

# Report on Assignment for Automated Planning: Theory and Practice

By Fariha Al Ferdous

ID: 238888

Program: Artificial Intelligence Systems

This report is for the assignment for the final exam of Automated Planning course. The main goal of all the planning problems in the given Assignment here were to supply all the work stations with either all, or some of the contents. And the contents could be either bolts or valves or tools. In the first problem, only one robot has been supposed to act. And from the second to all of the rest of the problems, multiple robots with each having one carrier, were assigned. Here, in my domain definitions I have assumed that boxes could be reloaded. The goal of the project was to having all the workstations have been supplied with either all type of contents or some of the contents. Locations are either connected to each other or not. But the main location, which is the central\_warehouse, is connected to all locations.

In all of my domains, I have assumed that locations are connected. Further manipulation could be done to the code if needed for more complex scenarios. I have considered 3 robots, each with one carrier and varying capacities. Total boxes are 3 and total locations including central\_warehouse are 3. One location has 2 workstations and the other one has one. So there are 3 workstations total. But however, central\_warehouse doesn't have any workstations. I have used a generic type content here and 3 specific other content types representing bolts, valves and tools.

In the submitted archive, following are the contents:

- **Problem1: workstation**  
(command: downward --alias seq-sat-lama-2011 /root/pddl/workstation/domain1.pddl /root/pddl/workstation/problem1.pddl )
- **Problem2: Carrier**  
(command: downward --alias seq-sat-lama-2011 /root/pddl/carrier/domain2.pddl /root/pddl/carrier/problem2.pddl

- **Problem3: htnproblem**  
(command: `java -jar PANDA.jar -parser hddl domain2.hddl problem2.hddl`)
- **Problem4: durativeaction**  
(command: `tfd pddl/durativeaction/domain2.pddl pddl/durativeaction/problem2.pddl`)
- **Problem5: plansys2\_project**  
(command: `ros2 launch plansys2_project plansys2_project.launch.py`)

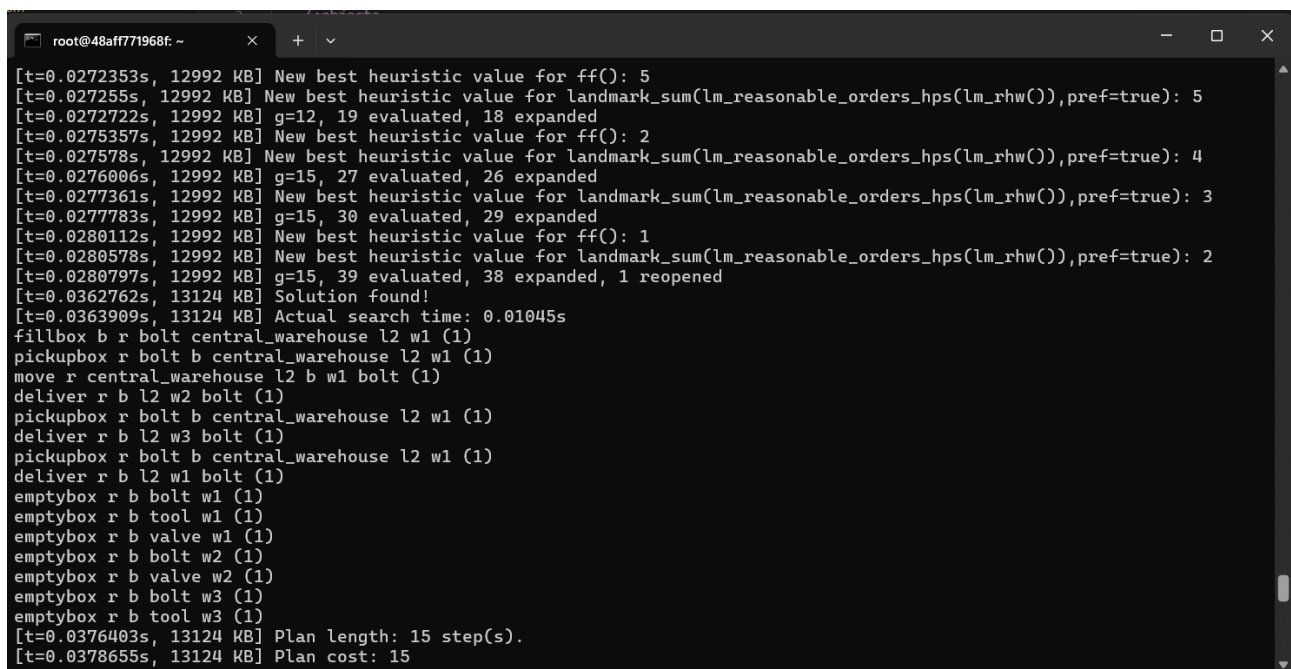
## PROBLEM 1:

### Domain Definition:

I have defined 16 predicates and 5 types of objects. For fulfilling the goal I have defined 6 actions. Basically the work flow for my defined domain would be:

fillbox > pickupbox > move > deliver > emptybox > return

For the compilation I have used llama algorithm using downward planner. And from the screenshot you can see the goal could be achieved and multiple solutions could be found.



```
root@48aff771968f: ~  
[t=0.0272353s, 12992 KB] New best heuristic value for ff(): 5  
[t=0.027255s, 12992 KB] New best heuristic value for landmark_sum(lm_reasonable_orders_hps(lm_rhw()),pref=true): 5  
[t=0.0272722s, 12992 KB] g=12, 19 evaluated, 18 expanded  
[t=0.0275357s, 12992 KB] New best heuristic value for ff(): 2  
[t=0.027578s, 12992 KB] New best heuristic value for landmark_sum(lm_reasonable_orders_hps(lm_rhw()),pref=true): 4  
[t=0.0276006s, 12992 KB] g=15, 27 evaluated, 26 expanded  
[t=0.0277361s, 12992 KB] New best heuristic value for landmark_sum(lm_reasonable_orders_hps(lm_rhw()),pref=true): 3  
[t=0.0277783s, 12992 KB] g=15, 30 evaluated, 29 expanded  
[t=0.0280112s, 12992 KB] New best heuristic value for ff(): 1  
[t=0.0280578s, 12992 KB] New best heuristic value for landmark_sum(lm_reasonable_orders_hps(lm_rhw()),pref=true): 2  
[t=0.0280797s, 12992 KB] g=15, 39 evaluated, 38 expanded, 1 reopened  
[t=0.0362762s, 13124 KB] Solution found!  
[t=0.0363909s, 13124 KB] Actual search time: 0.01045s  
fillbox b r bolt central_warehouse l2 w1 (1)  
pickupbox r bolt b central_warehouse l2 w1 (1)  
move r central_warehouse l2 b w1 bolt (1)  
deliver r b l2 w2 bolt (1)  
pickupbox r bolt b central_warehouse l2 w1 (1)  
deliver r b l2 w3 bolt (1)  
pickupbox r bolt b central_warehouse l2 w1 (1)  
deliver r b l2 w1 bolt (1)  
emptybox r b bolt w1 (1)  
emptybox r b tool w1 (1)  
emptybox r b valve w1 (1)  
emptybox r b bolt w2 (1)  
emptybox r b valve w2 (1)  
emptybox r b bolt w3 (1)  
emptybox r b tool w3 (1)  
[t=0.0376403s, 13124 KB] Plan length: 15 step(s).  
[t=0.0378655s, 13124 KB] Plan cost: 15
```

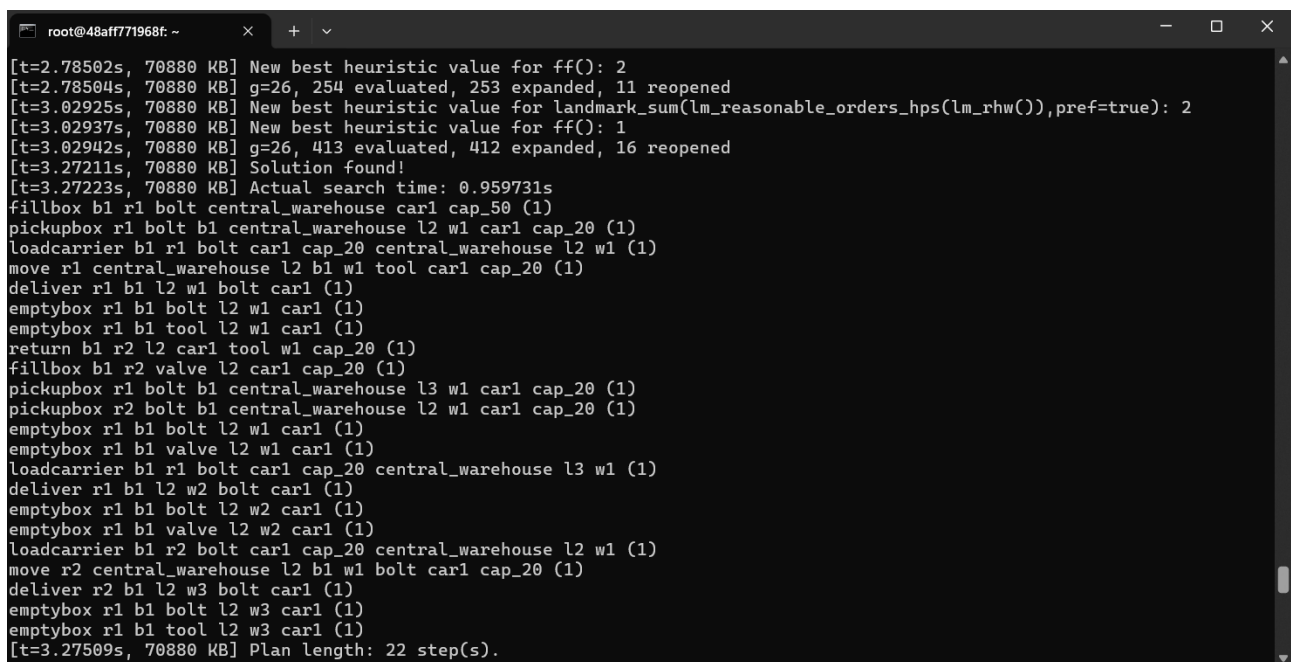
## PROBLEM 2:

### Domain Definition:

I have defined 22 predicates and 7 types of objects. For fulfilling the goal I have defined 7 actions. Since this problem involves a carrier for each robot, I have defined necessary types, predicates and actions for it. Basically the work flow for my defined domain would be:

fillbox > pickupbox > loadcarrier > move > deliver > emptybox > return

For the compilation I have used llama algorithm using downward planner. And from the screenshot we can see the goal could be achieved and multiple solutions could be found.



```
root@48aff771968f: ~
[t=2.78502s, 70880 KB] New best heuristic value for ff(): 2
[t=2.78504s, 70880 KB] g=26, 254 evaluated, 253 expanded, 11 reopened
[t=3.02925s, 70880 KB] New best heuristic value for landmark_sum(lm_reasonable_orders_hps(lm_rhw()),pref=true): 2
[t=3.02937s, 70880 KB] New best heuristic value for ff(): 1
[t=3.02942s, 70880 KB] g=26, 413 evaluated, 412 expanded, 16 reopened
[t=3.27211s, 70880 KB] Solution found!
[t=3.27223s, 70880 KB] Actual search time: 0.959731s
fillbox b1 r1 bolt central_warehouse car1 cap_50 (1)
pickupbox r1 bolt b1 central_warehouse l2 w1 car1 cap_20 (1)
loadcarrier b1 r1 bolt car1 cap_20 central_warehouse l2 w1 (1)
move r1 central_warehouse l2 b1 w1 tool car1 cap_20 (1)
deliver r1 b1 l2 w1 bolt car1 (1)
emptybox r1 b1 bolt l2 w1 car1 (1)
emptybox r1 b1 tool l2 w1 car1 (1)
return b1 r2 l2 car1 tool w1 cap_20 (1)
fillbox b1 r2 valve l2 car1 cap_20 (1)
pickupbox r1 bolt b1 central_warehouse l3 w1 car1 cap_20 (1)
pickupbox r2 bolt b1 central_warehouse l2 w1 car1 cap_20 (1)
emptybox r1 b1 bolt l2 w1 car1 (1)
emptybox r1 b1 valve l2 w1 car1 (1)
loadcarrier b1 r1 bolt car1 cap_20 central_warehouse l3 w1 (1)
deliver r1 b1 l2 w2 bolt car1 (1)
emptybox r1 b1 bolt l2 w2 car1 (1)
emptybox r1 b1 valve l2 w2 car1 (1)
loadcarrier b1 r2 bolt car1 cap_20 central_warehouse l2 w1 (1)
move r2 central_warehouse l2 b1 w1 bolt car1 cap_20 (1)
deliver r2 b1 l2 w3 bolt car1 (1)
emptybox r1 b1 bolt l2 w3 car1 (1)
emptybox r1 b1 tool l2 w3 car1 (1)
[t=3.27509s, 70880 KB] Plan length: 22 step(s).
```

### PROBLEM 3:

#### Domain Definition:

I have defined 20 predicates and 7 types of objects. For fulfilling the goal I have defined 6 actions with 4 tasks and 4 methods. Basically the work flow for my defined domain would be:

##### *Tasks:*

fill\_up\_robot > deliver\_content > return\_to\_central\_warehouse > finish\_and\_return

##### *Methods:*

fill\_up\_robot\_method > deliver\_method > return\_robot\_method > planner\_complete\_method

##### *Actions:*

fillbox > pickupbox > loadcarrier > move > deliver > emptybox > return

##### *The hierarchy would be:*

1. fill\_up\_robot > fill\_up\_robot\_method >
  - Fillbox
  - Pickupbox
  - Loadcarrier
  
2. deliver\_content > deliver\_method >
  - Move
  - Deliver
  - Emptybox
  
3. return\_to\_central\_warehouse > return\_robot\_method >
  - Return

For the compilation I have used PANDA planner and AStar search algorithm. And from the screenshot we can see tasks could be decomposed to methods and gradually to sub tasks, although the plan was unsolvable.

```
Command Prompt
size of largest method          = -1
average method size             = -1

----- TIMINGS -----
===== total =====
total = 929
===== parsing =====
total                          = 538
file parser                    = 189
sort expansion                  = 115
closed world assumption        = 70
inherit preconditions           = 0
shop methods                    = 24
eliminate identical variables   = 46
strip domain of hybridity       = 47
flatten formula                 = 41
===== preprocessing =====
total                          = 370
compile negative preconditions  = 36
compile unit methods           = 0
split parameter                 = 26
expand choiceless abstract tasks = 0
compile methods with identical tasks = 0
removing unnecessary predicates = 17
lifted reachability analysis    = 30
grounded planning graph analysis = 90
grounded task decomposition graph analysis = 91
grounding                       = 58
create artificial top task      = 0
```

```
Command Prompt

Parsing Configuration
-----
Parser                : HDDL Parser (Daniel's format)
Expand Sort Hierarchy : true
ClosedWorldAssumption : true
CompileSHOPMethods    : true
Eliminate Equality     : true
Strip Hybridity        : true
Reduce General Tasks  : true

Preprocessing Configuration
-----
Compile negative preconditions : true
Compile unit methods           : false
Compile order in methods       : false
Compile initial plan           : true
Ensure Methods Have Last Task  : false
Split independent parameters   : true
Remove unnecessary predicates   : true
Expand choiceless abstract tasks : true
Domain Cleanup                 : true
Convert to SAS+                 : false
Grounded Reachability Analysis  : Planning Graph (mutex-free)
Grounded Task Decomposition Graph : Two Way TDG
Iterate reachability analysis   : true
Ground domain                   : true
Iterate reachability analysis   : true
Stop directly after grounding   : false

Progression-search Configuration
-----
Search Algorithm           : AStarActionsType(2.0)
Heuristic                   : hhRC(hFF)
Abstract task selection strategy : random

Post-processing Configuration
-----
search status
search result
timings
statistics
```

## PROBLEM 4:

### Domain Definition:

I have defined 16 predicates and 7 types of objects. For fulfilling the goal I have defined 6 durative actions. Basically the work flow for my defined domain would be:

fillbox > pickupbox > loadcarrier > move > deliver > emptybox > return

For the compilation I have used TFD planner and AStar search algorithm. And from the screenshot we can see the goal could be achieved and multiple solutions could be found.

```
root@48aff771968f: ~  
0.00000000: (return b3 r1 central_warehouse car3 bolt1 w1) [3.00000000]  
0.01000000: (return b3 r1 central_warehouse car3 bolt2 w2) [3.00000000]  
0.02000000: (return b3 r1 central_warehouse car3 bolt3 w3) [3.00000000]  
0.03000000: (return b3 r1 central_warehouse car3 tool1 w1) [3.00000000]  
0.04000000: (return b3 r1 central_warehouse car3 tool2 w3) [3.00000000]  
0.05000000: (return b3 r1 central_warehouse car3 valve1 w1) [3.00000000]  
0.06000000: (return b3 r1 central_warehouse car3 valve2 w2) [3.00000000]  
Rescheduled Plan:  
0.00100000: (return b3 r1 central_warehouse car3 bolt1 w1) [3.00000000]  
0.00200000: (return b3 r1 central_warehouse car3 bolt2 w2) [3.00000000]  
0.00300000: (return b3 r1 central_warehouse car3 bolt3 w3) [3.00000000]  
0.00400000: (return b3 r1 central_warehouse car3 tool1 w1) [3.00000000]  
0.00500000: (return b3 r1 central_warehouse car3 tool2 w3) [3.00000000]  
0.00600000: (return b3 r1 central_warehouse car3 valve1 w1) [3.00000000]  
0.00700000: (return b3 r1 central_warehouse car3 valve2 w2) [3.00000000]  
Solution with original makespan 3.06 found (ignoring no-moving-targets-rule).  
Plan length: 7 step(s).  
Makespan : 3.007  
Rescheduled Makespan : 3.007  
Search time: 0.79 seconds - Walltime: 0.795808 seconds  
Total time: 0.82 seconds - Walltime: 0.835186 seconds  
0.00100000: (return b3 r1 central_warehouse car3 bolt1 w1) [3.00000000]  
0.00200000: (return b3 r1 central_warehouse car3 bolt2 w2) [3.00000000]  
0.00300000: (return b3 r1 central_warehouse car3 bolt3 w3) [3.00000000]  
0.00400000: (return b3 r1 central_warehouse car3 tool1 w1) [3.00000000]  
0.00500000: (return b3 r1 central_warehouse car3 tool2 w3) [3.00000000]  
0.00600000: (return b3 r1 central_warehouse car3 valve1 w1) [3.00000000]  
0.00700000: (return b3 r1 central_warehouse car3 valve2 w2) [3.00000000]
```

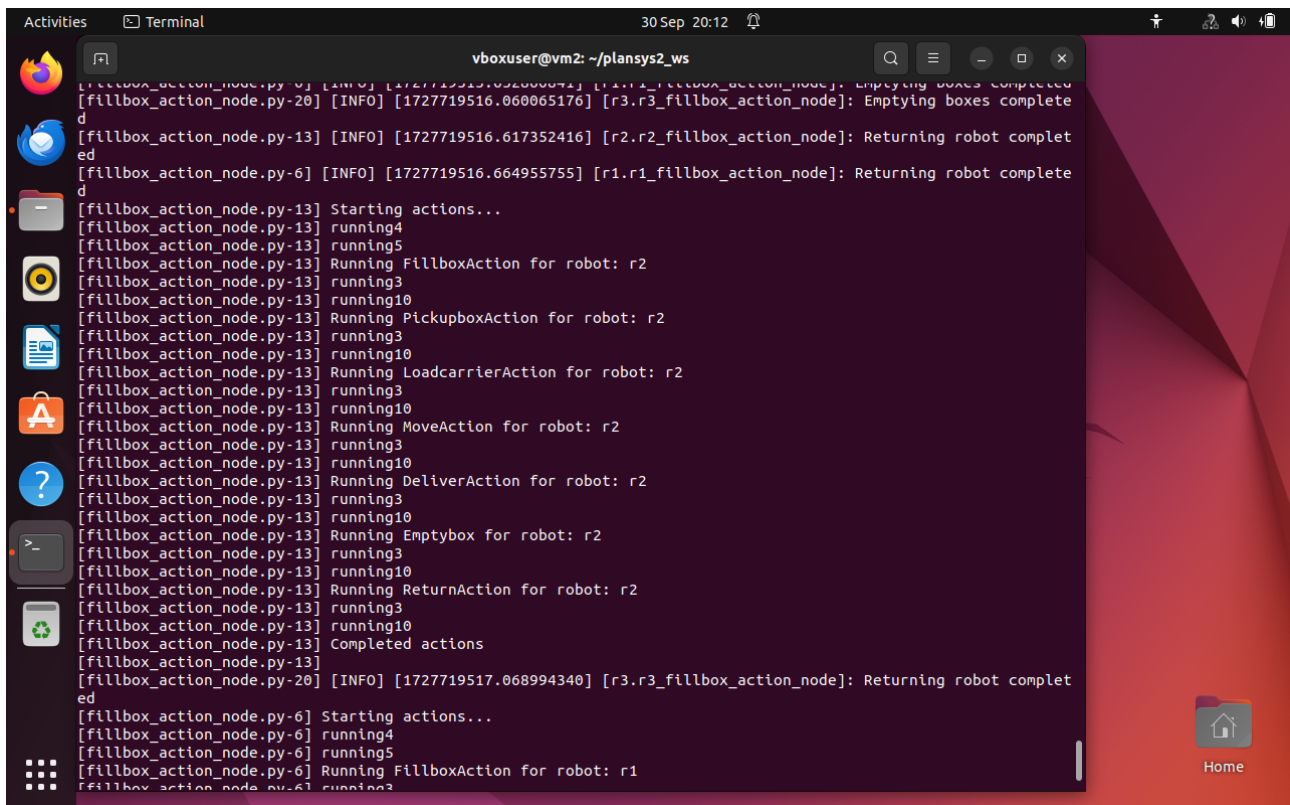
## PROBLEM 5:

### Domain Definition:

I have defined 16 predicates and 7 types of objects. For fulfilling the goal I have defined 6 durative actions. Basically the work flow for my defined domain would be:

fillbox > pickupbox > loadcarrier > move > deliver > emptybox > return

For the compilation I have used TFD planner and AStar search algorithm. And for the simulation plansys2 of ROS2 has been utilized. And from the screenshot we can see the goal could be achieved and solution could be found.



```
vboxuser@vm2: ~/plansys2_ws
[fillbox_action_node.py-6] [INFO] [1727719516.060065176] [r3.r3_fillbox_action_node]: Emptying boxes completed
[fillbox_action_node.py-20] [INFO] [1727719516.060065176] [r3.r3_fillbox_action_node]: Emptying boxes completed
[fillbox_action_node.py-13] [INFO] [1727719516.617352416] [r2.r2_fillbox_action_node]: Returning robot completed
[fillbox_action_node.py-6] [INFO] [1727719516.664955755] [r1.r1_fillbox_action_node]: Returning robot completed
[fillbox_action_node.py-13] Starting actions...
[fillbox_action_node.py-13] running4
[fillbox_action_node.py-13] running5
[fillbox_action_node.py-13] Running FillboxAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running PickupboxAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running LoadcarrierAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running MoveAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running DeliverAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running Emptybox for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running ReturnAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Completed actions
[fillbox_action_node.py-20] [INFO] [1727719517.068994340] [r3.r3_fillbox_action_node]: Returning robot completed
[fillbox_action_node.py-6] Starting actions...
[fillbox_action_node.py-6] running4
[fillbox_action_node.py-6] running5
[fillbox_action_node.py-6] Running FillboxAction for robot: r1
[fillbox_action_node.py-6] running3
```



```
Activities Terminal 30 Sep 20:11
vboxuser@vm2: ~/plansys2_ws

or provided node name. If this is due to multiple nodes with the same name then all logs for that logger name
will go out over the existing publisher. As soon as any node with that name is destructed it will unregister t
he publisher, preventing any further logs for that name from being published on the rosout topic.
[fillbox_action_node.py-6] [INFO] [1727719510.549772273] [r1.r1_fillbox_action_node]: Filling box completed
[fillbox_action_node.py-13] [INFO] [1727719510.565621986] [r2.r2_fillbox_action_node]: Filling box completed
[fillbox_action_node.py-20] [INFO] [1727719510.716328185] [r3.r3_fillbox_action_node]: Filling box completed
[fillbox_action_node.py-6] [INFO] [1727719511.575861992] [r1.r1_fillbox_action_node]: Picking up boxes complet
ed
[fillbox_action_node.py-13] [INFO] [1727719511.575823268] [r2.r2_fillbox_action_node]: Picking up boxes comple
ted
[lifecycle_manager_node-5] [ERROR] [1727719511.779802205] [lifecycle_manager_node]: Server time out while gett
ing current state for node domain_expert
[fillbox_action_node.py-20] [INFO] [1727719511.869164994] [r3.r3_fillbox_action_node]: Picking up boxes comple
ted
[fillbox_action_node.py-13] [INFO] [1727719512.579833499] [r2.r2_fillbox_action_node]: LoadcarrierAction compl
eted
[fillbox_action_node.py-6] [INFO] [1727719512.598609097] [r1.r1_fillbox_action_node]: LoadcarrierAction comple
ted
[fillbox_action_node.py-20] [INFO] [1727719512.995420329] [r3.r3_fillbox_action_node]: LoadcarrierAction compl
eted
[fillbox_action_node.py-13] [INFO] [1727719513.590832410] [r2.r2_fillbox_action_node]: Moving robot completed
[fillbox_action_node.py-6] [INFO] [1727719513.603390668] [r1.r1_fillbox_action_node]: Moving robot completed
[fillbox_action_node.py-20] [INFO] [1727719514.002359250] [r3.r3_fillbox_action_node]: Moving robot completed
[fillbox_action_node.py-13] [INFO] [1727719514.606467485] [r2.r2_fillbox_action_node]: Delivering boxes comple
ted
[fillbox_action_node.py-6] [INFO] [1727719514.619110233] [r1.r1_fillbox_action_node]: Delivering boxes complet
ed
[fillbox_action_node.py-20] [INFO] [1727719515.047842238] [r3.r3_fillbox_action_node]: Delivering boxes comple
ted
[fillbox_action_node.py-13] [INFO] [1727719515.613233488] [r2.r2_fillbox_action_node]: Emptying boxes complete
d
[fillbox_action_node.py-6] [INFO] [1727719515.632860841] [r1.r1_fillbox_action_node]: Emptying boxes completed
[fillbox_action_node.py-20] [INFO] [1727719516.060065176] [r3.r3_fillbox_action_node]: Emptying boxes complete
d
[fillbox_action_node.py-13] [INFO] [1727719516.617352416] [r2.r2_fillbox_action_node]: Returning robot complet
ed
[fillbox_action_node.py-6] [INFO] [1727719516.664955755] [r1.r1_fillbox_action_node]: Returning robot complete
d
[fillbox_action_node.py-13] Starting actions...
[fillbox_action_node.py-43] Success
```

```
Activities Terminal 30 Sep 20:09
vboxuser@vm2: ~/plansys2_ws

[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Running PickupboxAction for robot: r3
[fillbox_action_node.py-20] running3
[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Running LoadcarrierAction for robot: r3
[fillbox_action_node.py-20] running3
[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Running MoveAction for robot: r3
[fillbox_action_node.py-20] running3
[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Running DeliverAction for robot: r3
[fillbox_action_node.py-20] running3
[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Running Emptybox for robot: r3
[fillbox_action_node.py-20] running3
[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Running ReturnAction for robot: r3
[fillbox_action_node.py-20] running3
[fillbox_action_node.py-20] running10
[fillbox_action_node.py-20] Completed actions
[fillbox_action_node.py-20]
[INFO] [fillbox_action_node.py-6]: process has finished cleanly [pid 11787]
[INFO] [fillbox_action_node.py-20]: process has finished cleanly [pid 11894]
```

Activities Terminal 30 Sep 20:12

vboxuser@vm2: ~/plansys2\_ws

```
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Running ReturnAction for robot: r2
[fillbox_action_node.py-13] running3
[fillbox_action_node.py-13] running10
[fillbox_action_node.py-13] Completed actions
[fillbox_action_node.py-13]
[fillbox_action_node.py-20] [INFO] [1727719517.068994340] [r3.r3_fillbox_action_node]: Returning robot complet
ed
[fillbox_action_node.py-6] Starting actions...
[fillbox_action_node.py-6] running4
[fillbox_action_node.py-6] running5
[fillbox_action_node.py-6] Running FillboxAction for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Running PickupboxAction for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Running LoadcarrierAction for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Running MoveAction for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Running DeliverAction for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Running Emptybox for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Running ReturnAction for robot: r1
[fillbox_action_node.py-6] running3
[fillbox_action_node.py-6] running10
[fillbox_action_node.py-6] Completed actions
[fillbox_action_node.py-6]
[INFO] [fillbox_action_node.py-13]: process has finished cleanly [pid 11849]
[fillbox_action_node.py-20] Starting actions...
[fillbox_action_node.py-20] running4
[fillbox_action_node.py-20] running5
[fillbox_action_node.py-20] Running FillboxAction for robot: r3
```