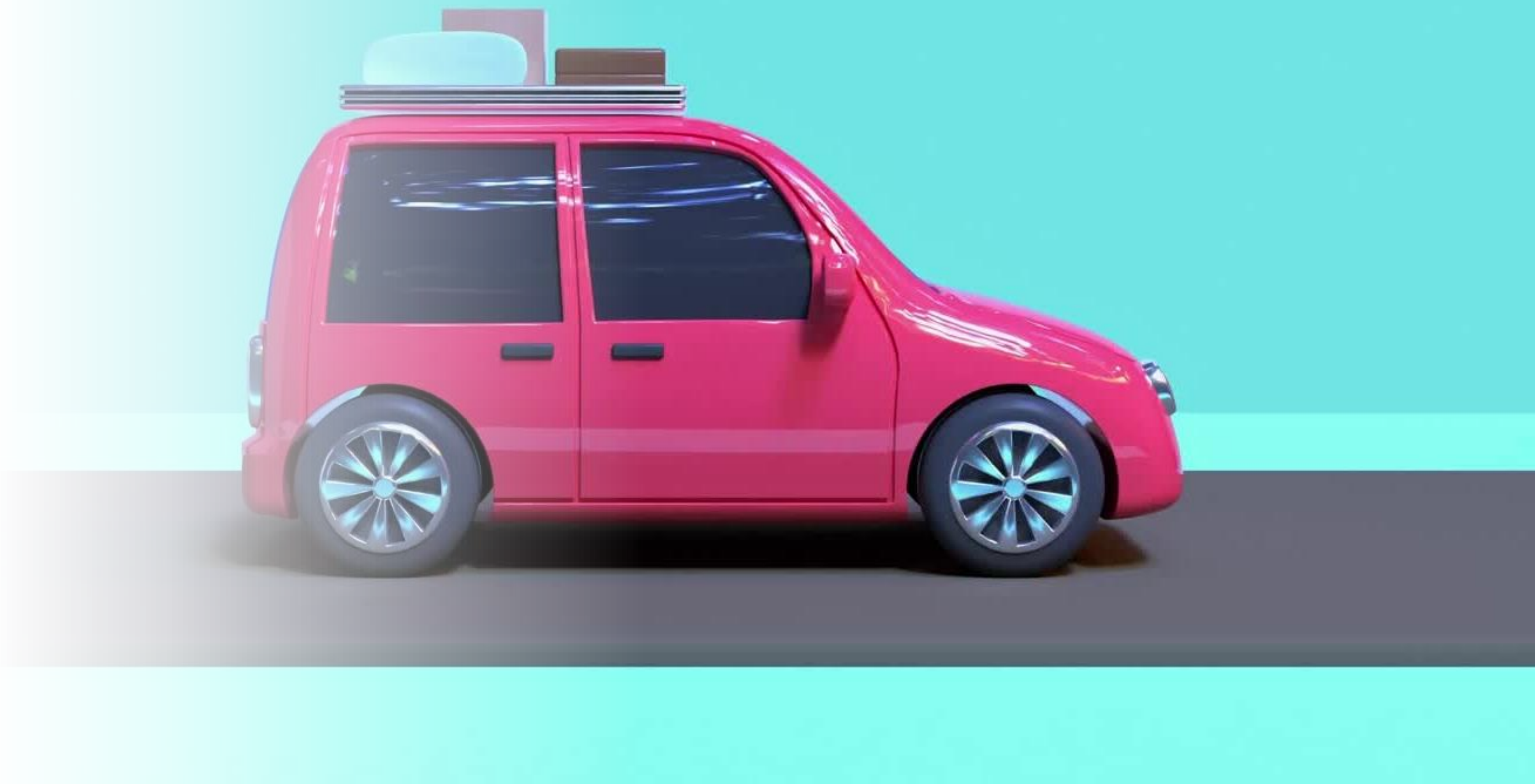
Project proposal for P&R:

Efficient Temporal and Resource-Based Automated Car Washing System



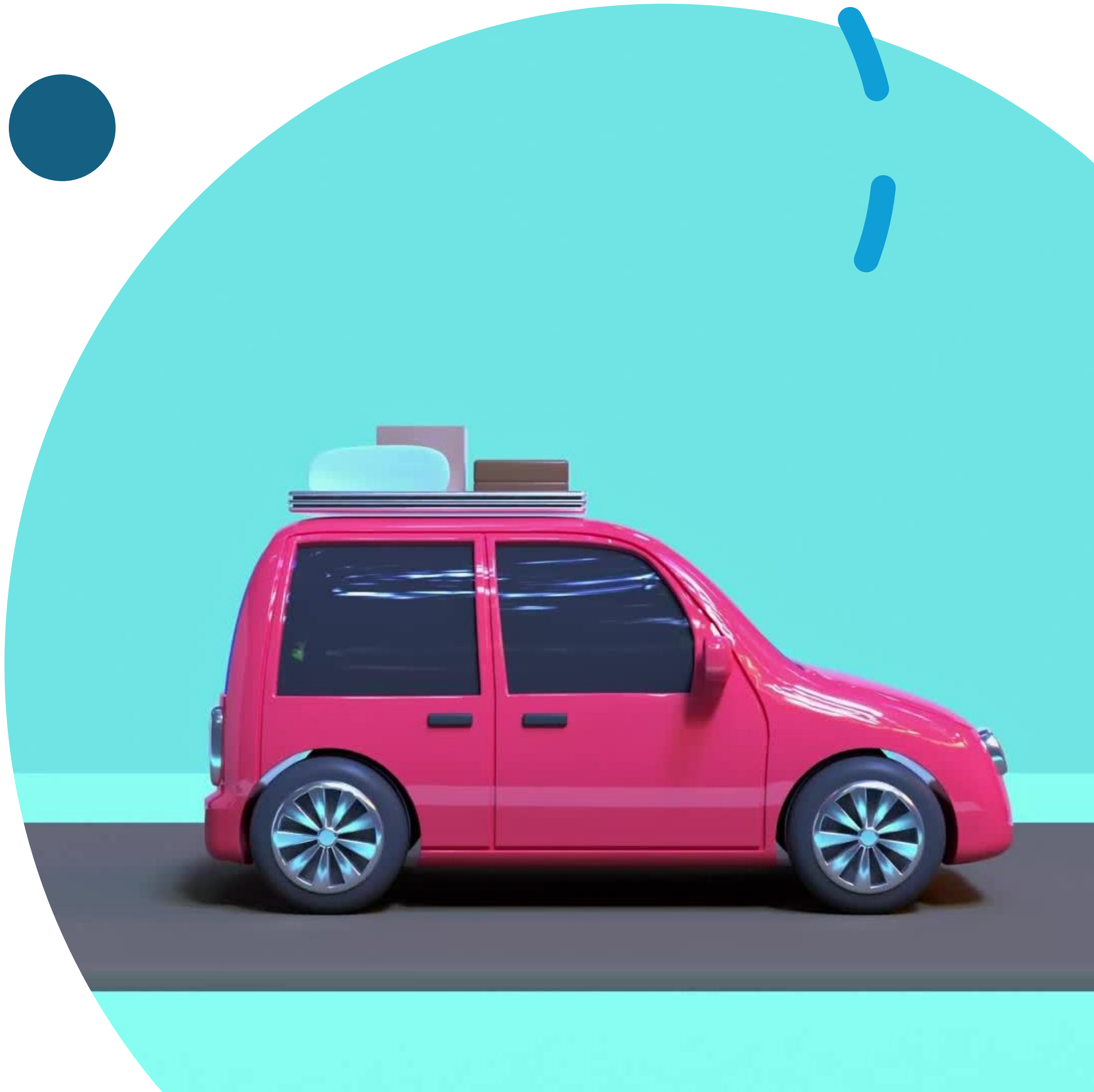
- Filippo Croce
- Federico Occelli

ABOUT THE IDEA

The idea for this project is to develop efficient automated car washing facilities that can handle multiple vehicle types, multiple cleaning program while managing resources effectively.

- **Different vehicle sizes such as small car, big car, motocycle, tir.**
- **Various cleaning programs such as fast, basic, premium with different resource requirements and durations**
- **Resource management including water, soap and wax,**
- **Optimal scheduling to minimize wait times and maximize throughput**

The goal is to minimize cleaning time, efficiently allocate resources, and maximize station throughput while ensuring that resource replenishment and scheduling constraints are respected. The domain also incorporates temporal elements, such as cleaning durations, resource replenishment delays, and scheduling conflicts.



THE DOMAIN

- Multiple cleaning stations
- Various vehicle types requiring different cleaning approaches
- Limited resources that need to be managed and replenished
- Different cleaning programs with varying durations and resource requirements

The predicates

•(vehicle_at ?v - vehicle ?s - station)	TRUE if a vehicle is at a certain station
•(free_vehicle ?v – vehicle)	TRUE if a vehicle is free
•(station_available ?s - station)	TRUE if a cleaning station is available for use
•(has_resources ?s - station ?r - resource)	TRUE if a station has sufficient resources
•(program_compatibility ?p - program ?v - vehicle)	TRUE if a cleaning program is compatible with a vehicle type
•(program_selected ?p - program ?v – vehicle)	TRUE if a cleaning program is selected and compatible with a vehicle type
•(cleaning_started?v - vehicle ?s - station)	TRUE if a vehicle is being cleaned at a station
•(resource_need_refil ?s – station ?r - resource)	TRUE if a station needs maintenance
•(station_ready ?s – station ?r –resource ?r –resource ?r –resource)	TRUE if a station has all the resource available

•V = (small car, big car, motocycle, tir)
•S = 3
•R= (water tank, active soap tank, wax tank)
•P = (fast, basic, premium, moto)

fast = water –
basic= soap, water
premium = soap, water, wax
moto = soap water

THE GENERAL PROBLEM

In the problem file we can make different choices and model whatever we want such as:

- **Number of cleaning stations**
- **Number of vehicles**
- **Types of vehicles (motorcycle, car, van, truck)**
- **Available resources**
- **Cleaning program types**
- **Resource consumption rates**
- **Cleaning durations**

In the Initial State we set:

- **Station availability**
- **Resource levels**
- **Vehicle positions**
- **Resource consumption rates**
- **Cleaning duration parameters**

The Goal is to clean all vehicles according to their required programs while:

- **Minimizing total cleaning time**
- **Optimizing resource usage**
- **Maintaining proper service quality**
- **Ensuring efficient station utilization**

3 INSTANCES OF THE PROBLEM

PROBLEM 1: Basic Setup

- 2 stations
- 3 vehicles (2 cars, 1 motorcycle)
- Basic resources (water, soap)
- Limited resources: 100 liters of water and 50 liters of soap
- 2 cleaning programs (Basic, Premium)
- *Goal*: Clean all vehicles while ensuring no resource depletion.

PROBLEM 2: Multiple Station Types

•3 stations
•5 vehicles (3 cars, 1 motorcycle, 1 van)
•Extended resources (water, soap, wax)
•Resources include water (200 liters), soap (100 liters), and wax (50 liters).
•3 cleaning programs (fast, basic, premium)
•*Goal*: Optimize cleaning while managing resource replenishment.

PROBLEM 3: Full System

•6 stations
•12 vehicles (all types)
•Complete resource management
•Large but constrained resources: 50
liters of water, 250 liters of soap, and
liters of wax.
•All cleaning programs
•Maintenance and scheduling
•*Goal*: Optimize cleaning while managing resource replenishment.

The planner

To solve the numerical planning problem, we have decided to use ENHSP-20 (Expressive Numeric Heuristic Search Planner) which is a PDDL automated planning system that supports both Classical and Numeric Planning (PDDL2.1).

Different configurations tested:

- sat-hmrp: For basic resource management
- sat-hadd: For complex scheduling problems
- opt-hmax: For optimization scenarios

The chosen metric to be minimized is a combination of:

- Total cleaning time
- Resource consumption
- Number of station switches

Search Heuristics

The following heuristics will be employed:

- 1. *Hmax (Maximum Heuristic)*: To estimate the shortest path to achieving goals based on maximal resource usage.
- 2. *Hadd (Additive Heuristic)*: For considering cumulative resource effects when multiple actions contribute to achieving a goal.
- 3. *Temporal Relaxation*: Ensures scheduling of actions considers time constraints.

Reasoning Tasks with IndiGolog

We will define three reasoning tasks using IndiGolog and the SWI-Prolog interpreter:

- *Task 1: Dynamic Scheduling* - *Objective*: Develop a program to dynamically assign vehicles to available stations based on current workloads and resource levels. - *Input*: Real-time updates on station availability and vehicle queue. - *Output*: Updated schedule that minimizes waiting time
- *Task 2: Resource Management* - *Objective*: Reason about resource consumption and determine when to initiate replenishment actions. - *Input*: Current resource levels and consumption rates. - *Output*: Trigger actions for resource replenishment to avoid station downtime
- *Task 3: Handling Unexpected Events* - *Objective*: Implement a program to respond to unplanned events, such as resource shortages or station failures. - *Input*: Event notifications and system state. - *Output*: Adjusted schedules and resource allocations to minimize disruptions.