Journal of
Computer Science & Systems Biology

Redjaimia, J Comput Sci Syst Biol 2015, 8:6
DOI: 10.4172/jcsb.1000208

Review Article Open Access

Multi-Agent System and Events Plan Construction Using PDDL

Abdellah Redjaimia *
Lobachevsky State University of Nizhni Novgorod, Nizhny Novgorod, Russia

Abstract

Life is the main interesting thing in this existence and we are always searching to make it better and easy answering our duties as simple as it can, the purpose is easy to understand it cause we are looking for a comfortable life which handles all our routine tasks, for that today we introduce a system which can be plugged on our terminals and which it can make tasks automatically basing on the environment events. We present a dissertation project details and its application on both sides the web corner and terminal assistant, Multi-Agent System collaborates with PDDL to determinate which functions and procedures the agent has to execute at the time basing on environments events which are playing the rules of parameters in each functions or procedures.

Keywords: Multi-agent; Agent; PDDL; Engine; Domain; Predicates; Robot

Introduction

Multi-Agent System or Smart Agent Group is a system which divides rules between agents for fast execution and treatment of more tasks in one time (Parallelism) [1]. An Agent is a smart process which gets all its existence from human being's thoughts and it can react thousand times faster than him [2]. In our study, we are focusing on Multi-Agent System as an engine which creates a specific agent for a specific mission basing on parameters (Environment Events) sent from user (via voice or keyboard) or taken via sensors (Case Robot).

PDDL

The Planning Domain Definition Language (PDDL) is an attempt to standardize by Drew McDermott and his colleagues in Artificial Intelligence (AI) 1998 (inspired by STRIPS and ADL among others) mainly to make the planning languages. It was first developed 1998/2000 International Planning Competition (IPC) possible and then evolved with each competition. The adoption of a common formalism for describing planning domains fosters far greater reuse of research and allows more direct comparison of systems and approaches, and therefore supports faster progress in the field. A common formalism is a compromise between expressive power (in which development is strongly driven by potential applications) and the progress of basic research (which encourages development from well-understood foundations) [3]. The role of a common formalism as a communication medium for exchange demands that it is provided with a clear semantics. PDDL is a specific language to create plans for each agent basing on predicates that we are interested in. In our study we are going to change a little in PDDL structure matching our goals. We are focusing on PDDL Domain, PDDL predicates (Environment Events) and actions which match every predicate's state (True or False).

NB

The number of predicates determines automatically the number of agents which have to participate. For example, if we take our Domain: Car and predicates for example Rain and Snow then the agent's engine will create 02 agents equal to the number of predicates which is 02 (Rain and Snow). The Action Clause studies every predicate in 020020states.

Case TRUE: then the agent has some instructions to execute

Case FALSE: then the agent has some instructions to execute.

The max number of cases which the system can handle is equal to $2^{(\text{number of predicates})}$.

Below an example of PDDL which has got those proprieties:

1- Domain: Car,

2- Predicates got from the environment,

3- Init objects inherent from Car's features,

4- Actions basing on the plan definition,

According to each predicate received from the assistant which in our case a Car Robot.

PDDL-Car-Robot-Assistant Domain (define-domain Car-Robot-Assistant)

Predicates in our case represented by the car physical things such as windows, doors, lamps, post-radio, glasses, wheels, air conditioner, etc. In our PDDL we will just show an example, how to write our plan basing on some of those predicates which are: the car's windows, the car's air conditioner, the car's glass cleaner and the car's lamps [4].

Physical predicates

(Car-Windows?cw)

(Car-Air-Conditioner?cac)

(Car-Windows-Cleaner?cwc)

(Car-Windows-Lamps?cwl)

(Car-Doors: CD)

(Car-Speed: CSP)

***Corresponding author:** Abdellah Redjaimia, Lobachevsky State University of Nizhni Novgorod, Nizhny Novgorod, Russia, E-mail: redjaimia@gmail.com

Received October 06, 2015; **Accepted** October 27, 2015; **Published** November 02, 2015

Citation: Redjaimia A (2015) Multi-Agent System and Events Plan Construction Using PDDL. J Comput Sci Syst Biol 8:6 333-340. doi:10.4172/jcsb.1000208

Citation: Redjaimia A (2015) Multi-Agent System and Events Plan Construction Using PDDL. J Comput Sci Syst Biol 8:6 333-340. doi:10.4172/jcsb.1000208

Logical predicates, it means True or False situations

```
(is-closed-windows?bcw)
;; this predicate returns true if the car's windows are closed - false otherwise -b- means Boolean.
(is-switchedon-airC?bsa)
;; this predicate returns true if the car's air conditioner is on - false otherwise -b-means Boolean.
(is-lightedon-lamps?bll)
;; this predicate returns true if the car's
 lamps are on - false otherwise -b- means
 Boolean.
(is-closed-doors?bcd)
;; this predicate returns true if the car's
 doors are closed- false otherwise -b- means Boolean.
(is-speed-extreme?bcs)
;; this predicate returns true if the car's speed is extremely high- false otherwise -b- means Boolean.
)

;; Domain's Actions.

;; CloseWindows
(
: action CloseWindows
: parameters (?cw)
: precondition (and (car-windows ?cw))
: effect (and (is-closed-windows ?cw))
)

;; OpenWindows
(
: action OpenWindows
: parameters (?cw)
: precondition (and (car-windows ?cw))
: effect (and (not (is-closed-windows
?cw)))
)

;; Switch On Air Conditioner
(
: action Switch On Air Conditioner
: parameters (?cac ?cw ?cd)
: precondition (and (car-windows ?cw)
(car-air-conditioner ?cac) (car-doors
?cd))
: effect (and (is-switchedon-airC ?cac)
(is-closed-windows ?cw) (is-closed-
doors ?cd))
)

;; CloseDoors
(
: action CloseDoors
: parameters (?cd)
: precondition (and (car-doors ?cd))
: effect (and (is-closed-doors ?cd))
)

;; OpenDoors
(
: action OpenDoors
: parameters (?cd)
: precondition (and (car-doors ?cd))
: effect (and (not(is-closed-doors ?cd)))
)

;; SlowDown
(
: action SlowDown
: parameters (?csp)
: precondition (and (car-speed ?csp))
: effect (and (not(is-speed-extreme
?csp)))
)
```

NB

The Action Slow Down depends on the speed limit which it can be discussed with professional road experts cause simply if it is a public road the speed limit has to be: less than 50 Km/h and it depends also on each country, below we can site some countries with their speed limits range basing on the road taken [5] (Figure 1).

Basing on the map the assistant can create events which are matching with every country speed laws then the system and for sure has to be completed has some features in options view which allows every factory to initialize basic parameters depends on speed value.

Example of speed limit control

The speed limit in Algeria is 75 Km/h on all roads types, the assistant has to make control with this value in all cases then if the user speeds up 75 Km/h then the Assistant can inform him/her that the speed is extreme and for further features the assistant can allow for him/her to drive during a timer charged inside it or directly the assistant can slow down automatically and control the car instead the driver. Speed limit Value: 75 km/h.

Optional Features

Speech to text engine

This option frequently used in Terminals such as Computers, Credit Payment Terminals, etc. The option translates the voice to a text and nowadays is not a stable cause the engine can't recognize many voices especially in case of Dialects [6].

In our view point the engine needs some plug-in inside to make it more efficiency those plug-in helps to give the right text to the right speech, parsing one of the helpful plug-in which allows to Speech to Text engine knows the phrase parts then it will give approximately the right Text basing to the voice sent [1].

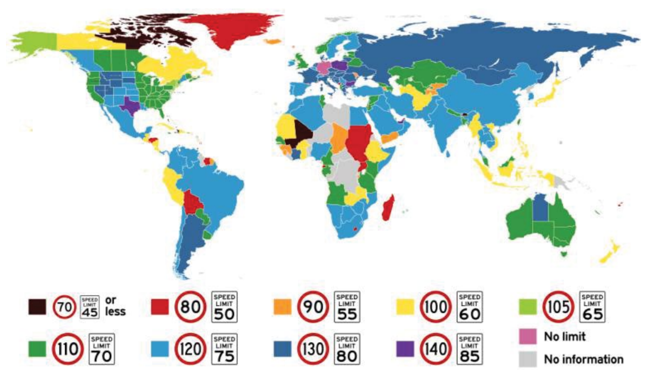

Figure 1: Speed limit values depending on countries.

The system security

System security one of the funny features in the system, some users would like launching the system at the same moment when the terminal starts. The logic says that and we are agree with the system going to be under function in parallel with its holder (Computer-Robot, etc.). But in this case also the system's owners wouldn't like seeing others using the system for their benefits, for example the system will not execute the instructions if the user is under 18 years old, in this case it is a useful feature if we root a security identifier such as: Finger Print- Eye Scan-Code Bar Scanner, etc. [7].

Implementation

We will take our previous example which is considering the nature as an environment, cars as the study domain and events or predicates which are wind and temperature.

Basing on our data we can recognize

- Number of Predicates: 2 (Wind+Temperature).

- The Number of Agents: 4 Agents.

- PDDL Possible Actions: 4 Actions (Table 1).

*__Act 1: Wind__=False
Temperature=False.

*__Act 2: Wind__=False
Temperature=True.

*__Act 3: Wind__=True
Temperature=False.

*__Act 4: Wind__=True
Temperature=True.

- Function creation basing on predicates:

It is a choice between creation of one function which handles all the 4 situations or to give every situation a function which handles it separately. In our case we prefer to assign one functions to all situations and this function gets parameters which are predicates and every agent uses the same function just it has to prepare the right parameters (Predicates) for the function created.

```
Function Action (Wind, Temperature)
{
if (Wind == True)
{
if (Temperature == True)
{
Close cars windows;
Switch on the Air conditioner;
}else
Close cars windows;
}else
{
if (Temperature == True)
Switch on the Air conditioner;
else
don't do anything;
}}
```

Wind	Temperature
0	0
0	1
1	0
1	1

Table 1: Basing on binary table.

The robot will execute all those instructions at the right moments basing on the events (Predicates, Parameters) which he has got from behavior which is the nature in our study all those things the robot will make them via it is sensors (Figure 2).

Car Sensors

In this section we will try to focus on some of car's sensors which are plugged automatically in it and we will see their functions their contacts and acts (their proprieties) [8]. As it is known our study is just to show the main functionality of the Multi-Agent System and nowadays there are plenty sensors which are represented inside the car engine such as: engine water temperature sensor, parking sensor, GPS sensor, speed sensor, etc. (Figure 3).

To cover all those sensors in our study is not our aim, our objective to choose some of those sensors to explain them basing on the user wishes, let's take some of those which are used on our PDDL file related to the Car Domain:

- Weather Temperature Sensor (in and out).

- Speed Sensor.

- Weather Wind Sensor.

- Parking Sensor.

A sensor is a transducer whose purpose is to sense (that is, to detect) some characteristic of its environments. It detects events or changes in quantities and provides a corresponding output, generally as an electrical or optical signal; for example, a thermo couple converts temperature to an output voltage [9]. But a mercury-in-glass thermometer is also a sensor; it converts the measured temperature into expansion and contraction of a liquid which can be read on a calibrated glass tube.

Sensors are used in everyday objects such as touch-sensitive elevator buttons (tactile sensor) and lamps which dim or brighten by touching the base, besides innumerable applications of which most people are never aware. With advances in micro machinery and easy-to-use micro controller platforms, the uses of sensors have expanded beyond the more traditional fields of temperature, pressure or flow measurement for example into MARG sensors [10].

Moreover, analog sensors such as potentiometers and force-sensing resistors are still widely used. Applications include manufacturing and machinery, airplanes and aerospace, cars, medicine and robotics.

Weather temperature sensor

Description

Temperature measurement range
-10°C to 50°C (14°F to 122°F)
Indoor:
-50°C to +70°C (-58°F to 158°F)
Outdoor:
can be used in car
Cable length:
1 m
Powered by 1 × 1.5V AG13 button cell

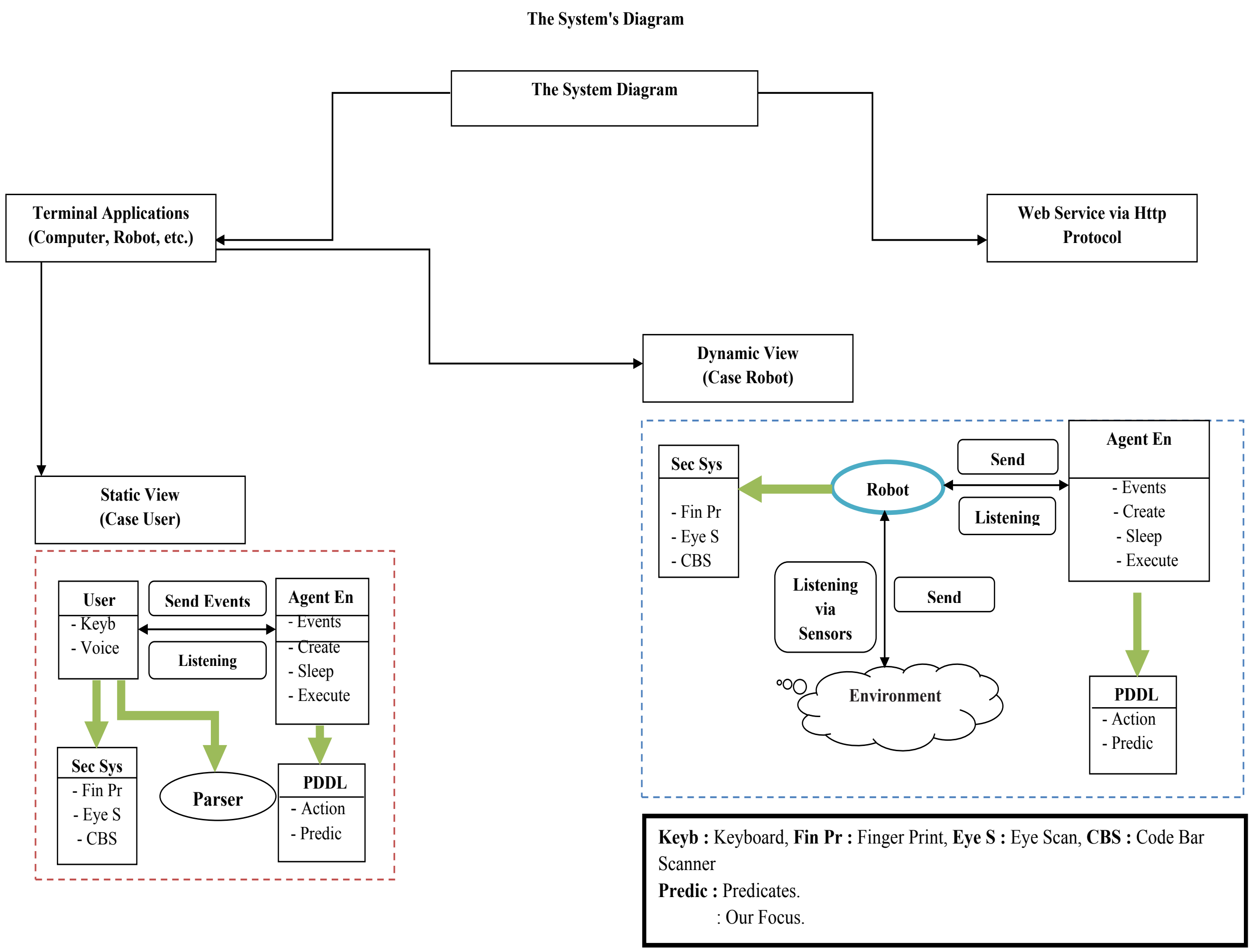

Figure 2: Multi-Agent system and events plan construction using PDDL.

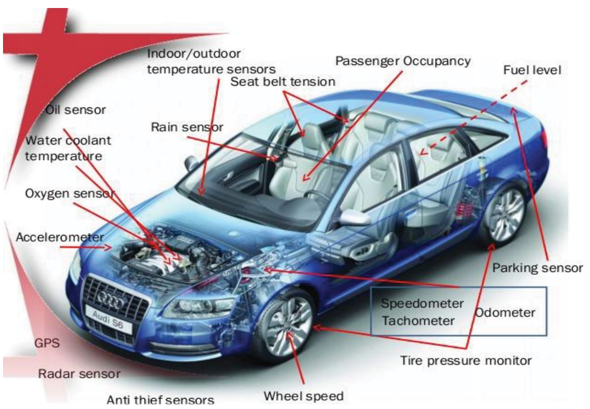

Figure 3: Car possible sensors.

2.1” LCD display screen
Size: 5.80 × 4.00 × 1.30 cm
Included :

1 × Digital Thermometer

1 × AG13 1.5 V Button Cells

The description shows us that the temperature range which the sensor can detect can offer us the whole control basing on our comfortable parameters in our sensor example the interval is fixed [-50°C+70°C], then the assistant can fix by itself a control value depends on the user wishes this value has to be between the Min and Max value which the sensor can detect.

Control value € [-50°C+70°C] (Figure 4).

Speed sensor

A wheel speed sensor or vehicle speed sensor (VSS) is a type of tachometer. It is a sender device used for reading the speed of a vehicle’s wheel rotation. It usually consists of a toothed ring and pickup. A speedometer or a speed meter is a gauge that measures and displays the instantaneous speed of a vehicle. Now universally fitted to motor vehicles, they started to be available as options in the 1900s, and as standard equipment from about 1910 onwards. Speedometers for other vehicles have specific names and use other means of sensing speed. For a boat, this is a pit log. For an aircraft, this is an airspeed indicator. Charles Babbage is credited with creating an early type of a speedometer, which were usually fitted to locomotives. The electric speedometer was invented by the Croatian Josip Belušić in 1888, and was originally called a velocimeter [8,9].

Citation: Redjaimia A (2015) Multi-Agent System and Events Plan Construction Using PDDL. J Comput Sci Syst Biol 8:6 333-340. doi:10.4172/jcsb.1000208

As we figured out that the Speed control value depends on the country laws and speed interval allowed please check the section example of Speed limit controls (Figures 5 and 6).

Weather wind sensor

Wind speed sensor wind speed, or wind velocity, is a fundamental atmospheric rate. Wind speed is caused by air moving from high pressure to low pressure. Many aspects can be affected by wind speed such as weather forecasting, aircraft, mines, navigation and agriculture, so the observation of wind speed is necessary. An anemometer or wind meter is a device used for measuring wind speed, and is a common weather station instrument. The term is derived from the Greek word anemos, meaning wind, and is used to describe any air speed measurement instrument used in meteorology or aerodynamics. The first known description of an anemometer was given by Leon Battista Alberti around 1450 [9].

Types of wind speed sensor nowadays the anemometers are usually divided into seven types

1. Cup Anemometers

Figure 4: Weather sensor with car plug-in.

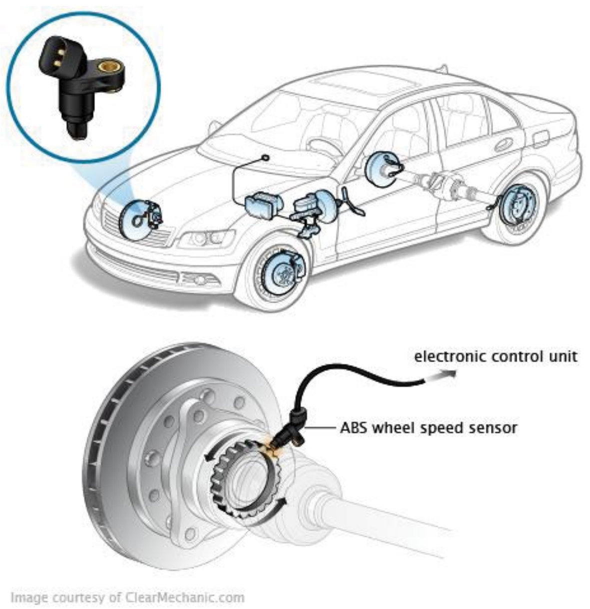

Figure 5: Car Speed Sensor.

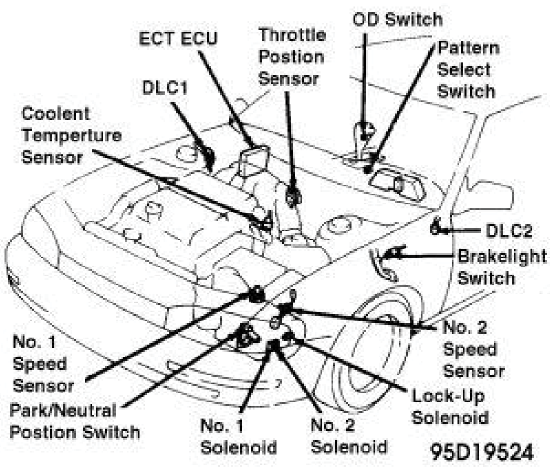

Figure 6: Car Speed Sensor Locations.

2. Vane Anemometers
3. Hot-wire Anemometers
4. Laser Anemometers
5. Sonic Anemometers
6. Acoustic Resonance Anemometers
7. Ping-pong Ball Anemometers

Wind speed sensor kinds are 7 as it is listed in our demonstration, we would to give a definition of one of them and our interesting is focusing on the top list one which is cup anemometers cause is the most widely used between all the rest of kinds.

The cup anemometer is the most widely used type which consists of several metal cups (usually three) attached to the ends of horizontal arms mounted on a vertical shaft. Wind catching in the cups causes them to revolve. This action turns the shaft, which is connected to a device that can calculate the wind speed (miles per hour, kilometers per hour, or some other ways to present speed). Commonly, the shaft is connected to an electrical generator and the amount of current produced by the generator varies with the speed of the wind. This kind of anemometer was further developed to measure both velocity and direction by adding a tag to one cup, which makes the cup wheel speed change along with the tag moving alternately with and against the mind. The fluctuation of cup wheel speed can be calculated to show the wind direction (Figure 7).

The Gill Anemometer is a traditional type, sensitive sensor for horizontal wind measurement. The Cup assembly is constructed with lightweight and UV resistant plastic cups. Housings are precision machined aluminum. A sensitive DC generator outputs wind speed. The anemometer mounting bracket installs on standard 3/4 inch threaded pipe.

Specifications

Wind Speed
0-60 m/s (130 mph)
Threshold
0.5 m/s (1.1 mph)

Wind Speed Signal

DC voltage linearly proportional to wind speed. 1800 RPM (2400 mV)=28.6 m/s (64.0 mph).

Dimensions

32 cm (12.5 in) H ×17 cm (6.7 in) dia.

Mounting

Standard 3/4 inch threaded pipe

Figure 7: Wind Sensor.

Weight

1.4 kg (3.1 lb)

Shipping Weight:

1.8 kg (4 lb)

Parking sensor

Parking sensors are proximity sensors for road vehicles designed to alert the driver to obstacles while parking. These systems, which use either electromagnetic or ultrasonic sensors, are marketed variously by vehicle manufacturers under proprietary brand names such as Park Distance Control, Park Assist, Parktronic or EPS. Parking Sensors detect obstacles behind you and also in front of your car if you wish, to aid your parking and enable you to get into those tight spots without damaging your car. Using Ultrasonic technology they emit a cone of sound behind your car which bounces of obstacles behind and returns to the sensor. The time taken for these sound waves to bounce and return off obstructions behind your car is how they calculate the remaining distance behind you (Figures 8-10).

NB

On both sides (Weather Wind Sensor and Parking Sensor) the assistant has just to inherent all the parameters from the plugged sensor and operate when something is going wrong with initial values for the second feature Parking Sensor the feature is prepared then the assistant will use just the information as parameters to execute it is tasks.

Figure 8: Car Parking Sensor.

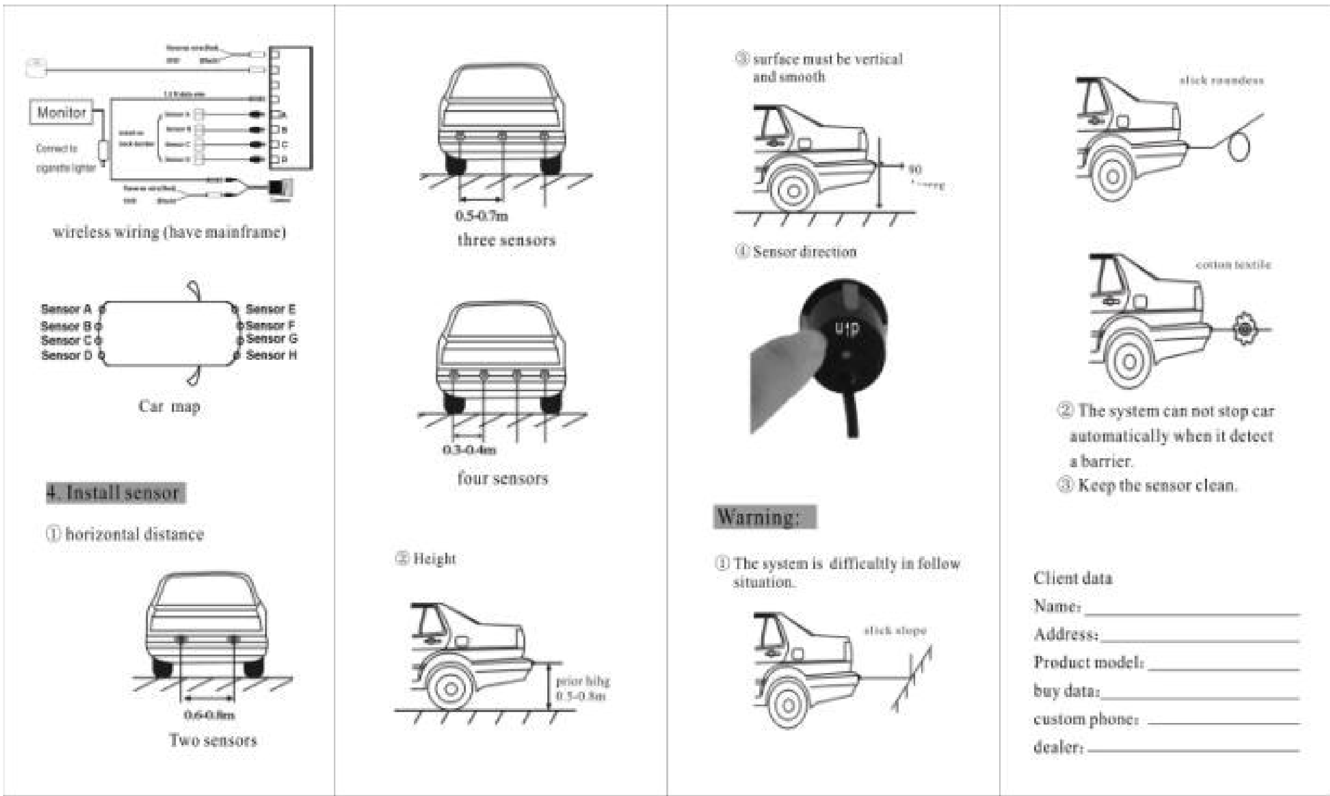

Figure 9: Parking Sensor Installation.

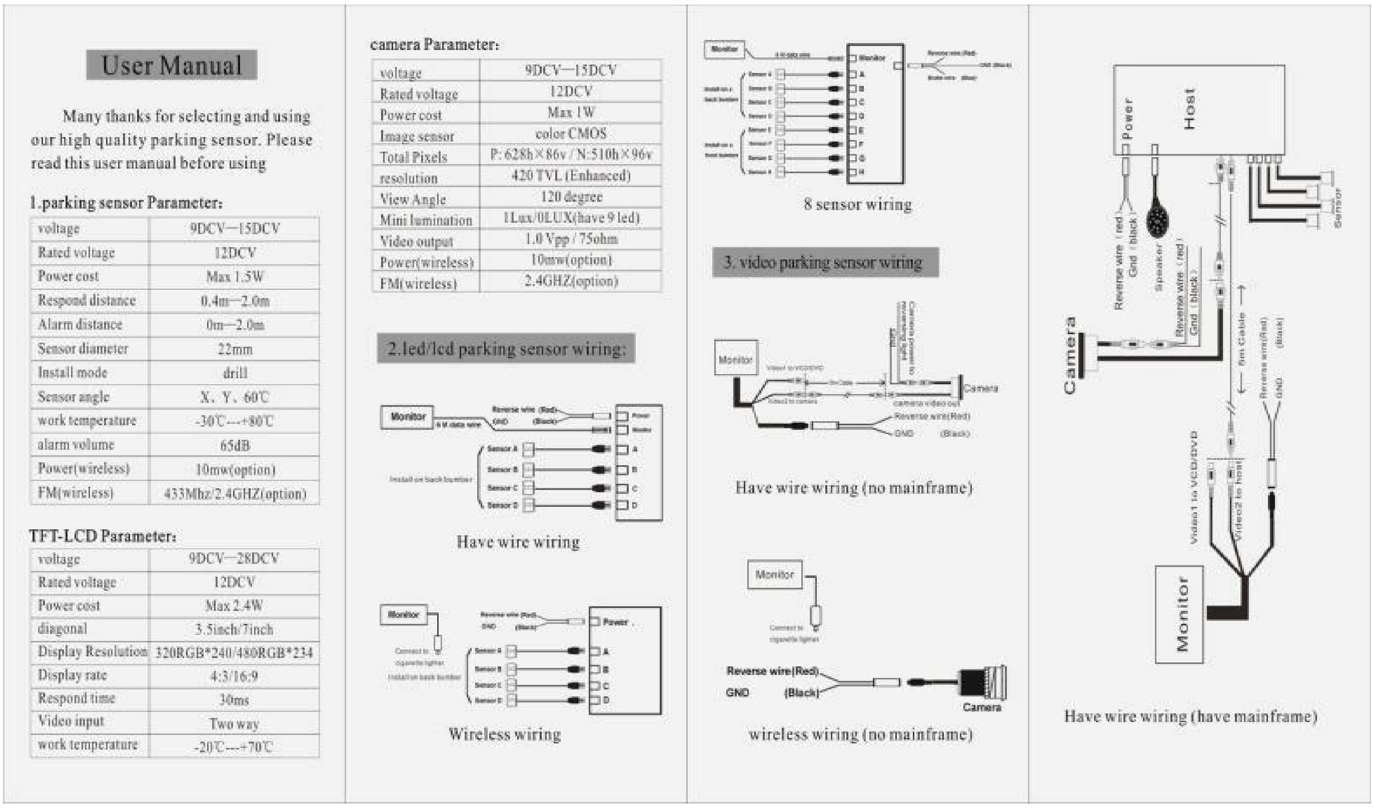

Figure 10: Parking Sensor User Manual.

Citation: Redjaimia A (2015) Multi-Agent System and Events Plan Construction Using PDDL. J Comput Sci Syst Biol 8:6 333-340. doi:10.4172/jcsb.1000208

References

1. Chaudhry B (2008) Improving the response to road death and injury Road Peace conference.
2. Worley H (2006) Road Traffic Accidents Increase Dramatically Worldwide. Population Reference Bureau.
3. Rabbi S (2007) Road Traffic Accident: A Leading Cause of the Global Burden of Public Health Injuries and Fatalities. Proceedings of the International Conference on Mechanical, Dhaka, Bangladesh.
4. Paefgen J, Michahelles F, Staake T (2011) GPS Trajectory Feature Extraction for Driver Risk Profiling. The ACM Conference on Ubiquitous Computing, Beijing, China.
5. O'Kane T, Ringwood JV (2013) Vehicle speed estimation using GPS/ RISS (Reduced Inertial Sensor System). Signals and Systems Conference, Letterkenny, Irish.
6. Eichner ML, Breckon TP (2008) Augmenting GPS Speed Limit Monitoring With Road Side Visual Information. Road Transport Information and Control - RTIC 2008 and ITS United Kingdom Members' Conference, Manchester, UK.
7. Puthon AS, Nashashibi F, Bradai B (2010) Improvement of multisensor fusion in speed limit determination by quantifying navigation reliability. 13th International IEEE Conference on Intelligent Transportation Systems, Funchal, Portugal.
8. Decker K, Fisher M, Luck M, Tennenholtz M, UKMAS Contributor (1999) Continuing Research in Multi-Agent Systems. The Knowledge Engineering Review 14: 279-283.
9. Rana O, Preist C, Luck M (2000) Progress in Multi-Agent Systems Research. UKMAS, UK.
10. Wang Hairui (2008) Research and Implementation of Multi-agent Based Test Paper Generation Algorithm. International Conference on Computer Science and Software Engineering, Wuhan, Hubei.