Task 2.1 - Problem 1

Domain File Problem 1

```
(define (domain windfarm-1)
  (:requirements :typing :negative-preconditions)

  (:types
    uuv
    location data - object
    ship waypoint - location
    image sonar sample - data
  )

  (:predicates
    (at ?u - uuv ?l - location)
    (on-ship ?u - uuv ?s - ship)
    (connected ?from - location ?to - location)
    (can-deploy ?u - uuv)
    (uuv-deployed ?u - uuv)
    (memory-empty ?u - uuv)
    (has-data ?u - uuv ?d - data)
    (data-saved ?d - data ?l - location)
    (image-at ?img - image ?w - waypoint)
    (sonar-at ?snr - sonar ?w - waypoint)
  )

  (:action deploy-uuv
    :parameters (?u - uuv ?s - ship ?l - location)
    :precondition (and
      (on-ship ?u ?s)
      (connected ?s ?l)
      (can-deploy ?u)
    )
    :effect (and
      (not (on-ship ?u ?s))
      (at ?u ?l)
      (not (can-deploy ?u))
      (uuv-deployed ?u)
    )
  )
```

```
(:action move-uuv
   :parameters (?u - uuv ?from ?to - location)
   :precondition (and
      (at ?u ?from)
      (connected ?from ?to)
      (uuv-deployed ?u)
   )
   :effect (and
      (not (at ?u ?from))
      (at ?u ?to)
   )
)

(:action capture-image
   :parameters (?u - uuv ?w - waypoint ?img - image)
   :precondition (and
      (at ?u ?w)
      (image-at ?img ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?img)
      (not (memory-empty ?u))
   )
)

(:action perform-sonar-scan
   :parameters (?u - uuv ?w - waypoint ?snr - sonar)
   :precondition (and
      (at ?u ?w)
      (sonar-at ?snr ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?snr)
      (not (memory-empty ?u))
   )
)
```

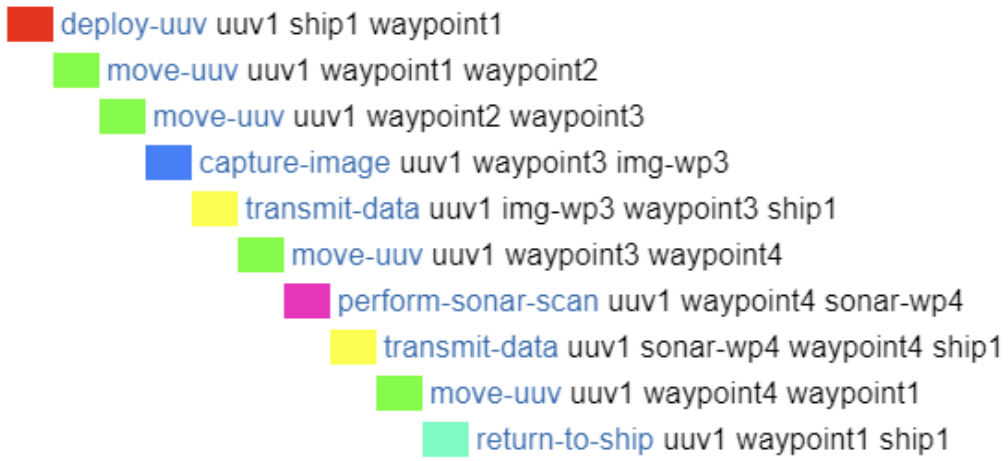
```
  (:action transmit-data
     :parameters (?u - uuv ?d - data ?l - location ?s - ship)
     :precondition (and
        (at ?u ?l)
        (has-data ?u ?d)
     )
     :effect (and
        (not (has-data ?u ?d))
        (data-saved ?d ?l)
        (memory-empty ?u)
     )
  )

  (:action return-to-ship
     :parameters (?u - uuv ?w - waypoint ?s - ship)
     :precondition (and
        (at ?u ?w)
        (connected ?w ?s)
     )
     :effect (and
        (not (at ?u ?w))
        (on-ship ?u ?s)
        (can-deploy ?u)
        (not (uuv-deployed ?u))
     )
  )
)
```

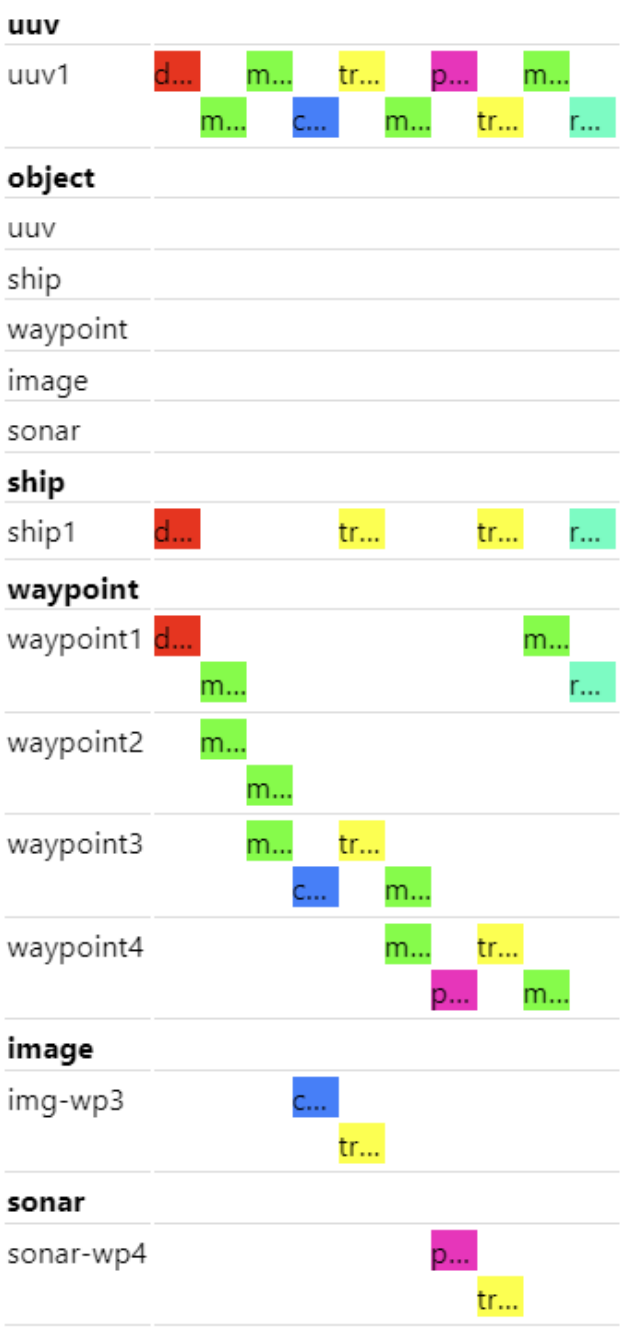
Problem File Problem 1

```
(define (problem windfarm-mission-1)
(:domain windfarm-1)

   (:objects
      uuv1 - uuv
      ship1 - ship
      waypoint1 waypoint2 waypoint3 waypoint4 - waypoint
      img-wp3 - image
      sonar-wp4 - sonar
   )

   (:init
      ; UUV starts on ship1
      (on-ship uuv1 ship1)

      ; Image and sonar data locations
      (image-at img-wp3 waypoint3)
      (sonar-at sonar-wp4 waypoint4)
      ; Ship-waypoint connections
      (connected ship1 waypoint1)
      (connected waypoint1 ship1)
      ; Bidirectional connections between waypoints
      (connected waypoint1 waypoint2)
      (connected waypoint2 waypoint1)
      (connected waypoint3 waypoint4)
      (connected waypoint4 waypoint3)
      ; Unidirectional connections
      (connected waypoint2 waypoint3)
      (connected waypoint4 waypoint1)

      ; Initial UUV state
      (memory-empty uuv1)
      (can-deploy uuv1)
   )
   (:goal (and
      (data-saved img-wp3 waypoint3)
      (data-saved sonar-wp4 waypoint4)
      (on-ship uuv1 ship1)
      )

   )
)
```

Found Plan (output)

(deploy-uuv uuv1 ship1 waypoint1)
(move-uuv uuv1 waypoint1 waypoint2)
(move-uuv uuv1 waypoint2 waypoint3)
(capture-image uuv1 waypoint3 img-wp3)
(transmit-data uuv1 img-wp3 waypoint3 ship1)
(move-uuv uuv1 waypoint3 waypoint4)
(perform-sonar-scan uuv1 waypoint4 sonar-wp4)
(transmit-data uuv1 sonar-wp4 waypoint4 ship1)
(move-uuv uuv1 waypoint4 waypoint1)
(return-to-ship uuv1 waypoint1 ship1)


deploy-uuv uuv1 ship1 waypoint1
move-uuv uuv1 waypoint1 waypoint2
move-uuv uuv1 waypoint2 waypoint3
capture-image uuv1 waypoint3 img-wp3
transmit-data uuv1 img-wp3 waypoint3 ship1
move-uuv uuv1 waypoint3 waypoint4
perform-sonar-scan uuv1 waypoint4 sonar-wp4
transmit-data uuv1 sonar-wp4 waypoint4 ship1
move-uuv uuv1 waypoint4 waypoint1
return-to-ship uuv1 waypoint1 ship1


uuv
uuv1
d... m... tr... p... m...
m... c... m... tr... r...
object
uuv
ship
waypoint
image
sonar
ship
ship1
d... tr... tr... r...
waypoint
waypoint1
d... m...
m... r...
waypoint2
m...
m...
waypoint3
m... tr...
c... m...
waypoint4
m... tr...
p... m...
image
img-wp3
c...
tr...
sonar
sonar-wp4
p...
tr...

Match tree built with 22 nodes.

PDDL problem description loaded:
Domain: WINDFARM-1
Problem: WINDFARM-MISSION-1
#Actions: 22
#Fluents: 21
Goals found: 3
Goals_Edges found: 3
Starting search with 1-BFWS...
--[2 / 0]--
--[2 / 2]--
--[2 / 3]--
--[2 / 4]--
--[2 / 5]--
--[2 / 6]--
--[2 / 7]--
--[1 / 0]--
--[1 / 4]--
--[0 / 0]--
--[0 / 3]--
Total time: 0.000163999
Nodes generated during search: 34
Nodes expanded during search: 17
Plan found with cost: 10
Fast-BFS search completed in 0.000163999 secs

Plan found:
0.00000: (DEPLOY-UUV UUV1 SHIP1 WAYPOINT1)
0.00100: (MOVE-UUV UUV1 WAYPOINT1 WAYPOINT2)
0.00200: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT3)
0.00300: (CAPTURE-IMAGE UUV1 WAYPOINT3 IMG-WP3)
0.00400: (TRANSMIT-DATA UUV1 IMG-WP3 WAYPOINT3 SHIP1)
0.00500: (MOVE-UUV UUV1 WAYPOINT3 WAYPOINT4)
0.00600: (PERFORM-SONAR-SCAN UUV1 WAYPOINT4 SONAR-WP4)
0.00700: (TRANSMIT-DATA UUV1 SONAR-WP4 WAYPOINT4 SHIP1)
0.00800: (MOVE-UUV UUV1 WAYPOINT4 WAYPOINT1)
0.00900: (RETURN-TO-SHIP UUV1 WAYPOINT1 SHIP1)
Metric: 0.009000000000000001
Makespan: 0.009000000000000001
States evaluated: undefined
Planner found 1 plan(s) in 3.489secs.

Task 2.2 - Problem 2

Domain File Problem 2

```
(define (domain windfarm-2)
   (:requirements :typing :negative-preconditions)
   (:types
      uuv - object
      location data - object
      ship waypoint - location
      image sonar sample - data
   )
   (:predicates
      (at ?u - uuv ?l - location)
      (on-ship ?u - uuv ?s - ship)
      (connected ?from - location ?to - location)
      (can-deploy ?u - uuv)
      (uuv-deployed ?u - uuv)
      (memory-empty ?u - uuv)
      (has-data ?u - uuv ?d - data)
      (data-saved ?d - data ?l - location)
      (sample-at ?s - sample ?w - waypoint)
      (image-at ?img - image ?w - waypoint)
      (sonar-at ?snr - sonar ?w - waypoint)
      (sample-stored ?s - sample ?ship - ship)
      (can-store-sample ?ship - ship)
   )
   (:action deploy-uuv
      :parameters (?u - uuv ?s - ship ?l - location )
      :precondition (and
         (on-ship ?u ?s)
         (connected ?s ?l)
         (can-deploy ?u)
      )
      :effect (and
         (not (on-ship ?u ?s))
         (at ?u ?l)
         (not (can-deploy ?u))
         (uuv-deployed ?u)
      )
   )
```

```
(:action move-uuv
   :parameters (?u - uuv ?from ?to - location)
   :precondition (and
      (at ?u ?from)
      (connected ?from ?to)
      (uuv-deployed ?u)
   )
   :effect (and
      (not (at ?u ?from))
      (at ?u ?to)
   )
)

(:action capture-image
   :parameters (?u - uuv ?w - waypoint ?img - image)
   :precondition (and
      (at ?u ?w)
      (image-at ?img ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?img)
      (not (memory-empty ?u))
   )
)

(:action perform-sonar-scan
   :parameters (?u - uuv ?w - waypoint ?snr - sonar)
   :precondition (and
      (at ?u ?w)
      (sonar-at ?snr ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?snr)
      (not (memory-empty ?u))
   )
)
```

```
(:action transmit-data
   :parameters (?u - uuv ?d - data ?l - location)
   :precondition (and
      (at ?u ?l)
      (has-data ?u ?d)
   )
   :effect (and
      (not (has-data ?u ?d))
      (data-saved ?d ?l)
      (memory-empty ?u)
   )
)

(:action collect-sample
   :parameters (?u - uuv ?w - waypoint ?s - sample)
   :precondition (and
      (at ?u ?w)
      (sample-at ?s ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?s)
      (not (sample-at ?s ?w))
      (not (memory-empty ?u))
   )
)

(:action store-sample
   :parameters (?u - uuv ?s - sample ?ship - ship)
   :precondition (and
      (at ?u ?ship)
      (has-data ?u ?s)
      (can-store-sample ?ship)
   )
   :effect (and
      (not (has-data ?u ?s))
      (sample-stored ?s ?ship)
      (not (can-store-sample ?ship))
      (memory-empty ?u)
   )
)
```

```
   (:action return-to-ship
      :parameters (?u - uuv ?w - waypoint ?s - ship )
      :precondition (and
         (at ?u ?w)
         (connected ?w ?s)
      )
      :effect (and
         (not (at ?u ?w))
         (on-ship ?u ?s)
         (can-deploy ?u)
         (not (uuv-deployed ?u))
      )
   )
)
```

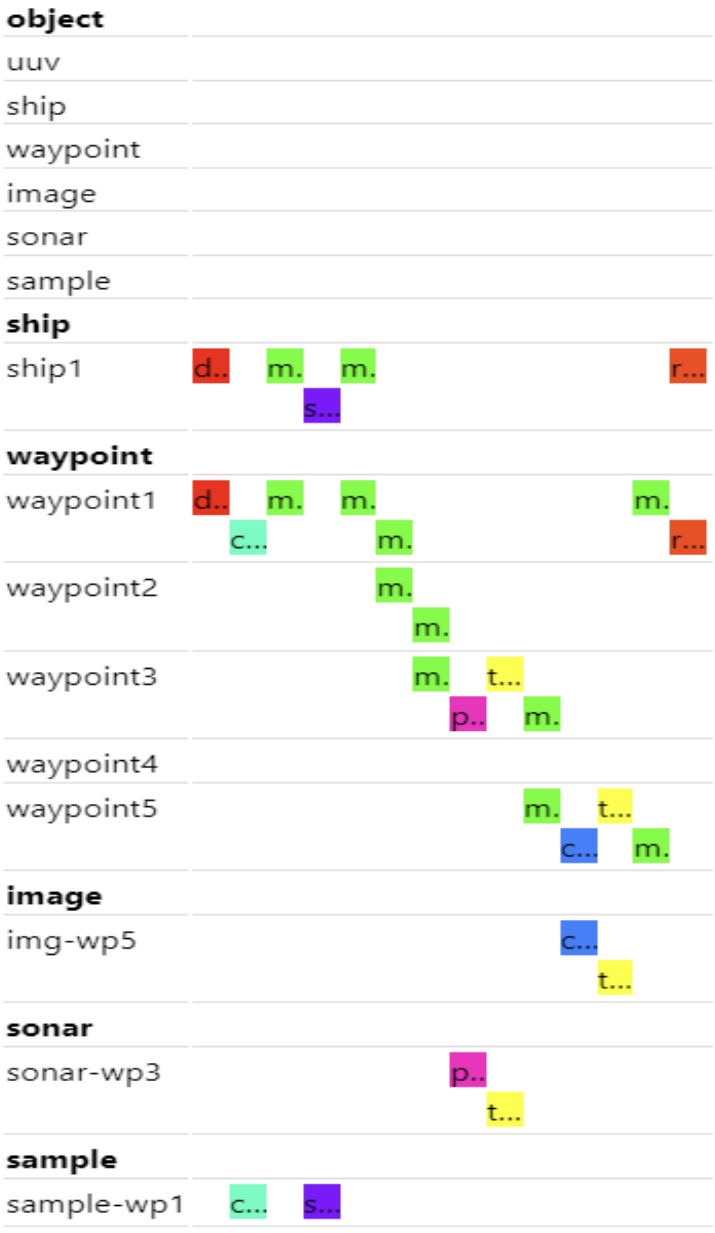
Problem File Problem 2

```
(define (problem windfarm-mission-1)
(:domain windfarm-2)

   (:objects
      uuv1 - uuv
      ship1 - ship
      waypoint1 waypoint2 waypoint3 waypoint4 - waypoint
      img-wp3 - image
      sonar-wp4 - sonar
   )

   (:init
      ; UUV starts on ship1
      (on-ship uuv1 ship1)
      ; Image and sonar data locations
      (image-at img-wp3 waypoint3)
      (sonar-at sonar-wp4 waypoint4)
      ; Ship-waypoint connections
      (connected ship1 waypoint1)
      (connected waypoint1 ship1)
      ; Bidirectional connections between waypoints
      (connected waypoint1 waypoint2)
      (connected waypoint2 waypoint1)
      (connected waypoint3 waypoint4)
      (connected waypoint4 waypoint3)
      ; Unidirectional connections
      (connected waypoint2 waypoint3)
      (connected waypoint4 waypoint1)
      ; Initial UUV state
      (memory-empty uuv1)
      (can-deploy uuv1)
   )
   (:goal (and
      (data-saved img-wp3 waypoint3)
      (data-saved sonar-wp4 waypoint4)
      (on-ship uuv1 ship1)
      )

   )
)
```

Found Plan (output)

(deploy-uuv uuv1 ship1 waypoint1)
(collect-sample uuv1 waypoint1 sample-wp1)
(move-uuv uuv1 waypoint1 ship1)
(store-sample uuv1 sample-wp1 ship1)
(move-uuv uuv1 ship1 waypoint1)
(move-uuv uuv1 waypoint1 waypoint2)
(move-uuv uuv1 waypoint2 waypoint3)
(perform-sonar-scan uuv1 waypoint3 sonar-wp3)
(transmit-data uuv1 sonar-wp3 waypoint3)
(move-uuv uuv1 waypoint3 waypoint5)
(capture-image uuv1 waypoint5 img-wp5)
(transmit-data uuv1 img-wp5 waypoint5)
(move-uuv uuv1 waypoint5 waypoint1)
(return-to-ship uuv1 waypoint1 ship1)

- deploy-uuv uuv1 ship1 waypoint1
  - collect-sample uuv1 waypoint1 sample-wp1
    - move-uuv uuv1 waypoint1 ship1
      - store-sample uuv1 sample-wp1 ship1
        - move-uuv uuv1 ship1 waypoint1
          - move-uuv uuv1 waypoint1 waypoint2
            - move-uuv uuv1 waypoint2 waypoint3
              - perform-sonar-scan uuv1 waypoint3 sonar-wp3
                - transmit-data uuv1 sonar-wp3 waypoint3
                  - move-uuv uuv1 waypoint3 waypoint5
                    - capture-image uuv1 waypoint5 img-wp5
                      - transmit-data uuv1 img-wp5 waypoint5
                        - move-uuv uuv1 waypoint5 waypoint1
                          - return-to-ship uuv1 waypoint1 ship1



Match tree built with 32 nodes.

PDDL problem description loaded:
Domain: WINDFARM-2
Problem: WINDFARM-MISSION-2
#Actions: 32
#Fluents: 34
Goals found: 4
Goals_Edges found: 4
Starting search with 1-BFWS...
--[3 / 0]--
--[3 / 2]--
--[3 / 3]--
--[3 / 4]--
--[3 / 5]--
--[3 / 6]--
--[2 / 0]--
--[2 / 3]--
--[2 / 4]--
--[2 / 5]--
--[2 / 6]--
--[1 / 0]--
--[1 / 4]--
--[0 / 0]--
--[0 / 3]--
Total time: 0.000399
Nodes generated during search: 58
Nodes expanded during search: 28
Plan found with cost: 14
Fast-BFS search completed in 0.000399 secs

Plan found:
0.00000: (DEPLOY-UUV UUV1 SHIP1 WAYPOINT1)
0.00100: (COLLECT-SAMPLE UUV1 WAYPOINT1 SAMPLE-WP1)
0.00200: (MOVE-UUV UUV1 WAYPOINT1 SHIP1)
0.00300: (STORE-SAMPLE UUV1 SAMPLE-WP1 SHIP1)
0.00400: (MOVE-UUV UUV1 SHIP1 WAYPOINT1)
0.00500: (MOVE-UUV UUV1 WAYPOINT1 WAYPOINT2)
0.00600: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT3)
0.00700: (PERFORM-SONAR-SCAN UUV1 WAYPOINT3 SONAR-WP3)
0.00800: (TRANSMIT-DATA UUV1 SONAR-WP3 WAYPOINT3)
0.00900: (MOVE-UUV UUV1 WAYPOINT3 WAYPOINT5)
0.01000: (CAPTURE-IMAGE UUV1 WAYPOINT5 IMG-WP5)
0.01100: (TRANSMIT-DATA UUV1 IMG-WP5 WAYPOINT5)
0.01200: (MOVE-UUV UUV1 WAYPOINT5 WAYPOINT1)
0.01300: (RETURN-TO-SHIP UUV1 WAYPOINT1 SHIP1)
Metric: 0.013000000000000005
Makespan: 0.013000000000000005
States evaluated: undefined
Planner found 1 plan(s) in 3.539secs.

Task 2.2 - Problem 3

Domain File Problem 3

```
(define (domain windfarm-3)
  (:requirements :typing :negative-preconditions)

  (:types
    uuv location data - object
    ship waypoint - location
    image sonar sample - data
  )

  (:predicates
    (at ?u - uuv ?l - location) ; UUV is at a location
    (on-ship ?u-uuv ?s-ship) ; UUV is on a ship
    (connected ?from-location ?to-location) ; Locations are connected
    (can-deploy ?u - uuv) ; UUV can be deployed
    (uuv-deployed ?u - uuv) ; UUV is deployed
    (memory-empty ?u - uuv); UUV memory is empty
    (has-data ?u-uuv ?d-data);  UUV has data
    (data-saved ?d - data ?l-location); Data is saved at a location
    (sample-at ?s - sample ?w-waypoint);    Sample is at a waypoint
    (image-at ?img - image ?w-waypoint);    Image is at a waypoint
    (sonar-at ?snr - sonar ?w - waypoint);  Sonar is at a waypoint
    (sample-stored ?s - sample ?ship - ship);   Sample is stored on a ship

    (can-store-sample ?ship-ship)
    ;; Predicates to link UUVs with their respective ships
    (assigned-to ?u - uuv ?s - ship)        ; UUV is assigned to a specific ship
  )
```

```
(:action deploy-uuv
   :parameters (?u - uuv ?s - ship ?l - location)
   :precondition (and
      (on-ship ?u ?s)
      (connected ?s ?l)
      (can-deploy ?u)
      (assigned-to ?u ?s)      ;; Ensure UUV is assigned to the ship
   )
   :effect (and
      (not (on-ship ?u ?s))
      (at ?u ?l)
      (not (can-deploy ?u))
      (uuv-deployed ?u)
   )
)

(:action move-uuv
   :parameters (?u - uuv ?from -location ?to -location)
   :precondition (and
      (at ?u ?from)
      (connected ?from ?to)
      (uuv-deployed ?u)
   )
   :effect (and
      (not (at ?u ?from))
      (at ?u ?to)
   )
)

(:action capture-image
   :parameters (?u - uuv ?w - waypoint ?img - image)
   :precondition (and
      (at ?u ?w)
      (image-at ?img ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?img)
      (not (memory-empty ?u))
   )
)
```

```
(:action perform-sonar-scan
   :parameters (?u - uuv ?w - waypoint ?snr - sonar)
   :precondition (and
      (at ?u ?w)
      (sonar-at ?snr ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?snr)
      (not (memory-empty ?u))
   )
)

(:action transmit-data
   :parameters (?u - uuv ?d - data ?l - location ?s - ship)
   :precondition (and
      (at ?u ?l)
      (has-data ?u ?d)
      (assigned-to ?u ?s)        ;; Ensure UUV is assigned to the ship
   )
   :effect (and
      (not (has-data ?u ?d))
      (data-saved ?d ?l)
      (memory-empty ?u)
   )
)

(:action collect-sample
   :parameters (?u - uuv ?w - waypoint ?s - sample)
   :precondition (and
      (at ?u ?w)
      (sample-at ?s ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?s)
      (not (sample-at ?s ?w))
      (not (memory-empty ?u))
   )
)
```

```
  (:action store-sample
    :parameters (?u - uuv ?s - sample ?ship - ship)
    :precondition (and
      (at ?u ?ship)
      (has-data ?u ?s)
      (assigned-to ?u ?ship)        ;; Ensure UUV is assigned to the ship
      (can-store-sample ?ship)
    )
    :effect (and
      (not (has-data ?u ?s))
      (sample-stored ?s ?ship)
      (not (can-store-sample ?ship))
      (memory-empty ?u)
    )
  )

  (:action return-to-ship
    :parameters (?u - uuv ?w - waypoint ?s - ship)
    :precondition (and
      (at ?u ?w)
      (connected ?w ?s)
      (uuv-deployed ?u)
      (assigned-to ?u ?s)      ;; Ensure UUV is assigned to the ship
    )
    :effect (and
      (not (at ?u ?w))
      (on-ship ?u ?s)
      (can-deploy ?u)
      (not (uuv-deployed ?u))
    )
  )
)
```

```
(define (problem windfarm-mission-3)
(:domain windfarm-3)

(:objects
   ; Vehicles and personnel
   uuv1 uuv2 - uuv

   ; Locations
   ship1 ship2 - ship
   waypoint1 waypoint2 waypoint3 waypoint4 waypoint5 waypoint6 - waypoint


   ; Mission data
   img-wp2 img-wp3 - image
   sonar-wp4 sonar-wp6 - sonar
   sample-wp1 sample-wp5 - sample
)

(:init
   ; Initial positions
   (at uuv1 waypoint2) ; UUV1 starts deployed at waypoint2
   (on-ship uuv2 ship2) ; UUV2 starts on ship2

   ;; UUV assignments
   (assigned-to uuv1 ship1)
   (assigned-to uuv2 ship2)

   ; Initial UUV states
   (uuv-deployed uuv1)
   (can-deploy uuv2)
   (memory-empty uuv1)
   (memory-empty uuv2)

   ; Ship states
   (can-store-sample ship1)
   (can-store-sample ship2)
   ; Ship to waypoint connections
   (connected ship1 waypoint2)
   (connected waypoint2 ship1)
   (connected ship2 waypoint3)
   (connected waypoint3 ship2)

   ; Waypoint connections
   (connected waypoint1 waypoint2)  ; Waypoint Bi-directional
   (connected waypoint2 waypoint1)

    (connected waypoint2 waypoint3)  ; Waypoint Uni-directional

   (connected waypoint3 waypoint5) ; Waypoint Bi-directional
   (connected waypoint5 waypoint3)
```

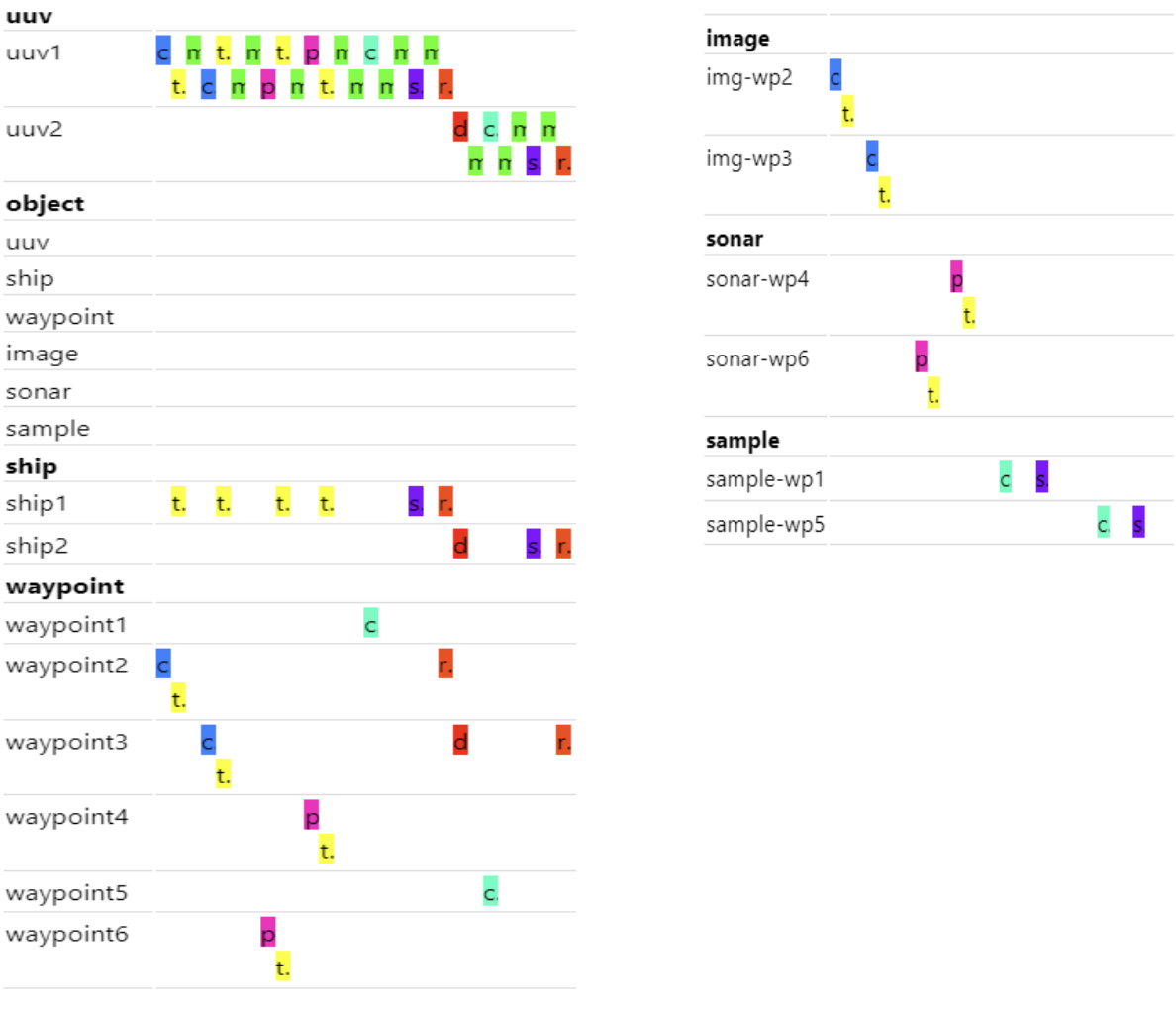
```
        (connected waypoint2 waypoint4) ; Waypoint Bi-directional
        (connected waypoint4 waypoint2)

        (connected waypoint5 waypoint6) ; Waypoint Uni-directional
        (connected waypoint6 waypoint4) ; Waypoint Uni-directional

        ; Sample locations
        (sample-at sample-wp1 waypoint1)
        (sample-at sample-wp5 waypoint5)
        ; Image locations
        (image-at img-wp2 waypoint2)
        (image-at img-wp3 waypoint3)
        ; Sonar locations
        (sonar-at sonar-wp4 waypoint4)
        (sonar-at sonar-wp6 waypoint6)

    )
    (:goal (and
        ;; UUV1-specific tasks
        (data-saved img-wp2 waypoint2)
        (data-saved sonar-wp4 waypoint4)
        (sample-stored sample-wp1 ship1)
        (on-ship uuv1 ship1)

        ;; UUV2-specific tasks
        (data-saved img-wp3 waypoint3)
        (data-saved sonar-wp6 waypoint6)
        (sample-stored sample-wp5 ship2)
        (on-ship uuv2 ship2)
    )
  )
)
```

Found Plan (output)

Raw Result

```
(capture-image uuv1 waypoint2 img-wp2)
(transmit-data uuv1 img-wp2 waypoint2 ship1)
(move-uuv uuv1 waypoint2 waypoint3)
(capture-image uuv1 waypoint3 img-wp3)
(transmit-data uuv1 img-wp3 waypoint3 ship1)
(move-uuv uuv1 waypoint3 waypoint5)
(move-uuv uuv1 waypoint5 waypoint6)
(perform-sonar-scan uuv1 waypoint6 sonar-wp6)
(transmit-data uuv1 sonar-wp6 waypoint6 ship1)
(move-uuv uuv1 waypoint6 waypoint4)
(perform-sonar-scan uuv1 waypoint4 sonar-wp4)
(transmit-data uuv1 sonar-wp4 waypoint4 ship1)
(move-uuv uuv1 waypoint4 waypoint2)
(move-uuv uuv1 waypoint2 waypoint1)
(collect-sample uuv1 waypoint1 sample-wp1)
(move-uuv uuv1 waypoint1 waypoint2)
(move-uuv uuv1 waypoint2 ship1)
(store-sample uuv1 sample-wp1 ship1)
(move-uuv uuv1 ship1 waypoint2)
(return-to-ship uuv1 waypoint2 ship1)
(deploy-uuv uuv2 ship2 waypoint3)
(move-uuv uuv2 waypoint3 waypoint5)
(collect-sample uuv2 waypoint5 sample-wp5)
(move-uuv uuv2 waypoint5 waypoint3)
(move-uuv uuv2 waypoint3 ship2)
(store-sample uuv2 sample-wp5 ship2)
(move-uuv uuv2 ship2 waypoint3)
(return-to-ship uuv2 waypoint3 ship2)
```

- capture-image uuv1 waypoint2 img-wp2
  - transmit-data uuv1 img-wp2 waypoint2 ship1
    - move-uuv uuv1 waypoint2 waypoint3
      - capture-image uuv1 waypoint3 img-wp3
        - transmit-data uuv1 img-wp3 waypoint3 ship1
          - move-uuv uuv1 waypoint3 waypoint5
            - move-uuv uuv1 waypoint5 waypoint6
              - perform-sonar-scan uuv1 waypoint6 sonar-wp6
                - transmit-data uuv1 sonar-wp6 waypoint6 ship1
                  - move-uuv uuv1 waypoint6 waypoint4
                    - perform-sonar-scan uuv1 waypoint4 sonar-wp4
                      - transmit-data uuv1 sonar-wp4 waypoint4 ship1
                        - move-uuv uuv1 waypoint4 waypoint2
                          - move-uuv uuv1 waypoint2 waypoint1
                            - collect-sample uuv1 waypoint1 sample-wp1
                              - move-uuv uuv1 waypoint1 waypoint2
                                - move-uuv uuv1 waypoint2 ship1
                                  - store-sample uuv1 sample-wp1 ship1
                                    - move-uuv uuv1 ship1 waypoint2
                                      - return-to-ship uuv1 waypoint2 ship1
                                        - deploy-uuv uuv2 ship2 waypoint3
                                          - move-uuv uuv2 waypoint3 waypoint5
                                            - collect-sample uuv2 waypoint5 sample-wp5
                                              - move-uuv uuv2 waypoint5 waypoint3
                                                - move-uuv uuv2 waypoint3 ship2
                                                  - store-sample uuv2 sample-wp5 ship2
                                                    - move-uuv uuv2 ship2 waypoint3
                                                      - return-to-ship uuv2 waypoint3 ship2



Match tree built with 142 nodes.

PDDL problem description loaded:
Domain: WINDFARM
Problem: WINDFARM-MISSION-3
#Actions: 142
#Fluents: 92
Goals found: 8
Goals_Edges found: 8
Starting search with 1-BFWS...
--[7 / 0]--
--[6 / 0]--
--[6 / 2]--
--[6 / 3]--
--[5 / 0]--
--[5 / 2]--
--[5 / 3]--
--[5 / 4]--
--[5 / 5]--
--[4 / 0]--
--[4 / 5]--
--[3 / 0]--
--[3 / 4]--
--[2 / 0]--
--[2 / 3]--
--[2 / 4]--
--[2 / 5]--
--[2 / 6]--
--[1 / 0]--
--[1 / 3]--
--[1 / 4]--
--[1 / 5]--
--[1 / 6]--
--[1 / 7]--
--[0 / 0]--
--[0 / 3]--
Total time: 0.001296
Nodes generated during search: 254
Nodes expanded during search: 105
Plan found with cost: 28
Fast-BFS search completed in 0.001296 secs

Plan found:
0.00000: (CAPTURE-IMAGE UUV1 WAYPOINT2 IMG-WP2)
0.00100: (TRANSMIT-DATA UUV1 IMG-WP2 WAYPOINT2 SHIP1)
0.00200: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT3)
0.00300: (CAPTURE-IMAGE UUV1 WAYPOINT3 IMG-WP3)
0.00400: (TRANSMIT-DATA UUV1 IMG-WP3 WAYPOINT3 SHIP1)
0.00500: (MOVE-UUV UUV1 WAYPOINT3 WAYPOINT5)
0.00600: (MOVE-UUV UUV1 WAYPOINT5 WAYPOINT6)
0.00700: (PERFORM-SONAR-SCAN UUV1 WAYPOINT6 SONAR-WP6)
0.00800: (TRANSMIT-DATA UUV1 SONAR-WP6 WAYPOINT6 SHIP1)
0.00900: (MOVE-UUV UUV1 WAYPOINT6 WAYPOINT4)
0.01000: (PERFORM-SONAR-SCAN UUV1 WAYPOINT4 SONAR-WP4)
0.01100: (TRANSMIT-DATA UUV1 SONAR-WP4 WAYPOINT4 SHIP1)
0.01200: (MOVE-UUV UUV1 WAYPOINT4 WAYPOINT2)
0.01300: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT1)
0.01400: (COLLECT-SAMPLE UUV1 WAYPOINT1 SAMPLE-WP1)
0.01500: (MOVE-UUV UUV1 WAYPOINT1 WAYPOINT2)
0.01600: (MOVE-UUV UUV1 WAYPOINT2 SHIP1)
0.01700: (STORE-SAMPLE UUV1 SAMPLE-WP1 SHIP1)
0.01800: (MOVE-UUV UUV1 SHIP1 WAYPOINT2)
0.01900: (RETURN-TO-SHIP UUV1 WAYPOINT2 SHIP1)
0.02000: (DEPLOY-UUV UUV2 SHIP2 WAYPOINT3)
0.02100: (MOVE-UUV UUV2 WAYPOINT3 WAYPOINT5)
0.02200: (COLLECT-SAMPLE UUV2 WAYPOINT5 SAMPLE-WP5)
0.02300: (MOVE-UUV UUV2 WAYPOINT5 WAYPOINT3)
0.02400: (MOVE-UUV UUV2 WAYPOINT3 SHIP2)
0.02500: (STORE-SAMPLE UUV2 SAMPLE-WP5 SHIP2)
0.02600: (MOVE-UUV UUV2 SHIP2 WAYPOINT3)
0.02700: (RETURN-TO-SHIP UUV2 WAYPOINT3 SHIP2)
Metric: 0.027000000000000017
Makespan: 0.027000000000000017
States evaluated: undefined
Planner found 1 plan(s) in 5.199secs.

Task 3.1 - Problem 4

Domain File Problem 4

```
(define (domain windfarm-extended-4)
  (:requirements :typing :negative-preconditions)

  (:types
    uuv engineer - object
    location data - object
    ship waypoint - location
    bay control-center - location
    image sonar sample - data
  )

  (:predicates
    (at ?u - uuv ?l - location)
    (on-ship ?u - uuv ?s - ship)
    (engineer-at ?e - engineer ?loc - location)
    (connected ?from - location ?to - location)
    (can-deploy ?u - uuv)
    (uuv-deployed ?u - uuv)
    (memory-empty ?u - uuv)
    (has-data ?u - uuv ?d - data)
    (data-saved ?d - data ?l - location)
    (sample-at ?s - sample ?w - waypoint)
    (image-at ?img - image ?w - waypoint)
    (sonar-at ?snr - sonar ?w - waypoint)
    (sample-stored ?s - sample ?ship - ship)
    (can-store-sample ?ship - ship)

    ;; Predicates to link UUVs with their respective ships, engineers, and facilities
    (assigned-to ?u - uuv ?s - ship)        ; UUV is assigned to a specific ship
    (engineer-for-ship ?e - engineer ?s - ship) ; Engineer is for a specific ship
    (bay-for-ship ?b - bay ?s - ship)       ; Bay is for a specific ship
    (control-for-ship ?c - control-center ?s - ship) ; Control center for a specific ship
  )
```

```
(:action move-engineer
   :parameters (?e - engineer ?from ?to - location)
   :precondition (and
      (engineer-at ?e ?from)
      (connected ?from ?to)
   )
   :effect (and
      (not (engineer-at ?e ?from))
      (engineer-at ?e ?to)
   )
)

(:action deploy-uuv
   :parameters (?u - uuv ?s - ship ?l - location ?e - engineer ?b - bay)
   :precondition (and
      (on-ship ?u ?s)
      (connected ?s ?l)
      (can-deploy ?u)
      (engineer-at ?e ?b)
      (assigned-to ?u ?s)      ;; Ensure UUV is assigned to the ship
      (engineer-for-ship ?e ?s) ;; Ensure the engineer is for this ship
      (bay-for-ship ?b ?s)     ;; Ensure the bay is for this ship
   )
   :effect (and
      (not (on-ship ?u ?s))
      (at ?u ?l)
      (not (can-deploy ?u))
      (uuv-deployed ?u)
   )
)
```

```
(:action move-uuv
   :parameters (?u - uuv ?from ?to - location)
   :precondition (and
      (at ?u ?from)
      (connected ?from ?to)
      (uuv-deployed ?u)
   )
   :effect (and
      (not (at ?u ?from))
      (at ?u ?to)
   )
)

(:action capture-image
   :parameters (?u - uuv ?w - waypoint ?img - image)
   :precondition (and
      (at ?u ?w)
      (image-at ?img ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?img)
      (not (memory-empty ?u))
   )
)

(:action perform-sonar-scan
   :parameters (?u - uuv ?w - waypoint ?snr - sonar)
   :precondition (and
      (at ?u ?w)
      (sonar-at ?snr ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?snr)
      (not (memory-empty ?u))
   )
)
```

```
(:action transmit-data
   :parameters (?u - uuv ?d - data ?l - location ?e - engineer ?c - control-center ?s - ship)
   :precondition (and
      (at ?u ?l)
      (has-data ?u ?d)
      (engineer-at ?e ?c)
      (assigned-to ?u ?s)         ;; Ensure UUV is assigned to the ship
      (engineer-for-ship ?e ?s)    ;; Ensure the engineer is for this ship
      (control-for-ship ?c ?s)     ;; Ensure the control center is for this ship
   )
   :effect (and
      (not (has-data ?u ?d))
      (data-saved ?d ?l)
      (memory-empty ?u)
   )
)

(:action collect-sample
   :parameters (?u - uuv ?w - waypoint ?s - sample)
   :precondition (and
      (at ?u ?w)
      (sample-at ?s ?w)
      (memory-empty ?u)
   )
   :effect (and
      (has-data ?u ?s)
      (not (sample-at ?s ?w))
      (not (memory-empty ?u))
   )
)
```

```
  (:action store-sample
    :parameters (?u - uuv ?s - sample ?ship - ship)
    :precondition (and
      (at ?u ?ship)
      (has-data ?u ?s)
      (can-store-sample ?ship)
    )
    :effect (and
      (not (has-data ?u ?s))
      (sample-stored ?s ?ship)
      (not (can-store-sample ?ship))
      (memory-empty ?u)
    )
  )

  (:action return-to-ship
    :parameters (?u - uuv ?w - waypoint ?s - ship ?e - engineer ?b - bay)
    :precondition (and
      (at ?u ?w)
      (connected ?w ?s)
      (engineer-at ?e ?b)
      (uuv-deployed ?u)
      (assigned-to ?u ?s)      ;; Ensure UUV is assigned to the ship
      (engineer-for-ship ?e ?s) ;; Ensure the engineer is for this ship
      (bay-for-ship ?b ?s)     ;; Ensure the bay is for this ship
    )
    :effect (and
      (not (at ?u ?w))
      (on-ship ?u ?s)
      (can-deploy ?u)
      (not (uuv-deployed ?u))
    )
  )
)
```

```
(define (problem windfarm-mission-4)
(:domain windfarm-extended-4)

(:objects
   ; Vehicles and personnel
   uuv1 uuv2 - uuv
   engineer1 engineer2 - engineer

   ; Locations
   ship1 ship2 - ship
   waypoint1 waypoint2 waypoint3 waypoint4 waypoint5 waypoint6 - waypoint
   bay1 bay2 - bay ; bay1 for ship1, bay2 for ship2
   control1 control2 - control-center ; control1 for ship1, control2 for ship2

   ; Mission data
   img-wp2 img-wp3 - image
   sonar-wp4 sonar-wp6 - sonar
   sample-wp1 sample-wp5 - sample
)

(:init
   ; Initial positions
   (at uuv1 waypoint2) ; UUV1 starts deployed at waypoint2
   (on-ship uuv2 ship2) ; UUV2 starts on ship2
   (engineer-at engineer1 bay1) ;Engineer1 starts at ship1's bay
   (engineer-at engineer2 bay2) ; Engineer2 starts at ship2's bay

   ;; UUV and engineer assignments
   (assigned-to uuv1 ship1)
   (assigned-to uuv2 ship2)
   (engineer-for-ship engineer1 ship1)
   (engineer-for-ship engineer2 ship2)
   (bay-for-ship bay1 ship1)
   (bay-for-ship bay2 ship2)
   (control-for-ship control1 ship1)
   (control-for-ship control2 ship2)
```

; Initial UUV states
(uuv-deployed uuv1)
(can-deploy uuv2)
(memory-empty uuv1)
(memory-empty uuv2)

; Ship states
(can-store-sample ship1)
(can-store-sample ship2)

; Ship1 facility connections
(connected bay1 control1)
(connected control1 bay1)

; Ship2 facility connections
(connected bay2 control2)
(connected control2 bay2)

; Ship to waypoint connections
(connected ship1 waypoint1)
(connected waypoint1 ship1)
(connected ship2 waypoint3)
(connected waypoint3 ship2)

; Waypoint connections (bidirectional)
(connected waypoint1 waypoint2)
(connected waypoint2 waypoint1)
(connected waypoint2 waypoint3)

(connected waypoint3 waypoint5)
(connected waypoint5 waypoint3)

(connected waypoint2 waypoint4)
(connected waypoint4 waypoint2)

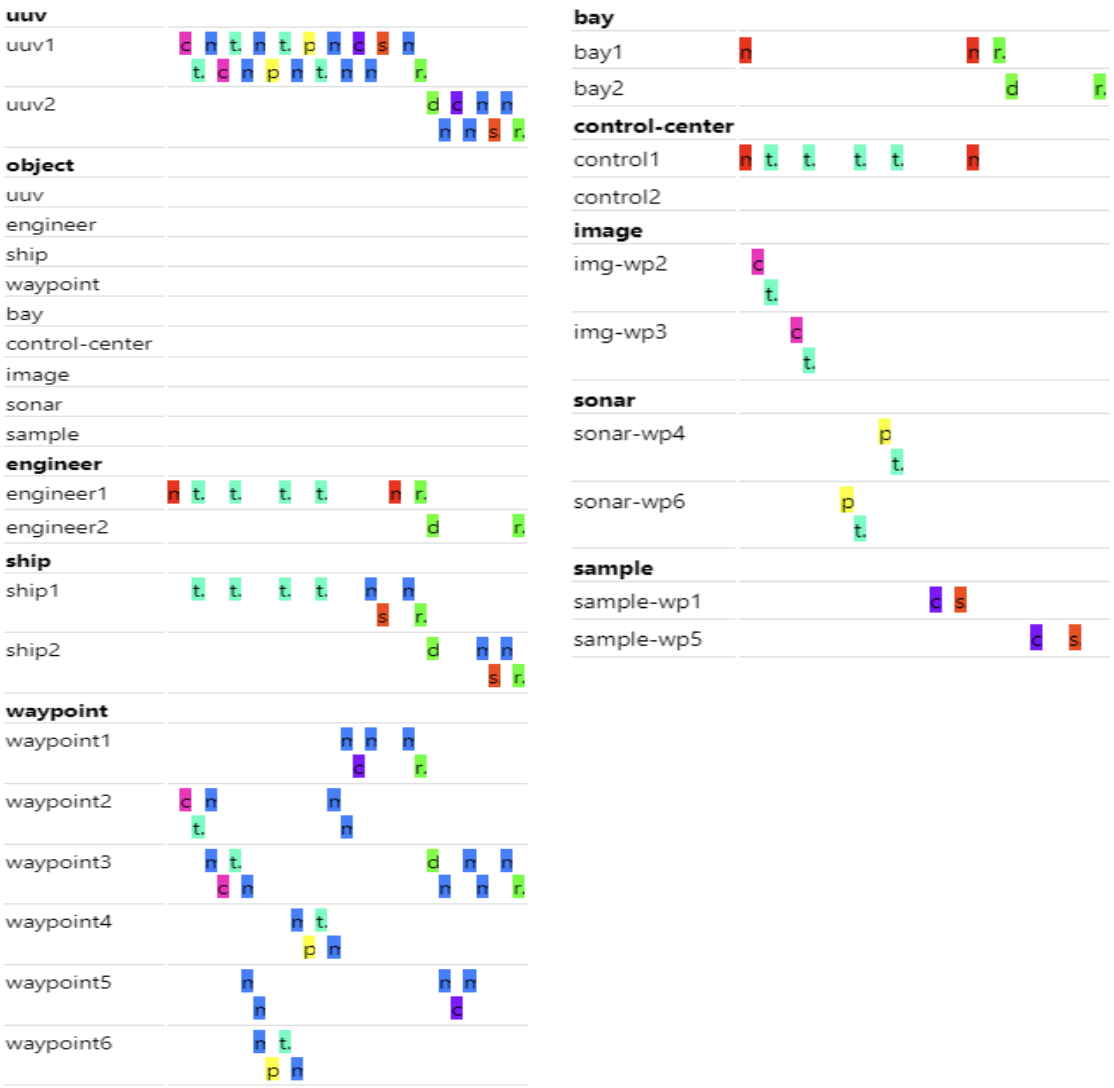
(connected waypoint5 waypoint6)
(connected waypoint6 waypoint4)

```
    ; Sample locations
    (sample-at sample-wp1 waypoint1)
    (sample-at sample-wp5 waypoint5)

    ; Image locations
    (image-at img-wp2 waypoint2)
    (image-at img-wp3 waypoint3)

    ; Sonar locations
    (sonar-at sonar-wp4 waypoint4)
    (sonar-at sonar-wp6 waypoint6)

  )

 (:goal (and
    ; Image collection goals
    (data-saved img-wp2 waypoint2)
    (data-saved img-wp3 waypoint3)

    ; Sonar scan goals
    (data-saved sonar-wp4 waypoint4)
    (data-saved sonar-wp6 waypoint6)

    ; Sample collection goals
    (sample-stored sample-wp1 ship1)
    (sample-stored sample-wp5 ship2)

    ; UUV1 and UUV2 return to their respective ships
    (on-ship uuv1 ship1)
    (on-ship uuv2 ship2)
  )
 )
)
```

Found Plan (output)

(move-engineer engineer1 bay1 control1)
(capture-image uuv1 waypoint2 img-wp2)
(transmit-data uuv1 img-wp2 waypoint2 engineer1 control1 ship1)
(move-uuv uuv1 waypoint2 waypoint3)
(capture-image uuv1 waypoint3 img-wp3)
(transmit-data uuv1 img-wp3 waypoint3 engineer1 control1 ship1)
(move-uuv uuv1 waypoint3 waypoint5)
(move-uuv uuv1 waypoint5 waypoint6)
(perform-sonar-scan uuv1 waypoint6 sonar-wp6)
(transmit-data uuv1 sonar-wp6 waypoint6 engineer1 control1 ship1)
(move-uuv uuv1 waypoint6 waypoint4)
(perform-sonar-scan uuv1 waypoint4 sonar-wp4)
(transmit-data uuv1 sonar-wp4 waypoint4 engineer1 control1 ship1)
(move-uuv uuv1 waypoint4 waypoint2)
(move-uuv uuv1 waypoint2 waypoint1)
(collect-sample uuv1 waypoint1 sample-wp1)
(move-uuv uuv1 waypoint1 ship1)
(store-sample uuv1 sample-wp1 ship1)
(move-engineer engineer1 control1 bay1)
(move-uuv uuv1 ship1 waypoint1)
(return-to-ship uuv1 waypoint1 ship1 engineer1 bay1)
(deploy-uuv uuv2 ship2 waypoint3 engineer2 bay2)
(move-uuv uuv2 waypoint3 waypoint5)
(collect-sample uuv2 waypoint5 sample-wp5)
(move-uuv uuv2 waypoint5 waypoint3)
(move-uuv uuv2 waypoint3 ship2)
(store-sample uuv2 sample-wp5 ship2)
(move-uuv uuv2 ship2 waypoint3)
(return-to-ship uuv2 waypoint3 ship2 engineer2 bay2)

- move-engineer engineer1 bay1 control1
  - capture-image uuv1 waypoint2 img-wp2
    - transmit-data uuv1 img-wp2 waypoint2 engineer1 control1 ship1
      - move-uuv uuv1 waypoint2 waypoint3
        - capture-image uuv1 waypoint3 img-wp3
          - transmit-data uuv1 img-wp3 waypoint3 engineer1 control1 ship1
            - move-uuv uuv1 waypoint3 waypoint5
              - move-uuv uuv1 waypoint5 waypoint6
                - perform-sonar-scan uuv1 waypoint6 sonar-wp6
                  - transmit-data uuv1 sonar-wp6 waypoint6 engineer1 control1 ship1
                    - move-uuv uuv1 waypoint6 waypoint4
                      - perform-sonar-scan uuv1 waypoint4 sonar-wp4
                        - transmit-data uuv1 sonar-wp4 waypoint4 engineer1 control1 ship1
                          - move-uuv uuv1 waypoint4 waypoint2
                            - move-uuv uuv1 waypoint2 waypoint1
                              - collect-sample uuv1 waypoint1 sample-wp1
                                - move-uuv uuv1 waypoint1 ship1
                                  - store-sample uuv1 sample-wp1 ship1
                                    - move-engineer engineer1 control1 bay1
                                      - move-uuv uuv1 ship1 waypoint1
                                        - return-to-ship uuv1 waypoint1 ship1 engineer1 bay1
                                          - deploy-uuv uuv2 ship2 waypoint3 engineer2 bay2
                                            - move-uuv uuv2 waypoint3 waypoint5
                                              - collect-sample uuv2 waypoint5 sample-wp5
                                                - move-uuv uuv2 waypoint5 waypoint3
                                                  - move-uuv uuv2 waypoint3 ship2
                                                    - store-sample uuv2 sample-wp5 ship2
                                                      - move-uuv uuv2 ship2 waypoint3
                                                        - return-to-ship uuv2 waypoint3 ship2 engineer2 bay2



Match tree built with 150 nodes.

PDDL problem description loaded:
Domain: WINDFARM-EXTENDED-4
Problem: WINDFARM-MISSION-4
#Actions: 150
#Fluents: 96
Goals found: 8
Goal Edges found: 8
Starting search with 1-BFWS...
--[7 / 0]--
--[7 / 1]--
--[7 / 2]--
--[6 / 0]--
--[6 / 3]--
--[6 / 4]--
--[5 / 0]--
--[5 / 4]--
--[4 / 0]--
--[4 / 5]--
--[3 / 0]--
--[3 / 4]--
--[2 / 0]--
--[2 / 5]--
--[2 / 6]--
--[1 / 0]--
--[1 / 4]--
--[1 / 5]--
--[1 / 7]--
--[0 / 0]--
--[0 / 3]--
Total time: 0.001079
Nodes generated during search: 315
Nodes expanded during search: 148
Plan found with cost: 29
Fast-BFS search completed in 0.001079 secs

Plan found:
0.00000: (MOVE-ENGINEER ENGINEER1 BAY1 CONTROL1)
0.00100: (CAPTURE-IMAGE UUV1 WAYPOINT2 IMG-WP2)
0.00200: (TRANSMIT-DATA UUV1 IMG-WP2 WAYPOINT2 ENGINEER1 CONTROL1 SHIP1)
0.00300: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT3)
0.00400: (CAPTURE-IMAGE UUV1 WAYPOINT3 IMG-WP3)
0.00500: (TRANSMIT-DATA UUV1 IMG-WP3 WAYPOINT3 ENGINEER1 CONTROL1 SHIP1)
0.00600: (MOVE-UUV UUV1 WAYPOINT3 WAYPOINT5)
0.00700: (MOVE-UUV UUV1 WAYPOINT5 WAYPOINT6)
0.00800: (PERFORM-SONAR-SCAN UUV1 WAYPOINT6 SONAR-WP6)
0.00900: (TRANSMIT-DATA UUV1 SONAR-WP6 WAYPOINT6 ENGINEER1 CONTROL1 SHIP1)
0.01000: (MOVE-UUV UUV1 WAYPOINT6 WAYPOINT4)
0.01100: (PERFORM-SONAR-SCAN UUV1 WAYPOINT4 SONAR-WP4)
0.01200: (TRANSMIT-DATA UUV1 SONAR-WP4 WAYPOINT4 ENGINEER1 CONTROL1 SHIP1)
0.01300: (MOVE-UUV UUV1 WAYPOINT4 WAYPOINT2)
0.01400: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT1)
0.01500: (COLLECT-SAMPLE UUV1 WAYPOINT1 SAMPLE-WP1)
0.01600: (MOVE-UUV UUV1 WAYPOINT1 SHIP1)
0.01700: (STORE-SAMPLE UUV1 SAMPLE-WP1 SHIP1)
0.01800: (MOVE-ENGINEER ENGINEER1 CONTROL1 BAY1)
0.01900: (MOVE-UUV UUV1 SHIP1 WAYPOINT1)
0.02000: (RETURN-TO-SHIP UUV1 WAYPOINT1 SHIP1 ENGINEER1 BAY1)
0.02100: (DEPLOY-UUV UUV2 SHIP2 WAYPOINT3 ENGINEER2 BAY2)
0.02200: (MOVE-UUV UUV2 WAYPOINT3 WAYPOINT5)
0.02300: (COLLECT-SAMPLE UUV2 WAYPOINT5 SAMPLE-WP5)
0.02400: (MOVE-UUV UUV2 WAYPOINT5 WAYPOINT3)
0.02500: (MOVE-UUV UUV2 WAYPOINT3 SHIP2)
0.02600: (STORE-SAMPLE UUV2 SAMPLE-WP5 SHIP2)
0.02700: (MOVE-UUV UUV2 SHIP2 WAYPOINT3)
0.02800: (RETURN-TO-SHIP UUV2 WAYPOINT3 SHIP2 ENGINEER2 BAY2)
Metric: 0.028000000000000018
Makespan: 0.028000000000000018
States evaluated: undefined
Planner found 1 plan(s) in 3.521secs.

Task 3.2 - Problem 5

Domain File Problem 5

```
(define (domain windfarm-extended-5)
  (:requirements :typing :negative-preconditions :conditional-effects)

  (:types
    uuv engineer - object
    location data - object
    ship waypoint - location
    bay control-center - location
    image sonar sample - data
  )

  (:predicates
    (at ?u - uuv ?l - location)
    (on-ship ?u - uuv ?s - ship)
    (engineer-at ?e - engineer ?loc - location)
    (connected ?from - location ?to - location)
    (can-deploy ?u - uuv)
    (uuv-deployed ?u - uuv)
    (memory-empty ?u - uuv)
    (has-data ?u - uuv ?d - data)
    (data-saved ?d - data ?l - location)
    (sample-at ?s - sample ?w - waypoint)
    (image-at ?img - image ?w - waypoint)
    (sonar-at ?snr - sonar ?w - waypoint)
    (sample-stored ?s - sample ?ship - ship)
    (can-store-sample ?ship - ship)

    ;; Algae-related predicates
    (algae-at ?w - location)  ; Indicates algae presence at a waypoint
    (uuv-stuck-at ?u - uuv ?w -location)     ; UUV is stuck at a waypoint due to algae
    (can-move ?u - uuv)      ; UUV can move

    ;; Predicates linking UUVs with ships, engineers, and facilities
    (assigned-to ?u - uuv ?s - ship)
    (engineer-for-ship ?e - engineer ?s - ship)
    (bay-for-ship ?b - bay ?s - ship)
    (control-for-ship ?c - control-center ?s - ship)
  )
```

```
(:action deploy-uuv
    :parameters (?u - uuv ?s - ship ?l - location ?e - engineer ?b - bay)
    :precondition (and
      (on-ship ?u ?s)
      (connected ?s ?l)
      (can-deploy ?u)
      (engineer-at ?e ?b)
      (assigned-to ?u ?s)
      (engineer-for-ship ?e ?s)
      (bay-for-ship ?b ?s)
    )
    :effect (and
      (not (on-ship ?u ?s))
      (at ?u ?l)
      (not (can-deploy ?u))
      (uuv-deployed ?u)
      (can-move ?u)   ; UUV can move after deployment
    )
  )

  (:action move-uuv
    :parameters (?u - uuv ?from ?to - location)
    :precondition (and
      (at ?u ?from)
      (connected ?from ?to)
      (uuv-deployed ?u)
      (can-move ?u)   ; UUV can only move if it's not stuck
    )
    :effect (and
      (not (at ?u ?from))
      (at ?u ?to)
      ;; Check if the destination has algae
      (when (algae-at ?to)
        (and
          (uuv-stuck-at ?u ?to)   ; UUV gets stuck at the waypoint
          (not (can-move ?u))     ; UUV can no longer move
        )
      )
    )
  )
```

```
(:action unstuck-uuv
    :parameters (?e - engineer ?u - uuv ?c - control-center ?s - ship ?w - waypoint)
    :precondition (and
      (uuv-stuck-at ?u ?w)
      (engineer-at ?e ?c)
      (engineer-for-ship ?e ?s)   ; Ensure the engineer is for this ship
      (control-for-ship ?c ?s)    ; Ensure the control center is for this ship
      (assigned-to ?u ?s)
    )
    :effect (and
      (not (uuv-stuck-at ?u ?w))    ; UUV is no longer stuck
      (can-move ?u)         ; UUV can move again
    )
  )

  (:action move-engineer
    :parameters (?e - engineer ?from ?to - location)
    :precondition (and
      (engineer-at ?e ?from)
      (connected ?from ?to)
    )
    :effect (and
      (not (engineer-at ?e ?from))
      (engineer-at ?e ?to)
    )
  )

  (:action capture-image
    :parameters (?u - uuv ?w - waypoint ?img - image)
    :precondition (and
      (at ?u ?w)
      (image-at ?img ?w)
      (memory-empty ?u)
    )
    :effect (and
      (has-data ?u ?img)
      (not (memory-empty ?u))
    )
  )
```

```
(:action perform-sonar-scan
    :parameters (?u - uuv ?w - waypoint ?snr - sonar)
    :precondition (and
      (at ?u ?w)
      (sonar-at ?snr ?w)
      (memory-empty ?u)
    )
    :effect (and
      (has-data ?u ?snr)
      (not (memory-empty ?u))
    )
  )
(:action transmit-data
    :parameters (?u - uuv ?d - data ?l - location ?e - engineer ?c - control-center ?s - ship)
    :precondition (and
      (at ?u ?l)
      (has-data ?u ?d)
      (engineer-at ?e ?c)
      (assigned-to ?u ?s)        ;; Ensure UUV is assigned to the ship
      (engineer-for-ship ?e ?s)   ;; Ensure the engineer is for this ship
      (control-for-ship ?c ?s)    ;; Ensure the control center is for this ship
    )
    :effect (and
      (not (has-data ?u ?d))
      (data-saved ?d ?l)
      (memory-empty ?u)
    )
  )

  (:action collect-sample
    :parameters (?u - uuv ?w - waypoint ?s - sample)
    :precondition (and
      (at ?u ?w)
      (sample-at ?s ?w)
      (memory-empty ?u)
    )
    :effect (and
      (has-data ?u ?s)
      (not (sample-at ?s ?w))
      (not (memory-empty ?u))
    )
  )
```

```
(:action store-sample
      :parameters (?u - uuv ?s - sample ?ship - ship)
      :precondition (and
         (at ?u ?ship)
         (has-data ?u ?s)
         (can-store-sample ?ship)
      )
      :effect (and
         (not (has-data ?u ?s))
         (sample-stored ?s ?ship)
         (not (can-store-sample ?ship))
         (memory-empty ?u)
      )
   )
(:action return-to-ship
      :parameters (?u - uuv ?w - waypoint ?s - ship ?e - engineer ?b - bay)
      :precondition (and
         (at ?u ?w)
         (connected ?w ?s)
         (engineer-at ?e ?b)
         (uuv-deployed ?u)
         (can-move ?u)
         (assigned-to ?u ?s)      ;; Ensure UUV is assigned to the ship
         (engineer-for-ship ?e ?s) ;; Ensure the engineer is for this ship
         (bay-for-ship ?b ?s)     ;; Ensure the bay is for this ship
      )
      :effect (and
         (not (at ?u ?w))
         (on-ship ?u ?s)
         (can-deploy ?u)
         (not (uuv-deployed ?u))
      )
   )
)
```

```
(define (problem windfarm-mission-5)
(:domain windfarm-extended-5)

(:objects
   ; Vehicles and personnel
   uuv1 uuv2 - uuv
   engineer1 engineer2 - engineer

   ; Locations
   ship1 ship2 - ship
   waypoint1 waypoint2 waypoint3 waypoint4 waypoint5 waypoint6 - waypoint
   bay1 bay2 - bay ; bay1 for ship1, bay2 for ship2
   control1 control2 - control-center ; control1 for ship1, control2 for ship2

   ; Mission data
   img-wp2 img-wp3 - image
   sonar-wp4 sonar-wp6 - sonar
   sample-wp1 sample-wp5 - sample
)

(:init
   ; Initial positions
   (at uuv1 waypoint2) ; UUV1 starts deployed at waypoint2
   (uuv-stuck-at uuv1 waypoint2) ; UUV1 is stuck at waypoint2
   (on-ship uuv2 ship2) ; UUV2 starts on ship2
   (engineer-at engineer1 bay1) ;Engineer1 starts at ship1's bay
   (engineer-at engineer2 bay2) ; Engineer2 starts at ship2's bay

   ;; UUV and engineer assignments
   (assigned-to uuv1 ship1)
   (assigned-to uuv2 ship2)
   (engineer-for-ship engineer1 ship1)
   (engineer-for-ship engineer2 ship2)
   (bay-for-ship bay1 ship1)
   (bay-for-ship bay2 ship2)
   (control-for-ship control1 ship1)
   (control-for-ship control2 ship2)
```

```
; Initial UUV states
    (uuv-deployed uuv1)
    (can-deploy uuv2)
    (memory-empty uuv1)
    (memory-empty uuv2)

    ; Ship states
    (can-store-sample ship1)
    (can-store-sample ship2)

    ; Ship1 facility connections
    (connected bay1 control1)
    (connected control1 bay1)


    ; Ship2 facility connections
    (connected bay2 control2)
    (connected control2 bay2)

    ; Ship to waypoint connections
    (connected ship1 waypoint2)
    (connected waypoint2 ship1)
    (connected ship2 waypoint3)
    (connected waypoint3 ship2)

    ; Waypoint connections (bidirectional)
    (connected waypoint1 waypoint2)
    (connected waypoint2 waypoint1)
    (connected waypoint2 waypoint3)

    (connected waypoint3 waypoint5)
    (connected waypoint5 waypoint3)

    (connected waypoint2 waypoint4)
    (connected waypoint4 waypoint2)

    (connected waypoint5 waypoint6)
    (connected waypoint6 waypoint4)
```

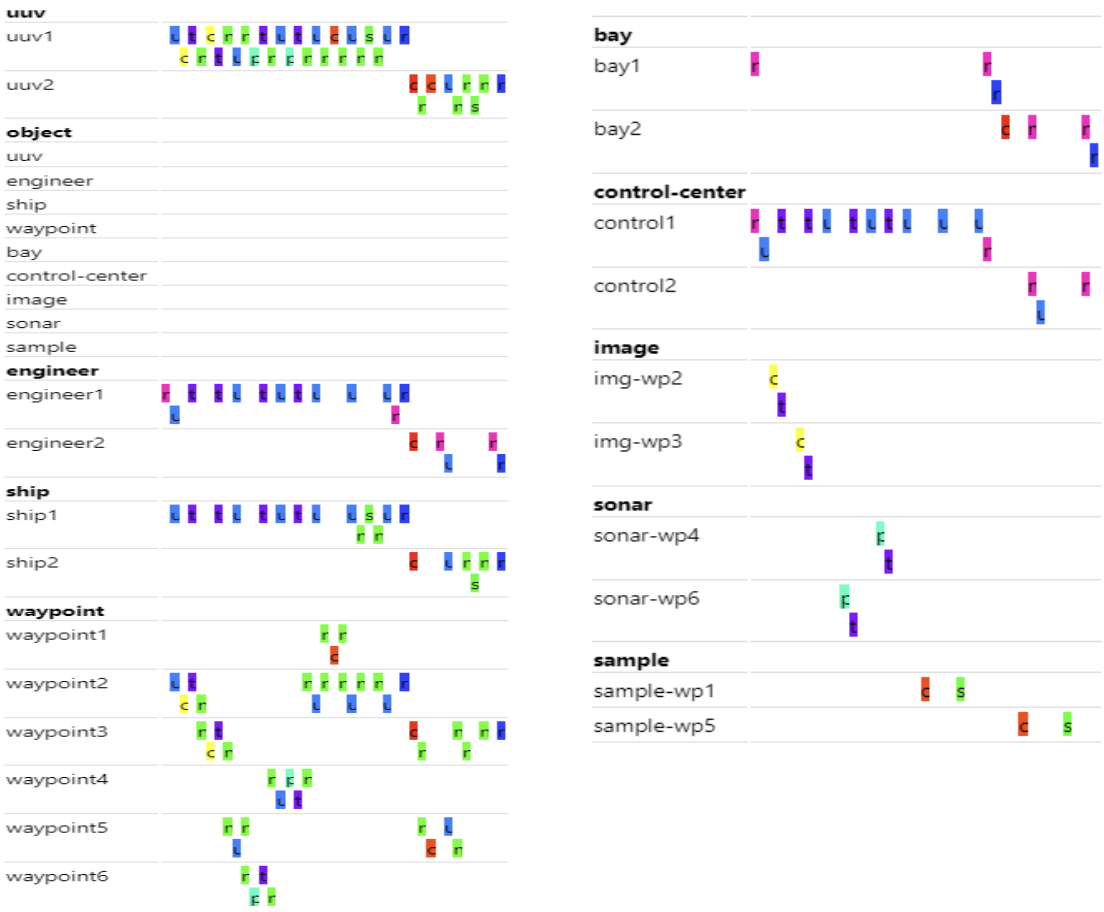
```
; Sample locations
      (sample-at sample-wp1 waypoint1)
      (sample-at sample-wp5 waypoint5)

      ; Image locations
      (image-at img-wp2 waypoint2)
      (image-at img-wp3 waypoint3)

      ; Sonar locations
      (sonar-at sonar-wp4 waypoint4)
      (sonar-at sonar-wp6 waypoint6)

      ; Algae locations
      (algae-at waypoint2)
      (algae-at waypoint4)
      (algae-at waypoint5)

   )

   (:goal (and
      ; Image collection goals
      (data-saved img-wp2 waypoint2)
      (data-saved img-wp3 waypoint3)

      ; Sonar scan goals
      (data-saved sonar-wp4 waypoint4)
      (data-saved sonar-wp6 waypoint6)

      ; Sample collection goals
      (sample-stored sample-wp1 ship1)
      (sample-stored sample-wp5 ship2)

      ; UUV1 and UUV2 return to their respective ships
      (on-ship uuv1 ship1)
      (on-ship uuv2 ship2)
   )
  )
)
```

Found Plan (output)

Raw Result

```
(move-engineer engineer1 bay1 control1)
(unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2)
(capture-image uuv1 waypoint2 img-wp2)
(transmit-data uuv1 img-wp2 waypoint2 engineer1 control1 ship1)
(move-uuv uuv1 waypoint2 waypoint3)
(capture-image uuv1 waypoint3 img-wp3)
(transmit-data uuv1 img-wp3 waypoint3 engineer1 control1 ship1)
(move-uuv uuv1 waypoint3 waypoint5)
(unstuck-uuv engineer1 uuv1 control1 ship1 waypoint5)
(move-uuv uuv1 waypoint5 waypoint6)
(perform-sonar-scan uuv1 waypoint6 sonar-wp6)
(transmit-data uuv1 sonar-wp6 waypoint6 engineer1 control1 ship1)
(move-uuv uuv1 waypoint6 waypoint4)
(unstuck-uuv engineer1 uuv1 control1 ship1 waypoint4)
(perform-sonar-scan uuv1 waypoint4 sonar-wp4)
(transmit-data uuv1 sonar-wp4 waypoint4 engineer1 control1 ship1)
(move-uuv uuv1 waypoint4 waypoint2)
(unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2)
(move-uuv uuv1 waypoint2 waypoint1)
(collect-sample uuv1 waypoint1 sample-wp1)
(move-uuv uuv1 waypoint1 waypoint2)
(unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2)
(move-uuv uuv1 waypoint2 ship1)
(store-sample uuv1 sample-wp1 ship1)
(move-uuv uuv1 ship1 waypoint2)
(unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2)
(move-engineer engineer1 control1 bay1)
(return-to-ship uuv1 waypoint2 ship1 engineer1 bay1)
(deploy-uuv uuv2 ship2 waypoint3 engineer2 bay2)
(move-uuv uuv2 waypoint3 waypoint5)
(collect-sample uuv2 waypoint5 sample-wp5)
(move-engineer engineer2 bay2 control2)
(unstuck-uuv engineer2 uuv2 control2 ship2 waypoint5)
(move-uuv uuv2 waypoint5 waypoint3)
(move-uuv uuv2 waypoint3 ship2)
(store-sample uuv2 sample-wp5 ship2)
(move-uuv uuv2 ship2 waypoint3)
(move-engineer engineer2 control2 bay2)
(return-to-ship uuv2 waypoint3 ship2 engineer2 bay2)
```

move-engineer engineer1 bay1 control1
unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2
capture-image uuv1 waypoint2 img-wp2
transmit-data uuv1 img-wp2 waypoint2 engineer1 control1 ship1
move-uuv uuv1 waypoint2 waypoint3
capture-image uuv1 waypoint3 img-wp3
transmit-data uuv1 img-wp3 waypoint3 engineer1 control1 ship1
move-uuv uuv1 waypoint3 waypoint5
unstuck-uuv engineer1 uuv1 control1 ship1 waypoint5
move-uuv uuv1 waypoint5 waypoint6
perform-sonar-scan uuv1 waypoint6 sonar-wp6
transmit-data uuv1 sonar-wp6 waypoint6 engineer1 control1 ship1
move-uuv uuv1 waypoint6 waypoint4
unstuck-uuv engineer1 uuv1 control1 ship1 waypoint4
perform-sonar-scan uuv1 waypoint4 sonar-wp4
transmit-data uuv1 sonar-wp4 waypoint4 engineer1 control1 ship1
move-uuv uuv1 waypoint4 waypoint2
unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2
move-uuv uuv1 waypoint2 waypoint1
collect-sample uuv1 waypoint1 sample-wp1
move-uuv uuv1 waypoint1 waypoint2
unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2
move-uuv uuv1 waypoint2 ship1
store-sample uuv1 sample-wp1 ship1
move-uuv uuv1 ship1 waypoint2
unstuck-uuv engineer1 uuv1 control1 ship1 waypoint2
move-engineer engineer1 control1 bay1
return-to-ship uuv1 waypoint2 ship1 engineer1 bay1
deploy-uuv uuv2 ship2 waypoint3 engineer2 bay2
move-uuv uuv2 waypoint3 waypoint5
collect-sample uuv2 waypoint5 sample-wp5
move-engineer engineer2 bay2 control2
unstuck-uuv engineer2 uuv2 control2 ship2 waypoint5
move-uuv uuv2 waypoint5 waypoint3
move-uuv uuv2 waypoint3 ship2
store-sample uuv2 sample-wp5 ship2
move-uuv uuv2 ship2 waypoint3
move-engineer engineer2 control2 bay2
return-to-ship uuv2 waypoint3 ship2 engineer2 bay2



Match tree built with 162 nodes.

PDDL problem description loaded:
Domain: WINDFARM-EXTENDED-5
Problem: WINDFARM-MISSION-5
#Actions: 162
#Fluents: 114
Goals found: 8
Goals_Edges found: 8
Starting search with 1-BFWS...
--[7 / 0]--
--[7 / 1]--
--[7 / 3]--
--[7 / 4]--
--[7 / 5]--
--[6 / 0]--
--[6 / 4]--
--[6 / 5]--
--[6 / 6]--
--[5 / 0]--
--[5 / 4]--
--[5 / 5]--
--[5 / 6]--
--[5 / 7]--
--[4 / 0]--
--[4 / 6]--
--[4 / 7]--
--[3 / 0]--
--[3 / 7]--
--[2 / 0]--
--[2 / 5]--
--[1 / 0]--
--[1 / 6]--
--[1 / 7]--
--[1 / 8]--
--[1 / 9]--
--[1 / 10]--
--[1 / 11]--
--[1 / 12]--
--[1 / 13]--
--[1 / 14]--
--[1 / 15]--
--[1 / 16]--
--[1 / 17]--
--[1 / 18]--
--[1 / 19]--
Total time: 0.009312
Nodes generated during search: 1765
Nodes expanded during search: 1560
Plan found with cost: NOTFOUND
Fast-BFS search completed in 0.009312 secs

Starting search with BFWS(novel,land,h_(add)ff)...
Landmarks found: 9
Landmarks_Edges found: 10
--[9 / 4294967295]--
--[9 / 55]--
--[9 / 46]--
--[9 / 38]--
--[9 / 30]--
--[8 / 30]--
--[6 / 30]--
--[6 / 28]--
--[6 / 27]--
--[6 / 26]--
--[5 / 26]--
--[5 / 24]--
--[5 / 19]--
--[5 / 17]--
--[4 / 17]--
--[4 / 16]--
--[4 / 12]--
--[4 / 11]--
--[4 / 9]--
--[2 / 9]--
--[2 / 8]--
--[2 / 7]--
--[1 / 7]--
--[1 / 5]--
--[1 / 4]--
--[1 / 3]--
--[1 / 2]--
--[1 / 1]--
--[1 / 0]--
--[0 / 0]--
Total time: 0.011755
Nodes generated during search: 991
Nodes expanded during search: 175
Plan found with cost: 39
BFS search completed in 0.011755 secs

Plan found:
0.00000: (MOVE-ENGINEER ENGINEER1 BAY1 CONTROL1)
0.00100: (UNSTUCK-UUV ENGINEER1 UUV1 CONTROL1 SHIP1 WAYPOINT2)
0.00200: (CAPTURE-IMAGE UUV1 WAYPOINT2 IMG-WP2)
0.00300: (TRANSMIT-DATA UUV1 IMG-WP2 WAYPOINT2 ENGINEER1 CONTROL1 SHIP1)
0.00400: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT3)
0.00500: (CAPTURE-IMAGE UUV1 WAYPOINT3 IMG-WP3)
0.00600: (TRANSMIT-DATA UUV1 IMG-WP3 WAYPOINT3 ENGINEER1 CONTROL1 SHIP1)
0.00700: (MOVE-UUV UUV1 WAYPOINT3 WAYPOINT5)
0.00800: (UNSTUCK-UUV ENGINEER1 UUV1 CONTROL1 SHIP1 WAYPOINT5)
0.00900: (MOVE-UUV UUV1 WAYPOINT5 WAYPOINT6)
0.01000: (PERFORM-SONAR-SCAN UUV1 WAYPOINT6 SONAR-WP6)
0.01100: (TRANSMIT-DATA UUV1 SONAR-WP6 WAYPOINT6 ENGINEER1 CONTROL1 SHIP1)
0.01200: (MOVE-UUV UUV1 WAYPOINT6 WAYPOINT4)
0.01300: (UNSTUCK-UUV ENGINEER1 UUV1 CONTROL1 SHIP1 WAYPOINT4)
0.01400: (PERFORM-SONAR-SCAN UUV1 WAYPOINT4 SONAR-WP4)
0.01500: (TRANSMIT-DATA UUV1 SONAR-WP4 WAYPOINT4 ENGINEER1 CONTROL1 SHIP1)
0.01600: (MOVE-UUV UUV1 WAYPOINT4 WAYPOINT2)
0.01700: (UNSTUCK-UUV ENGINEER1 UUV1 CONTROL1 SHIP1 WAYPOINT2)
0.01800: (MOVE-UUV UUV1 WAYPOINT2 WAYPOINT1)
0.01900: (COLLECT-SAMPLE UUV1 WAYPOINT1 SAMPLE-WP1)
0.02000: (MOVE-UUV UUV1 WAYPOINT1 WAYPOINT2)
0.02100: (UNSTUCK-UUV ENGINEER1 UUV1 CONTROL1 SHIP1 WAYPOINT2)
0.02200: (MOVE-UUV UUV1 WAYPOINT2 SHIP1)
0.02300: (STORE-SAMPLE UUV1 SAMPLE-WP1 SHIP1)
0.02400: (MOVE-UUV UUV1 SHIP1 WAYPOINT2)
0.02500: (UNSTUCK-UUV ENGINEER1 UUV1 CONTROL1 SHIP1 WAYPOINT2)
0.02600: (MOVE-ENGINEER ENGINEER1 CONTROL1 BAY1)
0.02700: (RETURN-TO-SHIP UUV1 WAYPOINT2 SHIP1 ENGINEER1 BAY1)
0.02800: (DEPLOY-UUV UUV2 SHIP2 WAYPOINT3 ENGINEER2 BAY2)
0.02900: (MOVE-UUV UUV2 WAYPOINT3 WAYPOINT5)
0.03000: (COLLECT-SAMPLE UUV2 WAYPOINT5 SAMPLE-WP5)
0.03100: (MOVE-ENGINEER ENGINEER2 BAY2 CONTROL2)
0.03200: (UNSTUCK-UUV ENGINEER2 UUV2 CONTROL2 SHIP2 WAYPOINT5)
0.03300: (MOVE-UUV UUV2 WAYPOINT5 WAYPOINT3)
0.03400: (MOVE-UUV UUV2 WAYPOINT3 SHIP2)
0.03500: (STORE-SAMPLE UUV2 SAMPLE-WP5 SHIP2)
0.03600: (MOVE-UUV UUV2 SHIP2 WAYPOINT3)
0.03700: (MOVE-ENGINEER ENGINEER2 CONTROL2 BAY2)
0.03800: (RETURN-TO-SHIP UUV2 WAYPOINT3 SHIP2 ENGINEER2 BAY2)
Metric: 0.03800000000000003
Makespan: 0.03800000000000003
States evaluated: undefined
Planner found 1 plan(s) in 3.518secs.