O-Plan2 Technical Report

The Pacifica Simulator

Glen A. Reece

Department of Artificial Intelligence
University of Edinburgh

Artificial Intelligence Applications Institute
University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
United Kingdom

March 14, 1994

ARPA-RL/O-Plan2/TR/14 Version 2

Contents

1 Motivation

The Pacifica Simulator is a testbed for studying transportation planning/scheduling, logistics, and Non-combatant Evacuation Operation (NEO) scenarios. It draws upon data from the PRECiS Environment [Reece *et al.* 1993] to define a fictional domain that is a suitably realistic for studying such scenarios. This domain is based on a hypothetical theater of operations called the island of Pacifica.

The simulator was designed to show how an execution agent is able to handle simultaneous tasks which take place over extended time intervals and which may possibly interact. The Pacifica domain models a command and control environment where the execution agent takes the role of a field commander receiving mission directives from a superior agent and issues orders to entities under its direct control to carry out those missions. These missions are in the context of NEOs. Such operations are undertaken to provide rapid response to a variety of circumstances such as storms and other natural disasters, evacuation of civilians from trouble zones, policing and medical missions, humanitarian relief, etc. They are often characterized by the need for rapid deployment of equipment and personnel to ensure effective aid is offered and to seek to minimize the escalation of the problems through delays. The transportation logistics associated with NEOs present many interesting problems for reactive execution.

The development of the Pacifica Simulator is motivated by research in the area of Reactive Execution Agents (REAs) [Reece 1992]. In this research a REA is tasked to carry out a NEO in Pacifica. The REA accepts a plan for performing the NEO from a planning agent, synthesizes a task directive which it can execute, and then begins to issue task requests (i.e., send messages) to the simulator.

The primary reason for the use of the Pacifica Simulator is to demonstrate the early failure detection, resource reasoning, reflexive knowledge, communication, change-of-focus, asynchronous input, and failure management capabilities of the REA design.

2 The Pacifica Simulator

The Pacifica Simulator is a message–based discrete event simulator which features exogenous change, complex interaction among objects in the environment, uncertainty, sensing, and continuous time.

The simulator consists of:

- Three primary windows (i.e., World History, World Agenda, and World Interaction) and one window for displaying the current simulated time (i.e., World Clock).
- Common Lisp Object System (CLOS) object definitions which define the entities of the world and their characteristics.
- Four exogenous event types (i.e., Meteorological, Resource, Terrorist-activity, and Natural-disaster) that occur probabilistically.

The World History window displays a summary of what has happened in Pacifica since the simulation began. The World Agenda shows the events which are scheduled to happen and have yet to occur, and the World Interaction window allows the simulation user to cause events to fail, be delayed, or occur sooner than scheduled (see Figure 1 for a sample run of Pacifica). The simulator is also capable of producing a snapshot file on command which details modeled characteristics for display with a graphical viewer (e.g., O-Plan2's World Viewer).

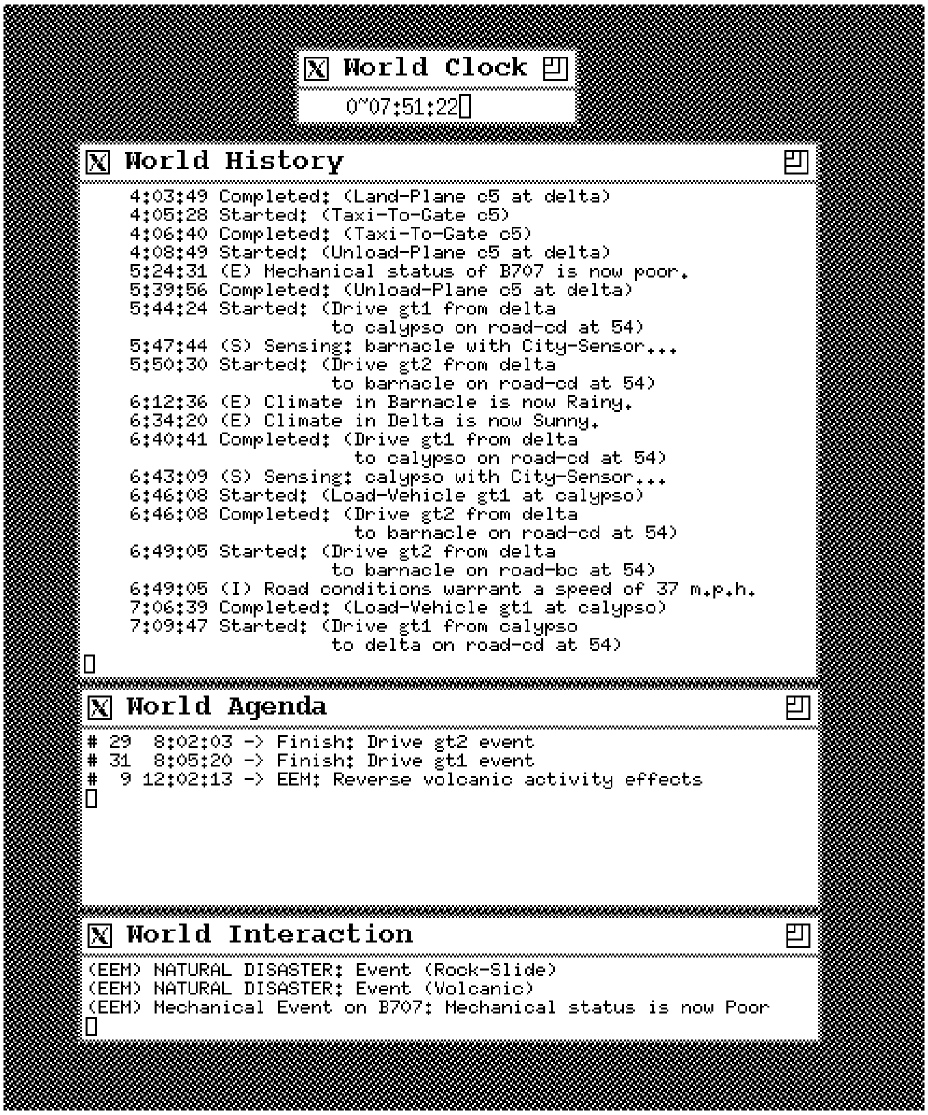

Figure 1: Pacifica Simulator

Events occur either as a result of receiving a message to initiate a particular action in the environment or by the Exogenous Event Manager (EEM). The EEM generates random events based upon user–defined probability distributions. At present, four types of exogenous events can occur during the simulation. These are meteorological events, resource events, terrorist activity, and natural disasters.

Meteorological events change the weather at randomly selected cities in Pacifica and subsequently effect the road conditions between cities. Road conditions along with road types determine how fast a ground-transport may travel on a particular road. The effects of meteorological events are to change the climate at a particular location to one of "sunny," "rainy," or "stormy."

Resource events effect characteristics of transportation resources such as fuel levels, mechanical status, and condition of tires. If the fuel level of a particular resource is below a user–defined threshold when the resource is to be used then this results in a failure. The mechanical status of a resource must be "good" and the condition of the tires must be "good" before a resource can be used. Otherwise, an appropriate failure is signaled to the REA. The effects of resource events can be no change to the resource, reduction of the resource's fuel level by 10%, reduction of fuel to zero, a change in the resource's mechanical status to one of "good," "poor," or "bad," or a change in the resource's tire status to "okay" or "blown."

Terrorist activity events effect ground transport resources and bridges. At present, terrorist events only occur on ground transports if the ground transport is found to be traveling down a road where a terrorist group is currently located. If attacked by terrorists, ground transport resources can have their mechanical status reduced to "poor" or "bad", tires shot out, or even be captured. However, they may also be lucky enough to escape unharmed.

The forth exogenous event type is Natural disaster. These events take the form of rock slides, volcanic activity, and floods. Rock slides slow down ground transport resources by 10% on the affected road. Volcanic activity closes roads and can potentially destroy ground transports if the ground transport is in a specific five mile stretch of road. Floods simply close roads. The effects of rock slides only affect a ground transport if on the road at the time and does not affect subsequent travel along the road. Volcanic and flood events can affect roads for long periods of time. The effects of volcanic activity lasts for 10 hours unless another volcanic event occurs during that 10 hour period. If another volcanic event occurs before the 10 hour period has expired then an additional 3 hours is added to the time before the road re–opens. Floods last for 5 hours unless another occurs in which case 2 more hours are added before the road is open for travel again.

Twenty-one message types are presently accepted by the Pacifica Simulator. These range from simple sensor requests to complex drive requests which can have anyone of seven pre–conditions not satisfied which result in failures.

2.1 Modeled Characteristics in Pacifica

The Pacifica Simulator currently models entities related to the Small-Scale NEO defined in the PRECiS Environment [Reece *et al.* 1993]. That is, it models ground-transports, air-cargo-transports, air-passenger-transports, cities, bridges, weather, terrorists, and natural disasters. The individual characteristics modeled depend on the type of entity. For instance, ground-transport resource objects have thirteen dynamic and ten static characteristics, while bridge objects only have two (one dynamic and one static) characteristics which are modeled in the Pacifica Simulator[1].

[1]See Appendix A for the CLOS definitions of currently modeled entities and their characteristics.

2.2 Simulator Interaction

Requests are issued by the REA in the general form:

(send-to-world <task-name> <arguments>)

The Pacifica Simulator checks any pre–conditions which may exist for the task which is being requested. If all of the pre–conditions are satisfied then the task is scheduled for a duration appropriate to the particular task. When the duration has elapsed the effects of the task are sent to the REA. If the pre–conditions of a task are not met, a failure is sent back to the REA with a reason as to why the failure occurred.

The REA models the types of effects which a task can be expected to yield upon successful completion. It receives two types of messages from the simulator—these being effects and sensor messages. The REA places all effects and sensor information into the its model of the environment so that it can base decisions upon the current situation in the environment. However, the beliefs which the REA holds about particular characteristics of entities in the environment may or may not correspond to their actual values. This is due to the fact that the EEM in the Pacifica Simulator causes changes in the world which the REA will not immediately be aware of unless it is actively monitoring for those particular changes. This provides a rich environment for studying issues related to reactive execution of plans. If a failure occurs then the REA is notified as to the reason why it occurred so that it can respond in an appropriate manner.

References

[Reece *et al.* 1993] G. Reece, A. Tate, D. Brown, and M. Hoffman. The PRECiS Environment. In *Proceedings of AAAI-93: DARPA-RL Planning Inititive Workshop*, Washington, D.C., 1993. Available as ARPA-RL/O-Plan2/TR/11 from Artifical Intelligence Applications Institue.

[Reece 1992] G. A. Reece. Reactive Execution in a Command, Planning, and Control Environment. Technical Report 121, Department of Artificial Intelligence, University of Edinburgh, Scotland, 1992.

A Pacifica CLOS Object Classes

The following classes define the entities of the Pacifica Simulator and thus, the characteristics which are presently modeled by the simulator. The slots which contain information which dynamically alters during the simulation are listed first in the class followed by the static information. This is indicated by a blank line.

```
(defclass world ()
  ((name :initarg :name :accessor name))
  (:documentation "Generic definition of a world object."))

(defclass city (world)
  ((climate :initarg :climate :accessor climate)
   (num-nationals :initarg :num-nationals :initform 0 :accessor num-nationals)
   (status :initarg :status :initform 'open :accessor status)

   (airport :initarg :airport :initform nil :accessor airport)
   (air-routes :initarg :air-routes :initform nil :accessor air-routes)
   (neighbors :initarg :neighbors :initform nil :accessor neighbors)
   (roads :initarg :roads :initform nil :accessor roads)
   (seaport :initarg :seaport :initform nil :accessor seaport))
  (:documentation "Generic definition of a city object."))

(defclass road (world)
  ((condition :initarg :condition :accessor condition)
   (status :initarg :status :accessor status)

   (bridge-xing :initarg :bridge-xing :initform nil :accessor bridge-xing)
   (distance :initarg :distance :accessor distance)
   (end-points :initarg :end-points :initform nil :accessor end-points)
   (road-type :initarg :road-type :accessor road-type))
  (:documentation "Generic definition of a road object."))

(defclass air-route (world)
  ((status :initarg :status :accessor status)
   (distance :initarg :distance :accessor distance)
   (end-points :initarg :end-points :initform nil :accessor end-points))
  (:documentation "Generic definition of a road object."))

(defclass bridge (world)
  ((status :initarg :status :accessor status)
   (roads :initarg :roads :accessor roads))
  (:documentation "Generic definition of a bridge object."))
```

```
(defclass airport (world)
  ((status :initarg :status :accessor status)
   (loc :initarg :loc :accessor loc))
  (:documentation "Generic definition of a airport object."))

(defclass seaport (world)
  ((status :initarg :status :accessor status))
  (:documentation "Generic definition of a seaport object."))

(defclass terrorist-group (world)
  ((loc :initarg :loc :accessor loc))
  (:documentation "Generic definition of a terrorist-group object."))

(defclass ground-transport (world)
  ((closest-city :initform nil :accessor closest-city)
   (failure-reason :initarg :failure-reason :initform nil
                   :accessor failure-reason)
   (fuel :initarg :fuel :initform 55 :accessor fuel)
   (in-country :initarg :in-country :initform nil :accessor in-country)
   (last-loc :initform nil :accessor last-loc)
   (loc :initarg :loc :initform nil :accessor loc)
   (mech-status :initarg :mech-status :initform 'good :accessor mech-status)
   (passengers :initarg :passengers :initform 0 :accessor passengers)
   (speed :initarg :speed :initform 0 :accessor speed)
   (status :initarg :status :accessor status)
   (status-load :initarg :status-load :initform 'empty
                :accessor status-load)
   (tire-status :initarg :tire-status :initform 'okay :accessor tire-status)
   (using-road :initarg :using-road :initform nil :accessor using-road)
   (world-event-struct :initarg :world-event-struct
                       :initform nil :accessor world-event-struct)

   (avg-speed :initarg :avg-speed :initform 40 :accessor avg-speed)
   (fuel-capacity :initarg :fuel-capacity :initform 55 :accessor fuel-capacity)
   (fuel-rate :initarg :fuel-rate :initform 6.3 :accessor fuel-rate)
   (offload :initarg :offload :initform 1200 :accessor offload)
   (onload :initarg :onload :initform 1200 :accessor onload)
   (range :initarg :range :initform 348 :accessor range)
   (refuel :initarg :refuel :initform 900 :accessor refuel)
   (tires :initarg :tires :initform 'standard :accessor tires)
   (w-cargo :initarg :w-cargo :initform 0 :accessor w-cargo)
   (wo-cargo :initarg :wo-cargo :initform 50 :accessor wo-cargo))
  (:documentation
   "Ground Transport specific characteristics.
    static and dynamic slots (note that all times are given
```

```
    in seconds)."))

(defclass air-cargo-transport (world)
  ((at-gate :initarg :at-gate :initform 'no :accessor at-gate)
   (failure-reason :initarg :failure-reason :initform nil
                 :accessor failure-reason)
   (fuel :initarg :fuel :initform 2000 :accessor fuel)
   (in-country :initarg :in-country :initform nil :accessor in-country)
   (last-loc :initform nil :accessor last-loc)
   (loc :initarg :loc :initform nil :accessor loc)
   (mech-status :initarg :mech-status :initform 'good :accessor mech-status)

   (world-event-struct :initarg :world-event-struct
                     :initform nil :accessor world-event-struct)
   (passengers :initarg :passengers :initform 0 :accessor passengers)
   (status :initarg :status :initform 'available :accessor status)
   (status-load :initarg :status-load :initform 'empty
              :accessor status-load)
   (tire-status :initarg :tire-status :initform 'okay :accessor tire-status)
   (tower-clearance :initarg :tower-clearance :initform 'no
                  :accessor tower-clearance)
   (using-route :initarg :using-route :initform nil :accessor using-route)

   (avg-speed :initarg :avg-speed :initform 436 :accessor avg-speed)
   (fuel-capacity :initarg :fuel-capacity
                :initform 2000 :accessor fuel-capacity)
   (fuel-rate :initarg :fuel-rate :initform 3.1 :accessor fuel-rate)
   (offload :initarg :offload :initform 5400 :accessor offload)
   (onload :initarg :onload :initform 6300 :accessor onload)
   (range :initarg :range :initform 6238 :accessor range)
   (refuel :initarg :refuel :initform 4500 :accessor refuel)
   (runway :initarg :runway :initform 9150 :accessor runway)
   (w-cargo :initarg :w-cargo :initform 30 :accessor w-cargo)
   (wo-cargo :initarg :wo-cargo :initform 329 :accessor wo-cargo))
  (:documentation
   "Air Cargo Transport specific characteristics.
    static and dynamic slots (note that all times are given
    in seconds)."))
```

```
(defclass air-passenger-transport (world)
  ((at-gate :initarg :at-gate :initform 'no :accessor at-gate)
   (failure-reason :initarg :failure-reason :initform nil
                   :accessor failure-reason)
   (fuel :initarg :fuel :initform 2500 :accessor fuel)
   (in-country :initarg :in-country :initform nil :accessor in-country)
   (last-loc :initform nil :accessor last-loc)
   (loc :initarg :loc :initform nil :accessor loc)
   (mech-status :initarg :mech-status :initform 'good :accessor mech-status)
   (world-event-struct :initarg :world-event-struct
                       :initform nil :accessor world-event-struct)
   (passengers :initarg :passengers :initform 0 :accessor passengers)
   (status :initarg :status :initform 'available :accessor status)
   (status-load :initarg :status-load :initform 'empty
                :accessor status-load)
   (tire-status :initarg :tire-status :initform 'okay :accessor tire-status)
   (tower-clearance :initarg :tower-clearance :initform 'no
                    :accessor tower-clearance)
   (using-route :initarg :using-route :initform nil :accessor using-route)

   (avg-speed :initarg :avg-speed :initform 435 :accessor avg-speed)
   (fuel-capacity :initarg :fuel-capacity :initform 2500
                  :accessor fuel-capacity)
   (fuel-rate :initarg :fuel-rate :initform 3.0 :accessor fuel-rate)
   (offload :initarg :offload :initform 2700 :accessor offload)
   (onload :initarg :onload :initform 1800 :accessor onload)
   (range :initarg :range :initform 8427 :accessor range)
   (refuel :initarg :refuel :initform 3600 :accessor refuel)
   (runway :initarg :runway :initform 6831 :accessor runway)
   (w-cargo :initarg :w-cargo :initform 10 :accessor w-cargo)
   (wo-cargo :initarg :wo-cargo :initform 311 :accessor wo-cargo))
  (:documentation
   "Air Passenger Transport specific characteristics.
    static and dynamic slots (note that all times are given
    in seconds)."))
```