* 14–19.

Edelkamp, S., and Hoffmann, J. 2004. Pddl2. 2: The language for the classical part of the 4th international planning competition. Technical report.

Fikes, R. E., and Nilsson, N. J. 1971. STRIPS: A new approach to the application of theorem proving to problem solving. In *IJCAI*, IJCAI'71, 608–620. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Fox, M., and Long, D. 2003. PDDL 2.1: An extension to PDDL for expressing temporal planning domains. *Journal of Artificial Intelligence Research (JAIR)* 20:61–124.

García, J.; Florez, J. E.; Torralba, Á.; Borrajo, D.; López, C. L.; García-Olaya, Á.; and Sáenz, J. 2013. Combining linear programming and automated planning to solve intermodal transportation problems. *European Journal of Operational Research* 227(1):216–226.

Gerevini, A., and Long, D. 2006. Preferences and soft constraints in pddl3. In *ICAPS workshop on planning with preferences and soft constraints*, 46–53.

Ghiani, G.; Laporte, G.; and Musmanno, R. 2004. *Introduction to Logistics Systems Planning and Control*. John Wiley & Sons.

Goel, A., and Gruhn, V. 2008. A general vehicle routing problem. *European Journal Of Operational Research* 191(3):650–660.

Golden, B. L.; Raghavan, S.; and Wasil, E. A., eds. 2010. *The Vehicle Routing Problem: Latest Advances and New Challenges*. Springer.

Kilby, P., and Verden, A. 2011. Flexible routing combing constraint programming, large neighbourhood search, and feature-based insertion. In *Artificial Intelligence and Logistics (AILog) Workshop at IJCAI'11*, 43–48.

Kovacs, D. L. 2011. BNF definition of PDDL 3.1. Technical report.

Marzal, E.; Sebastia, L.; and Onaindia, E. 2014. On the use of temporal landmarks for planning with deadlines. In *ICAPS*.

McDermott, D. 2000. The 1998 ai planning systems competition. *AI MAGAZINE* 21(2).

Penberthy, J. S., and Weld, D. S. 1994. Temporal planning with continuous change. In *AAAI*, AAAI'94, 1010–1015. Menlo Park, CA, USA: American Association for Artificial Intelligence.

Ramesh, T. 1981. Traveling purchaser problem. *Opsearch* 18:78–91.

Rankooh, M. F., and Ghassem-Sani, G. 2013. New encoding methods for sat-based temporal planning. In *ICAPS*.

Ropke, S., and Pisinger, D. 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science* 40(4):455–472.

Rushton, A.; Croucher, P.; and Baker, P. 2006. *The Handbook of Logistics and Distribution Management*. Kogan Page Publishers, 3rd edition.

Smith, D. E., and Weld, D. S. 1999. Temporal planning with mutual exclusion reasoning. In *IJCAI*, volume 99, 326–337.

Vidal, V., and Geffner, H. 2004. Branching and pruning: An optimal temporal POCL planner based on constraint programming. In *AAAI*, 570–577.

Vidal, V., and others. 2004. A lookahead strategy for heuristic search planning. In *ICAPS*, 150–160.

Younes, H. L., and Simmons, R. G. 2003. VHPOP: Versatile heuristic partial order planner. *Journal of Artificial Intelligence Research (JAIR)* 20:405–430.