RoboChart: Practical 3 - solutions

Software Engineering for Robotics

Pedro Ribeiro

ROBOSTAR

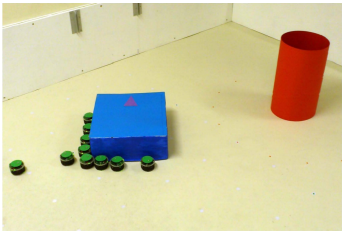
robostar.cs.york.ac.uk

Royal Academy of Engineering

EPSRC
Engineering and Physical Sciences Research Council

Practical 3: Exercise

Model the control software of a robot that by itself cannot push an object, but when combined in a swarm can achieve the desired goal of transporting a larger object.

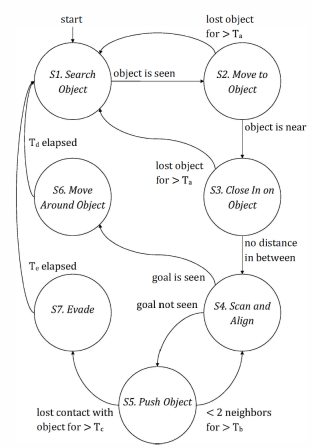


J. Chen, M. Gauci, and R. Gross, "A strategy for transporting tall objects with a swarm of miniature mobile robots," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 863–869, May 2013.

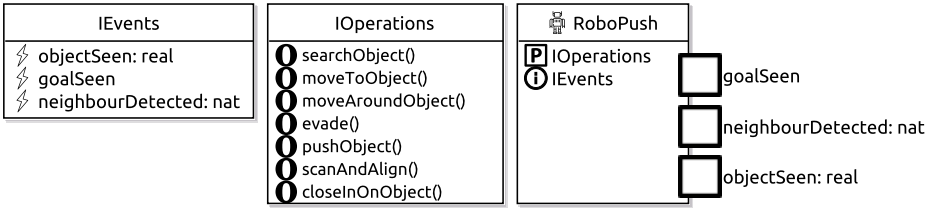
Practical 3: Exercise

Modelling

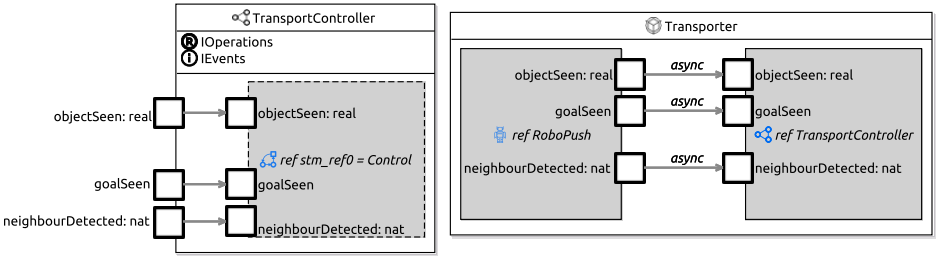
1. Define the **robotic platform** based on the events, variables and operations used in the state machine.
2. Write the state machine on the right in RoboChart.
3. Define the module: controller(s) and connections to the platform.



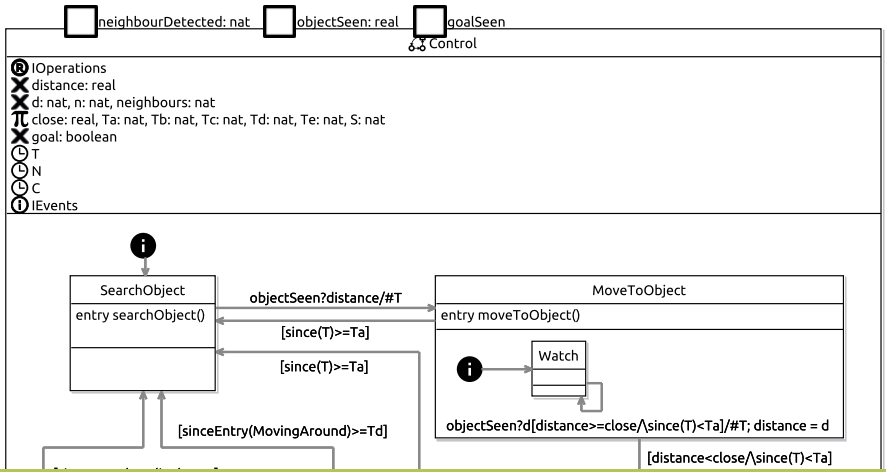
Practical 3: Robotic platform



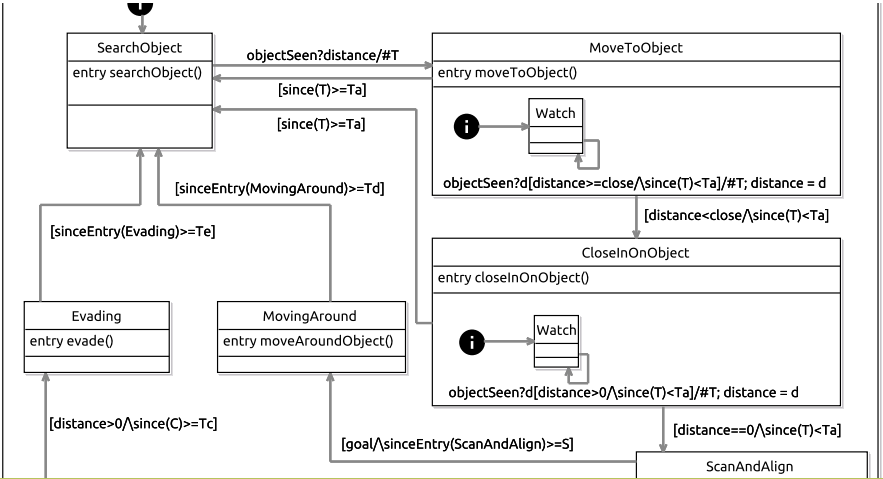
Practical 3: Module and controller



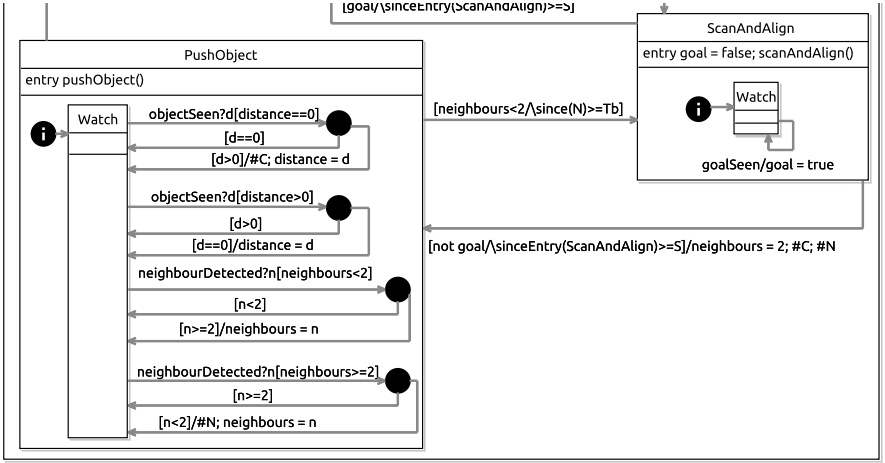
Practical 3: State machine (1/3)



Practical 3: State machine (2/3)



Practical 3: State machine (3/3)



Practical 3: Verification (1/3)

Core properties.

```
// Timed deadlock freedom
timed assertion C0 : Transporter is deadlock-free with
    constant Ta  of Control     set to 1,
    constant Tb  of Control     set to 0,
    constant Tc  of Control     set to 0,
    constant Td  of Control     set to 0,
    constant Te  of Control  set to 0,
    constant close of Control set to 1,
    and constant S of Control set to 2
```

Practical 3: Verification (2/3)

At least t time units is spent in ScanAndAlign.

```
timed csp P1_2s associated to Transporter csp_begin
Timed(OneStep) {
 AnyUntil(x) = TCHAOS(TransporterRPEvents) [|x|> SKIP
 Scanning(t) =
  let
   Main = AnyUntil({|Transporter::scanAndAlignCall|});
          TimedInterrupt(
            TCHAOS(diff(Events,{|pushObjectCall,moveAroundObjectCall|})),
            t);
          Main
  within
   timed_priority(Main)

 P1_2s = Scanning(2)
}
csp_end
```

Practical 3: Verification (3/3)

Verification via refinement checking.

```
timed assertion AP1_ok : Transporter refines P1_2s in the tick-tock model
  with  constant Ta of Control set to 1,
        constant Tb of Control set to 1,
        constant Tc of Control set to 1,
        constant Td of Control set to 1,
        constant Te of Control set to 1,
        constant close of Control set to 1,
        and constant S of Control set to 2
```

Practical 3: Solutions

Project with solutions as presented

robostar.cs.york.ac.uk/robotool/tutorial/Transporter.zip

For further details see the **Transport** entry under case studies in the RoboStar page.

robostar.cs.york.ac.uk/case_studies/transport/transport.html