

Massachusetts Institute of Technology

16.412J/6.834J Intelligent Embedded Systems

Problem Set #2

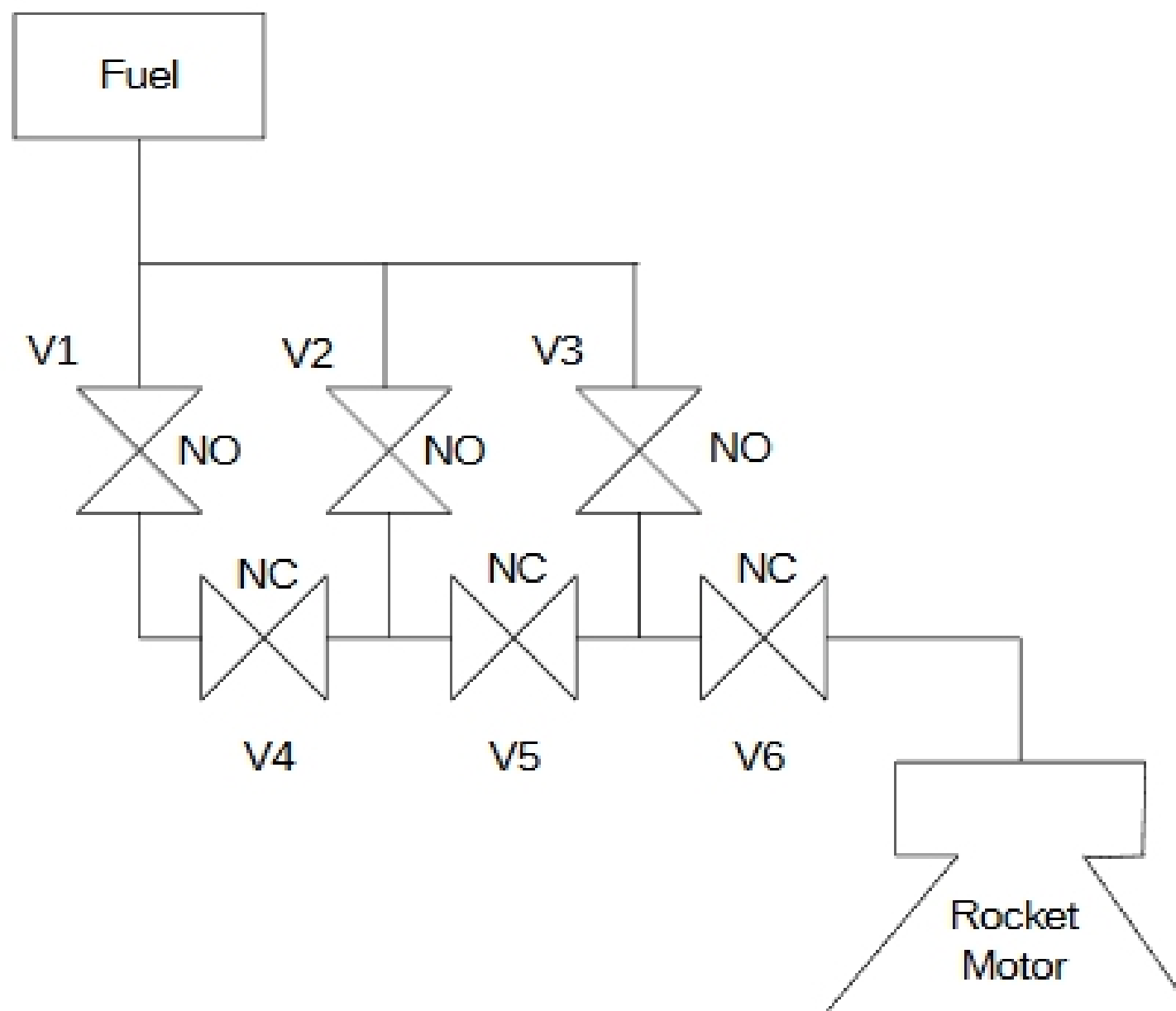
Due: in class Wed, 10/2/02

Background

The purpose of this problem set is to develop a grounded understanding of planning problems and their solution. This includes understanding of how such problems are formulated, of the variety of algorithms that can be used for solution, and of the limits of these formulations and algorithms.

Problem 1 – Simple Planning Problem

Consider the following valve network.



Valves V1 – V6 are pyro valves. Pyro valves are initially in one particular state (open or closed). An explosive bolt can be fired that switches a pyro valve to its other state. Thus, an important disadvantage of a pyro valve (with respect to a typical electrically activated on-off valve) is that a pyro valve can switch states only once. The advantage of using a pyro valve is that it is extremely reliable. It will stay in its initial state until it is fired. When it is fired, it is guaranteed to switch to the opposite state where it will remain.

In the above diagram, valves V1 – V3 are initially open (indicated by NO for normally open). Firing these closes them. Valves V4 – V6 are initially closed (NC for normally closed). Firing these opens them.

Now, consider the following STRIPS formulation of facts and operators for this problem (refer to Ch. 11 of the Russell and Norvig text for a description of the STRIPS language).

Operators:

```
(operator
  fire-open-1
  (params (<vout> Valve) (<vtop> Valve))
  (preconds
    (top-input-to <vtop> <vout>)
    (normally-closed <vout>) (normally-open <vtop>))
  (effects
    (del normally-closed <vout>) (fired-open <vout>)
    (del fuel-not-flowing <vout>) (fuel-flowing <vout>)))

(operator
  fire-open-2
  (params (<vout> Valve) (<vtop> Valve) (<vbottom> Valve))
  (preconds
    (top-input-to <vtop> <vout>) (bottom-input-to <vbottom> <vout>)
    (normally-closed <vout>) (fired-closed <vtop>) (fuel-flowing
<vbottom>))
  (effects
    (del normally-closed <vout>) (fired-open <vout>)
    (del fuel-not-flowing <vout>) (fuel-flowing <vout>)))

(operator
  fire-open-3
  (params (<vout> Valve) (<vtop> Valve) (<vbottom> Valve))
  (preconds
    (top-input-to <vtop> <vout>) (bottom-input-to <vbottom> <vout>)
    (normally-closed <vout>) (fired-closed <vtop>) (fuel-not-flowing
<vbottom>))
  (effects
    (del normally-closed <vout>) (fired-open <vout>)
    (del fuel-flowing <vout>) (fuel-not-flowing <vout>)))

(operator
  fire-close-top-1
  (params (<vout> Valve) (<vtop> Valve) (<vbottom> Valve))
  (preconds
    (top-input-to <vtop> <vout>) (bottom-input-to <vbottom> <vout>)
    (normally-open <vtop>) (fired-open <vout>) (fuel-flowing <vbottom>))
  (effects
    (del normally-open <vtop>) (fired-closed <vtop>)
    (del fuel-not-flowing <vout>) (fuel-flowing <vout>)))

(operator
  fire-close-top-2
  (params (<vout> Valve) (<vtop> Valve) (<vbottom> Valve))
  (preconds
    (top-input-to <vtop> <vout>) (bottom-input-to <vbottom> <vout>)
    (normally-open <vtop>) (fired-open <vout>) (fuel-not-flowing
<vbottom>))
  (effects
    (del normally-open <vtop>) (fired-closed <vtop>)
    (del fuel-flowing <vout>) (fuel-not-flowing <vout>)))
```