TDT4136 Logic and Reasoning Systems

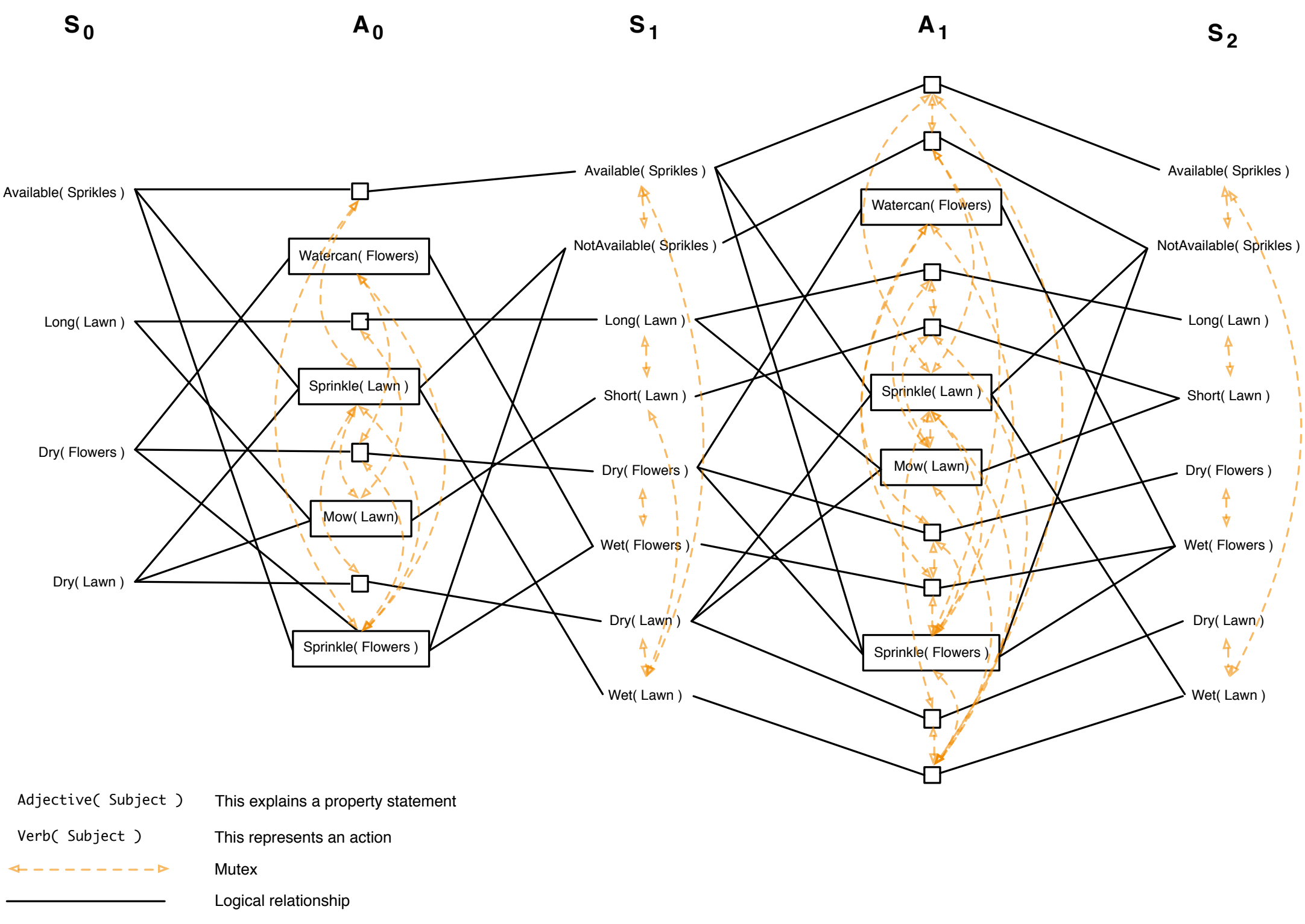
Jørgen Grimnes
Assignment 6

Fall 2013

1 Exercise 1

Planning graph I chose to draw the planning graph visually by using a modeling tool. There is a legend on the drawing which explains the meaning of the different colors and symbols.

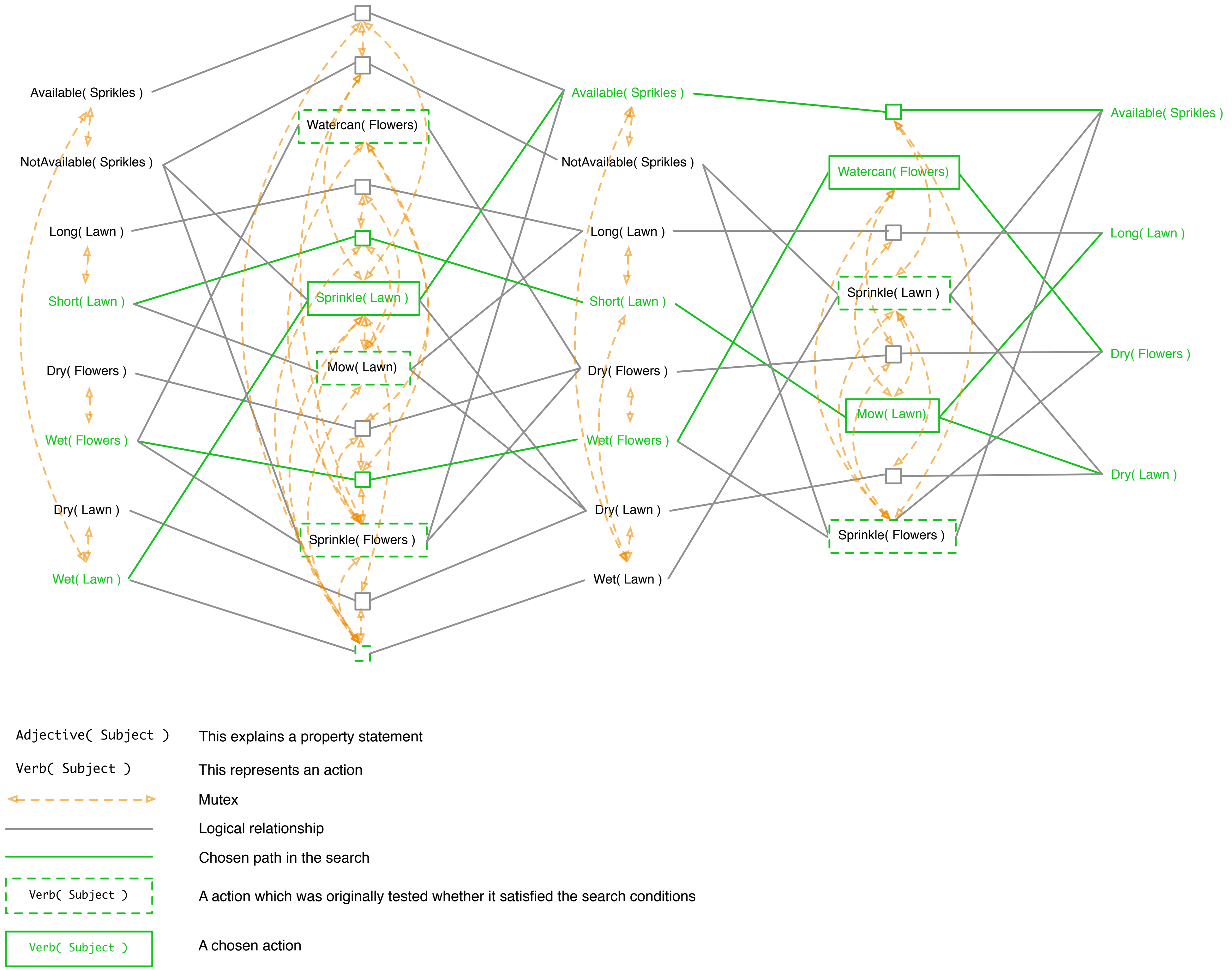
The graphical representation may been seen on the next page.


S0
A0
S1
A1
S2
Available(Sprikles)
Long(Lawn)
Dry(Flowers)
Dry(Lawn)
Watercan(Flowers)
Sprinkle(Lawn)
Mow(Lawn)
Sprinkle(Flowers)
Available(Sprikles)
NotAvailable(Sprikles)
Long(Lawn)
Short(Lawn)
Dry(Flowers)
Wet(Flowers)
Dry(Lawn)
Wet(Lawn)
Watercan(Flowers)
Sprinkle(Lawn)
Mow(Lawn)
Sprinkle(Flowers)
Available(Sprikles)
NotAvailable(Sprikles)
Long(Lawn)
Short(Lawn)
Dry(Flowers)
Wet(Flowers)
Dry(Lawn)
Wet(Lawn)
Adjective(Subject)
This explains a property statement
Verb(Subject)
This represents an action
Mutex
Logical relationship

Expanding levels

S_1 As we can observe from the graphical representation on the previous page, we have a mutex relationship between two of our goal states. *Wet(Lawn)* and *Short(Lawn)* has a mutex line drawn between them. Thus the graph had to be expanded another level.

S_2 We observe that the previous, problematic mutex have vanished. We try to perform the *Extract Solution* procedure. The result from this procedure can be seen on the next page.


Available(Sprikles)
NotAvailable(Sprikles)
Long(Lawn)
Short(Lawn)
Dry(Flowers)
Wet(Flowers)
Dry(Lawn)
Wet(Lawn)
Watercan(Flowers)
Sprinkle(Lawn)
Mow(Lawn)
Sprinkle(Flowers)
Available(Sprikles)
NotAvailable(Sprikles)
Long(Lawn)
Short(Lawn)
Dry(Flowers)
Wet(Flowers)
Dry(Lawn)
Wet(Lawn)
Watercan(Flowers)
Sprinkle(Lawn)
Mow(Lawn)
Sprinkle(Flowers)
Available(Sprikles)
Long(Lawn)
Dry(Flowers)
Dry(Lawn)
Adjective(Subject)
This explains a property statement
Verb(Subject)
This represents an action
Mutex
Logical relationship
Chosen path in the search
Verb(Subject)
A action which was originally tested whether it satisfied the search conditions
Verb(Subject)
A chosen action

Found solution

Since the search returned a complete and valid plan, we can terminate the algorithm. I have taken into account that the gardening agent may perform two actions in a single time slot (during A_0).

Action list:

```
[ {Watercan(Flowers), Mow(Lawn)}, {Sprinkle(Lawn)} ]
```