Adaptive Depth Contour Following Behavior

Christopher W. Gagner
NUWC Division Newport
Code 2534
christopher.w.gagner.civ@us.navy.mil

Scott R. Sideleau
NUWC Division Newport
Code 259
scott.r.sideleau.civ@us.navy.mil

Michael J. Walsh
NUWC Division Newport
Code 1511
michael.j.walsh2.civ@us.navy.mil

Michael L. Incze
NUWC Division Newport
Code 2534
Retired

Overview

- About US
  - NUWC Division Newport
  - Assets
- Adaptive Depth Contour Following
  - Concept of Operations (CONOPS)
  - Behavior / algorithm evolution
  - Supporting MOOS Apps
- Future Work

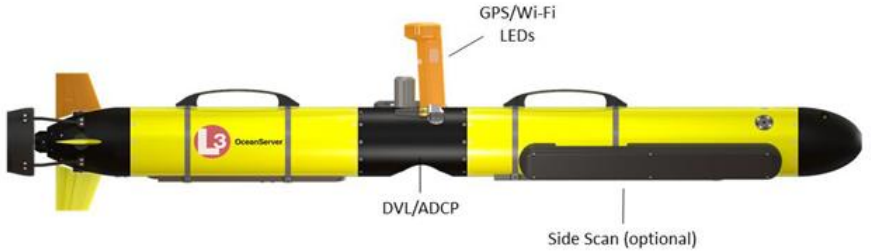
NUWC Division Newport

- RDT&E engineering and Fleet support facility for the US Navy
- Employing ~5k scientists and engineers (government and contractor support)
- Focus on all aspects of undersea warfare
  - Submarines
  - Offensive and Defensive Weapons Systems
  - Maritime UxVs

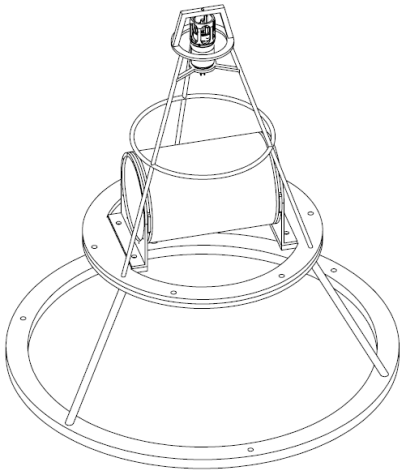
NUWC, Newport, RI

UxS Capability Development Team

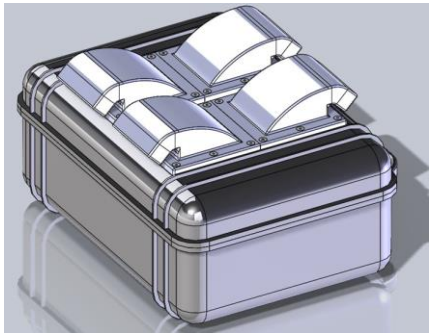
- Cross-department team with:
  - MOOS-IvP behavior development expertise
  - MOOS Application development expertise
  - Modeling & Simulation expertise
  - Operational expertise from participation in several US Navy and NATO exercises around the world over the past 15 years
- Maritime robotic platforms:
  - L3Harris Iver3 UUV - Three (3)
    - Communications: 1.8 GHz RF, Wi-Fi, ACOMMS (Blueprint SeaTrac)
    - Secondary CPU for running autonomy and intelligent communication
  - Seabed Node
    - Communications: ACOMMS (Blueprint SeaTrac)
    - Arm processor for intelligent communication
    - Integrated Compass, Accelerometer, and GPS for position and time (while on surface)
  - Comms Box
    - Communications: 1.8 GHz RF, Wi-Fi, ACOMMS (Blueprint SeaTrac)
    - Arm processor for intelligent communication
    - Integrated Compass, Accelerometer, and GPS for position and time



L3Harris Iver-3



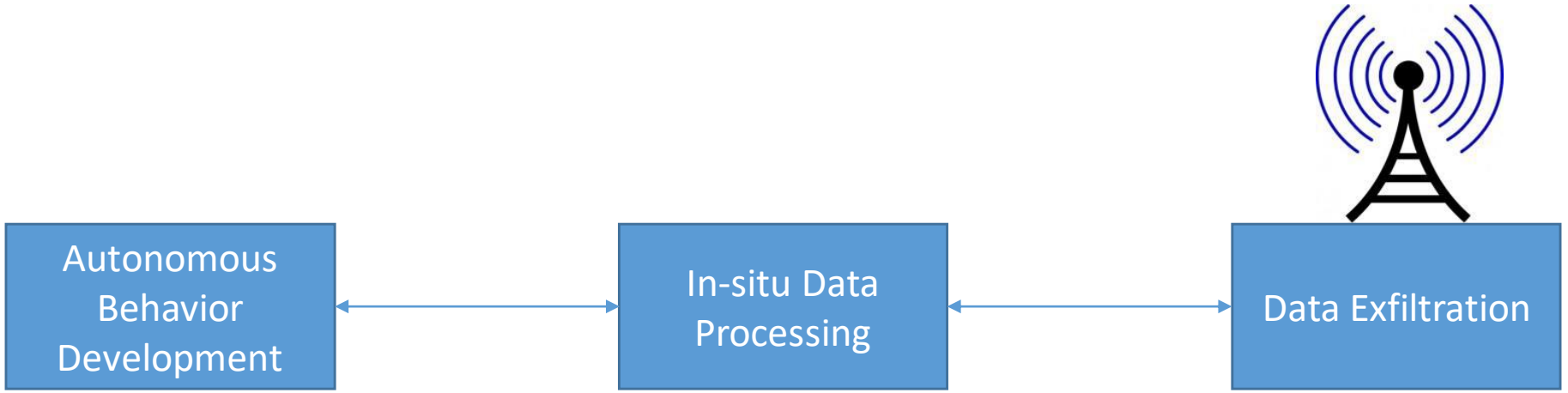
Seabed Node



Comms Box

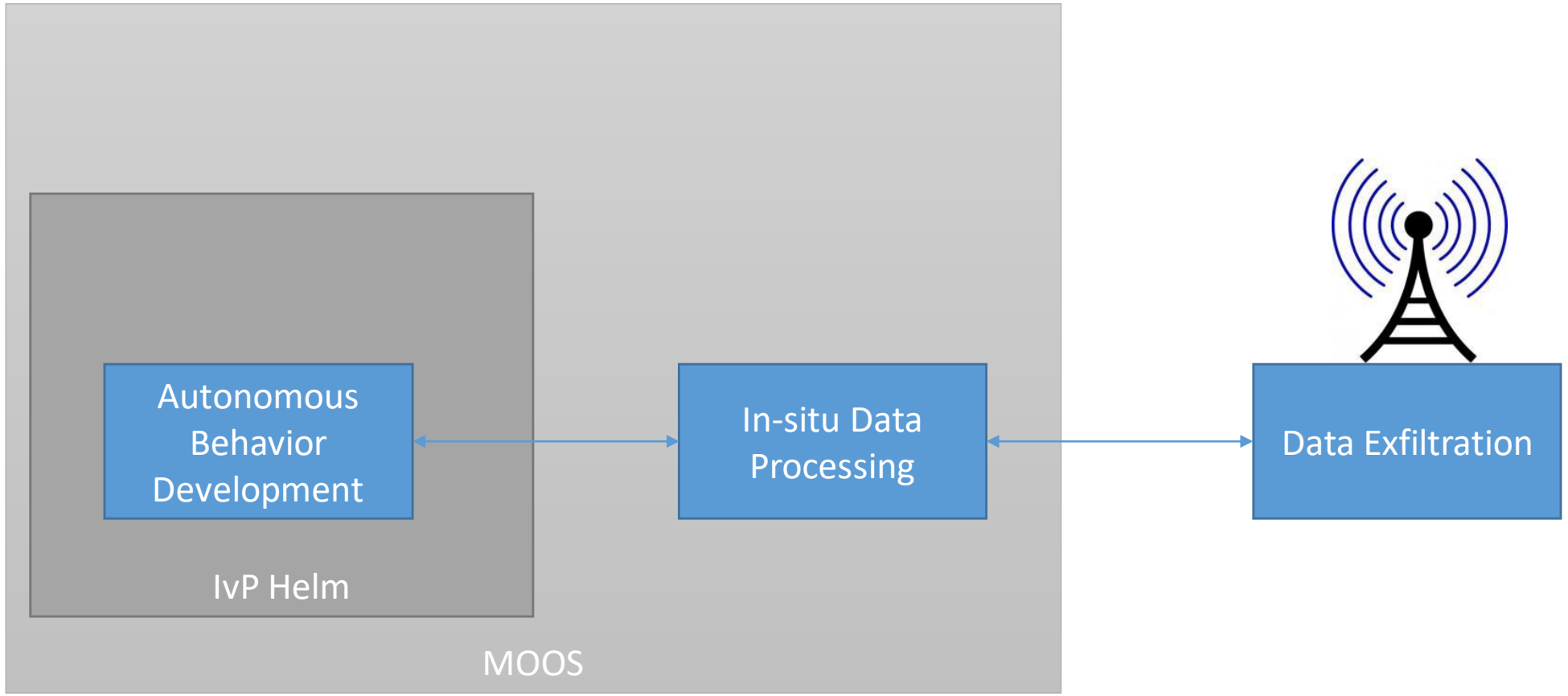
Modular assets with the ability to change on-board autonomy and communication protocols to meet exercise demands.

CONOPS: Three Pillars



Not just behavior development…

MOOS-IvP + CONOPS

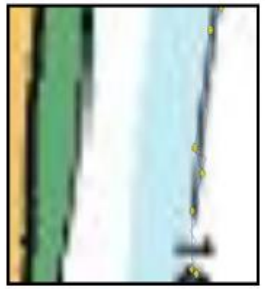


Full operational concept development demonstrated on surrogate platforms.

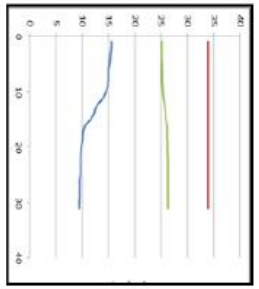
Serial Missions & In-Situ Data Processing

In-stride IPOE information products

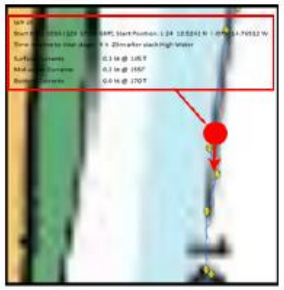
COIs (ATR imagery)



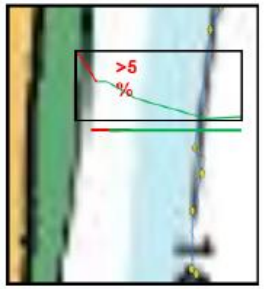
Depth Contours



Ocean Profiles



Currents



Slope

JAUS

Autonomous Evaluation and Planning

Manned C2 Node

Direct UxS - UxS Collaboration

Desire for Adaptive Depth Contour

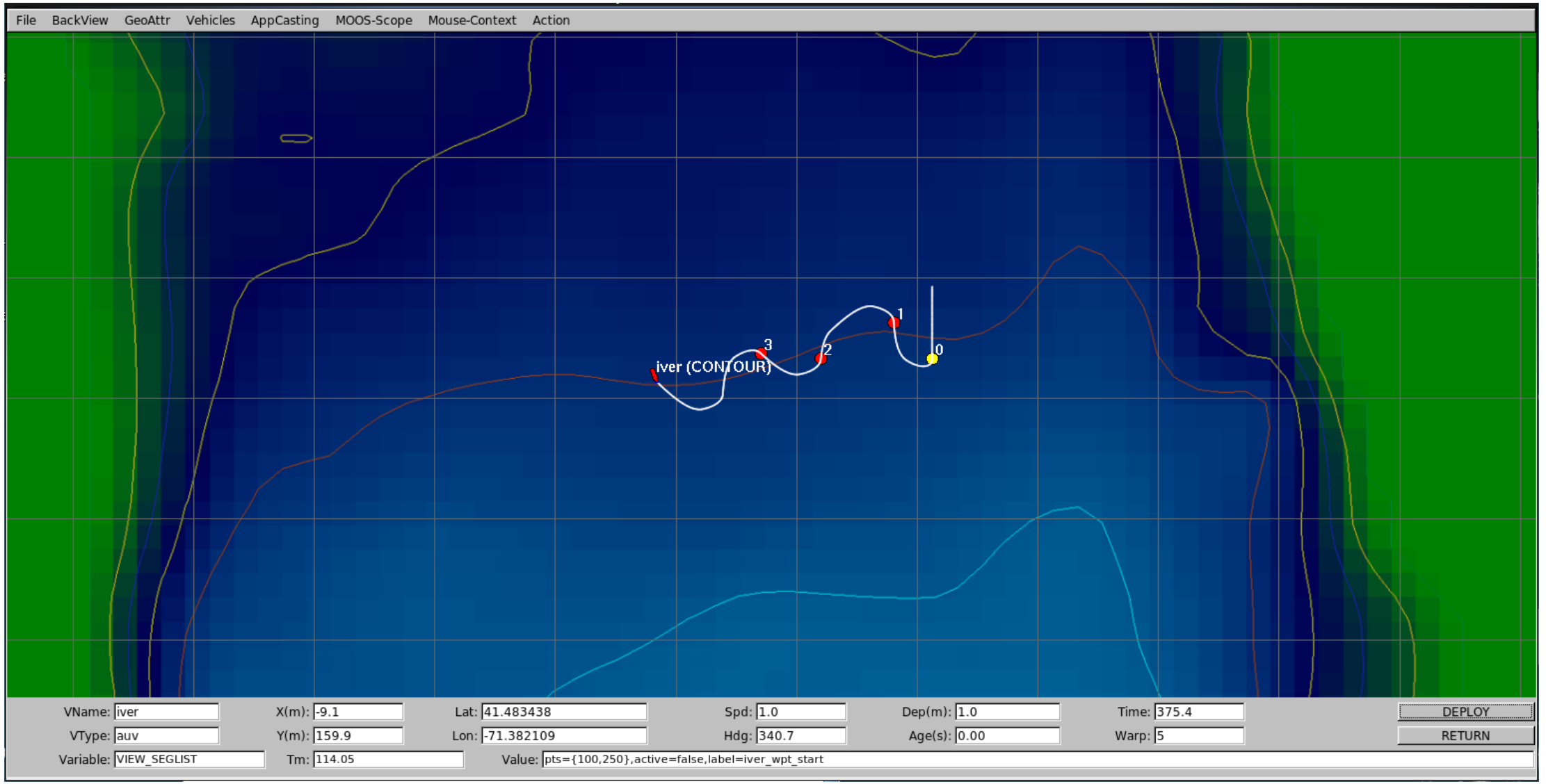
- Warfighters increasingly in operational areas where low resolution data exists close to shore
- Warfighter need to map depth contours of interest for:
  - Ingress/Egress
  - Other types of route planning (mission specific)
  - Safety
- In the past, warfighter might have mapped contours manually or waited for third-party data
- Can we enable robots to increase the OPTEMPO?

Rapidly provide an 80% solution to the warfighter.

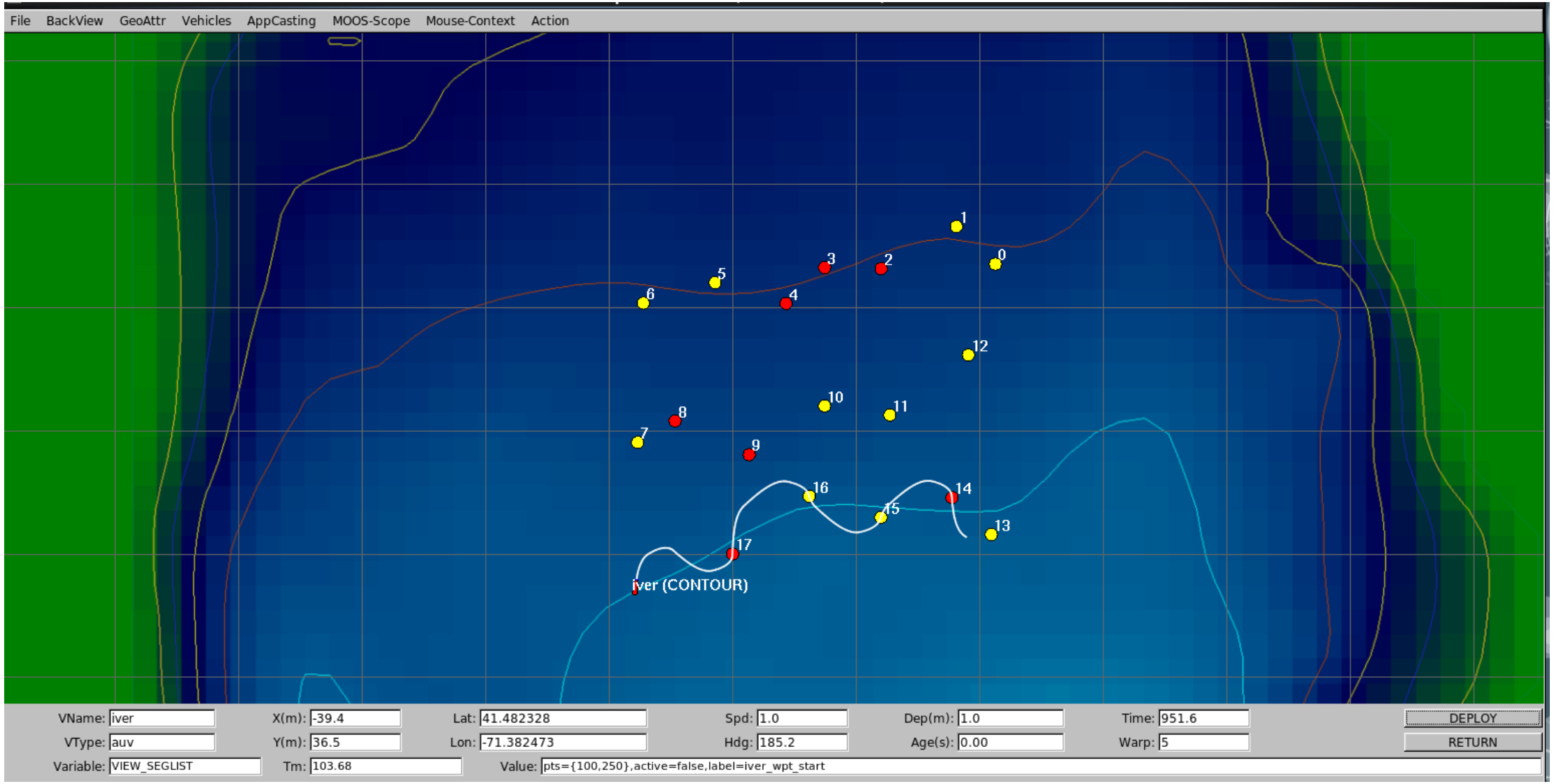
Rule-Based Adaptive Contour Behavior

- Description:
  - Originally developed in 2010 by Scott Sideleau and Mike Incze – Presented at MOOS-DAWG 2015
  - Utilizes single depth/altitude measurement to track desired total water depth
- Inputs:
  - Contours: List of Contours (m)
  - Distance to follow the contour (m)
  - Initial heading (degrees)
  - Maximum initial distance (m)
  - Deadband (m): Contour tolerance
  - First Turn Direction (Left/Right)
- Approach:
  - On startup, determine if starting shallow or deep
  - Travel on `initial_heading` until contour crossed
    - If `max_distance` reached without crossing contour, abort the behavior
  - If we cross the contour of interest…
    - Mark the location (X,Y)
    - First Crossing:
      - Increment/decrement desired heading by 135-degrees and resume search
        - To avoid traveling parallel to the contour if crossing was perpendicular
    - Subsequent crossings:
      - Increment/decrement desired heading by 45-degrees and resume search
  - If we don't cross the contour of interest (after first crossing)…
    - Spiral outward and continue the search
    - If `max_distance` (tracked while creating the spiral) reached without crossing contour, abort the behavior

Rule-Based: Single Contour Tracking



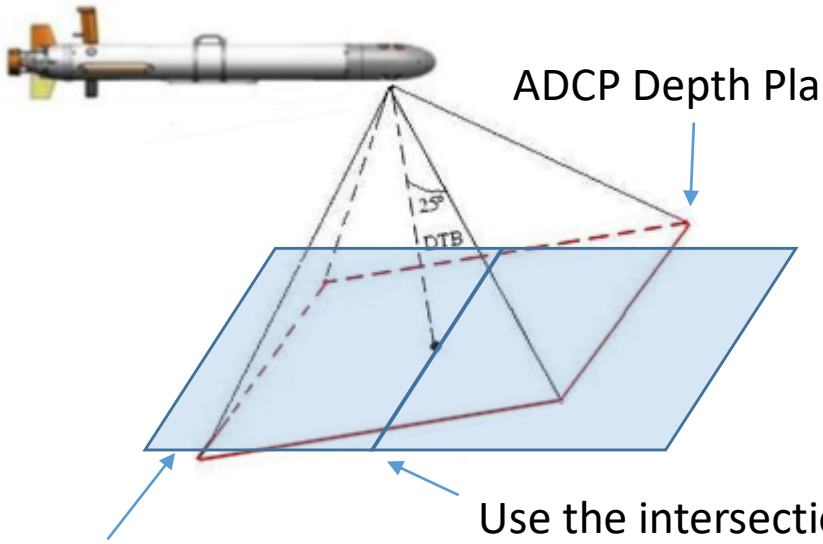
Rule-Based: Multiple Contour Tracking

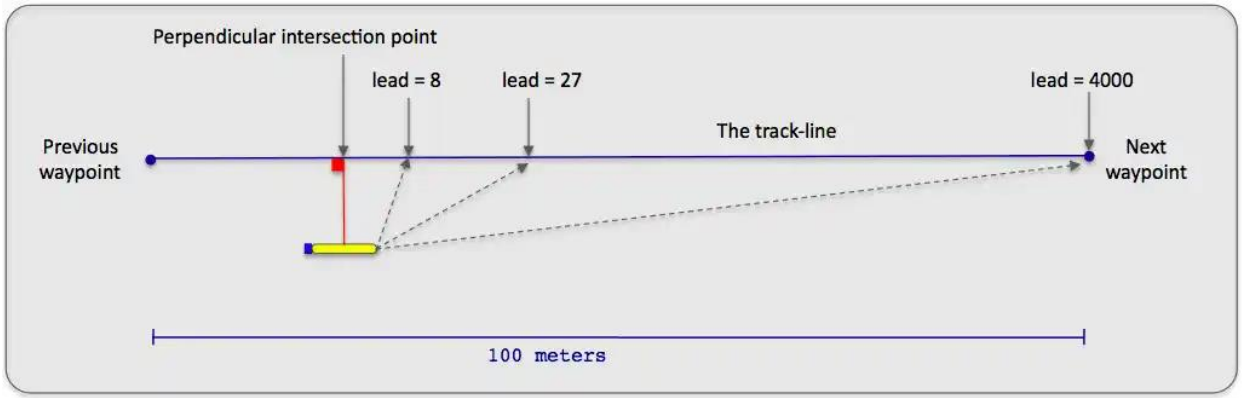


Improving the Rule-Based Approach

Proposed by Sideleau/Incze in 2015:

- Construct and map plane using DVL/ADCP data
  - By mapping plane, should be able to reduce total distance traveled
  - Desire to maintain a stream of points of contour of interest
    - Current approach only affords individual soundings on crossings



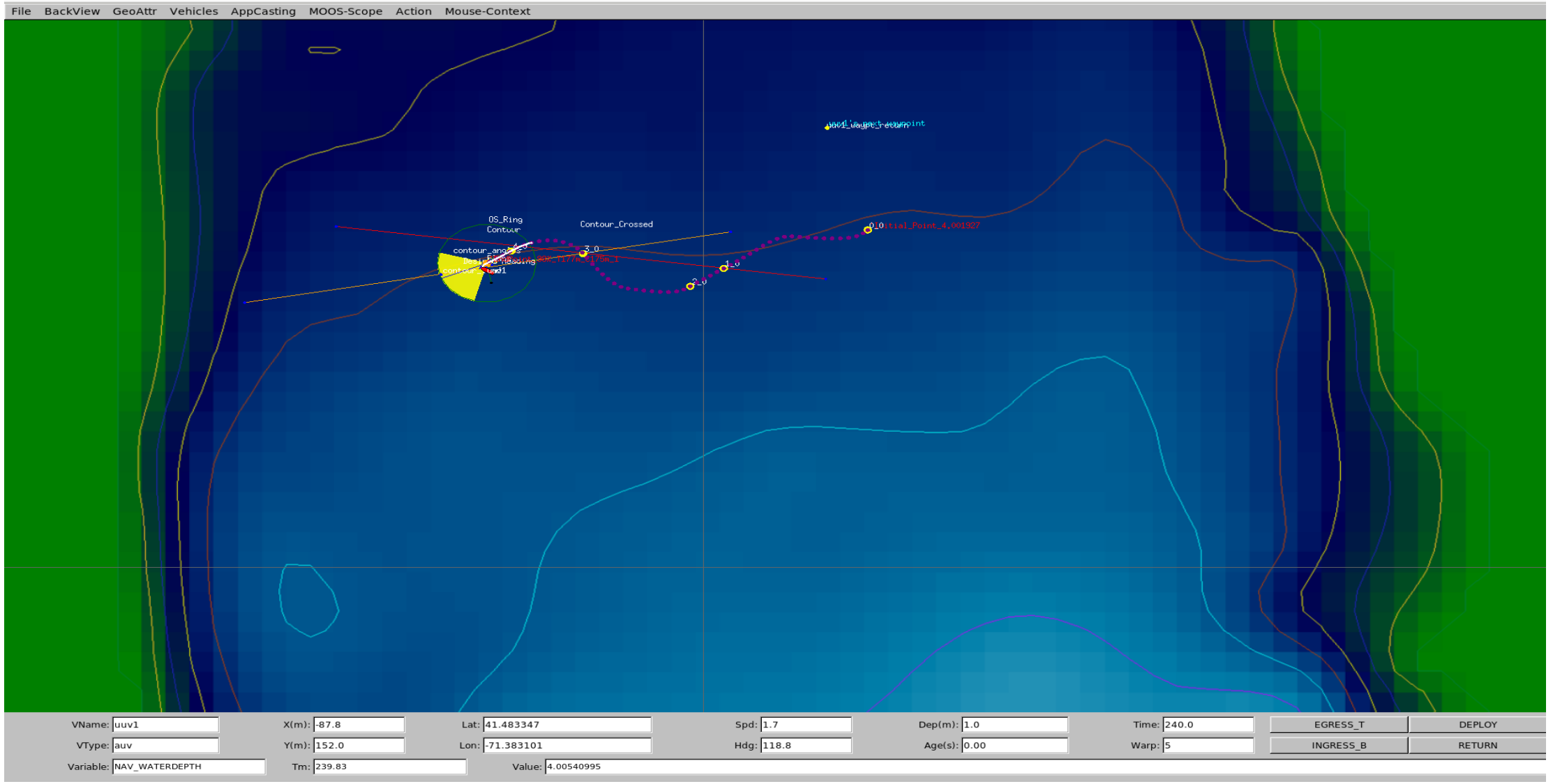


IvP Waypoint Behavior uses lead distance to control heading preference. The same approach is used to here to determine the desired heading to the contour track-line.

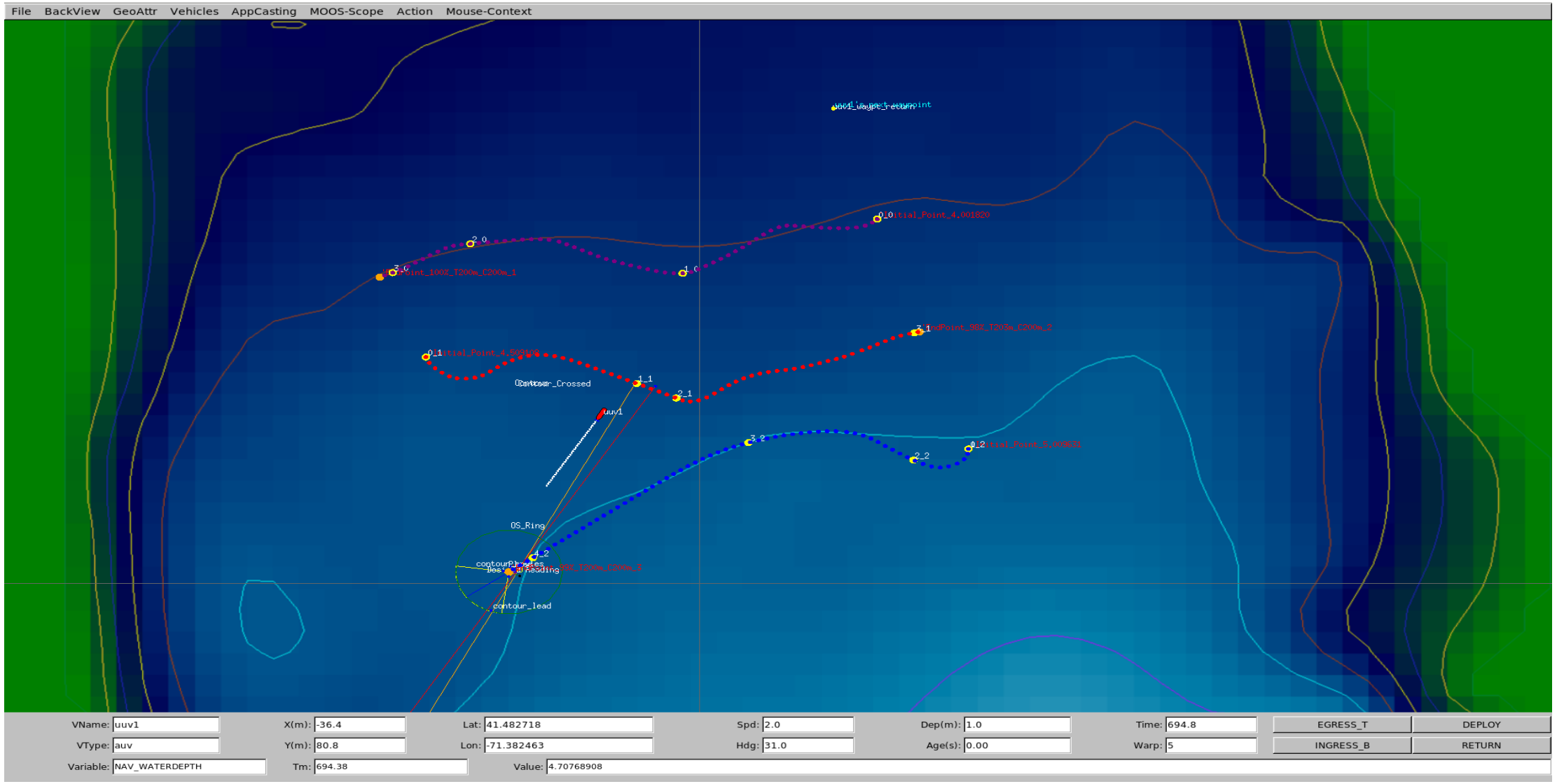
Plane-Based Adaptive Contour Behavior

- Additional Inputs:
  - Maximum Heading Change: Restricts the heading changes to prevent aggressive turns
  - Lead Distance (m):
  - Nudge Angle (degrees): Used to force the vehicle to cross the contour
- Approach:
  - On startup, determine if starting shallow or deep
  - Travel on `initial_heading` until contour crossed
    - If `max_distance` reached without crossing contour, abort the behavior
  - If we cross the contour of interest…
    - Mark the location (X,Y)
    - After Crossing:
      - Compute the ADCP Depth Plane
      - Intersect the ADCP Depth Plane with the desired depth plane to determine the track line
        - Intersect the desired bottom track line with a circle with the radius equal to the lead distance
          - Select the intersection point(s) that minimized heading change
          - Add or subtract the nudge angle (based on shallow or steep)
          - If there is no intersection, compute a heading perpendicular to the desired bottom track line
        - Clamp new desired heading to maximum heading change
      - If the planes do not intersect:
        - If we are within the deadband, continue on current heading
        - Otherwise, spiral outward and continue search
  - If we don't cross the contour of interest (after first crossing)…
    - Spiral outward and continue the search
    - If `max_distance` (tracked while creating the spiral) reached without crossing contour, abort the behavior

Plane-Based: Single Contour Tracking



Plane-Based: Multiple Contour Tracking



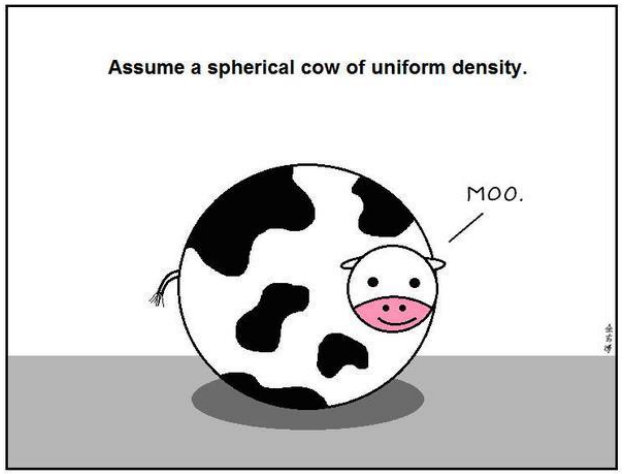
Supporting MOOS Applications

- **uBathyProvider**:
  - Subscribes to NAV_X and NAV_Y and publishes NAV_WATERDEPTH
  - Supports gridded and sparse datasets (SQLite)
  - Modified to support providing water depths at ADCP Beam locations
- **uSimRollPitch**:
  - Augments uSimMarine by publishing NAV_PITCH and NAV_ROLL
- **pContourCodec**:
  - Subscribes to the contour crossing points published by the contour behavior(s) and publishes a simplified contour line
- **pJAUSContourProvider**:
  - Subscribes to the simplified contour line produced by pContourCodec and distributes over the network using a custom JAUS service

Common Problems with Simulation

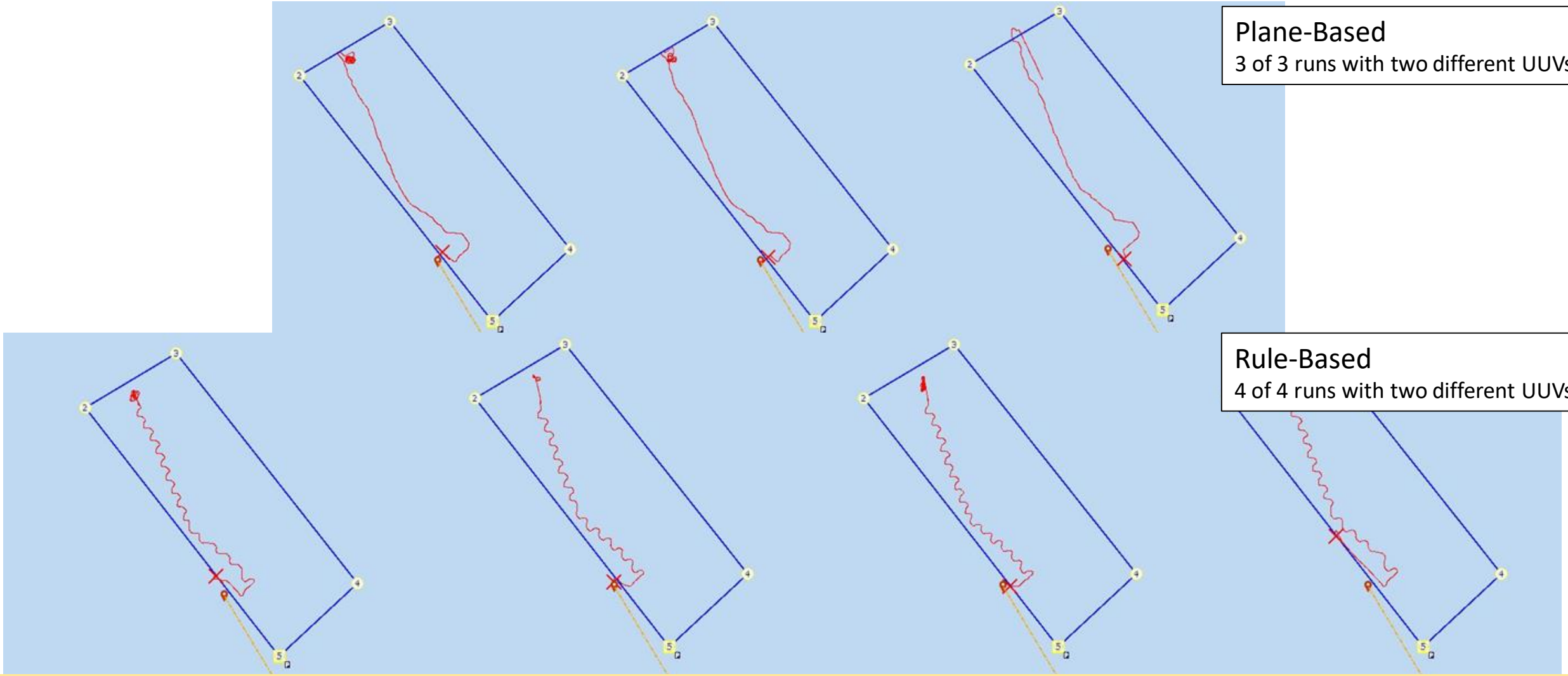
- Vehicle Dynamics:
  - Depth control: Our UUVs cannot change depth without moving forward. Effects achieving initial depth from the surface and maintaining depths while submerged
  - Roll/Pitch: uSimMarine does not provide roll or pitch. Effects projecting ADCP beams. Created uSimRollPitch.
- Data sampling:
  - DVL Beam simulator initially provided updates at 4Hz. Actual ADCP Beams are updated at 0.5 Hz.
- Lack of independent developers for creating simulator components and vehicle behaviors.
  - Same math/logic used in both places –Mistakes in one side fixed by the other side

Simulation



Reality

Comparison of Plane-Based vs Rule-Based Contour

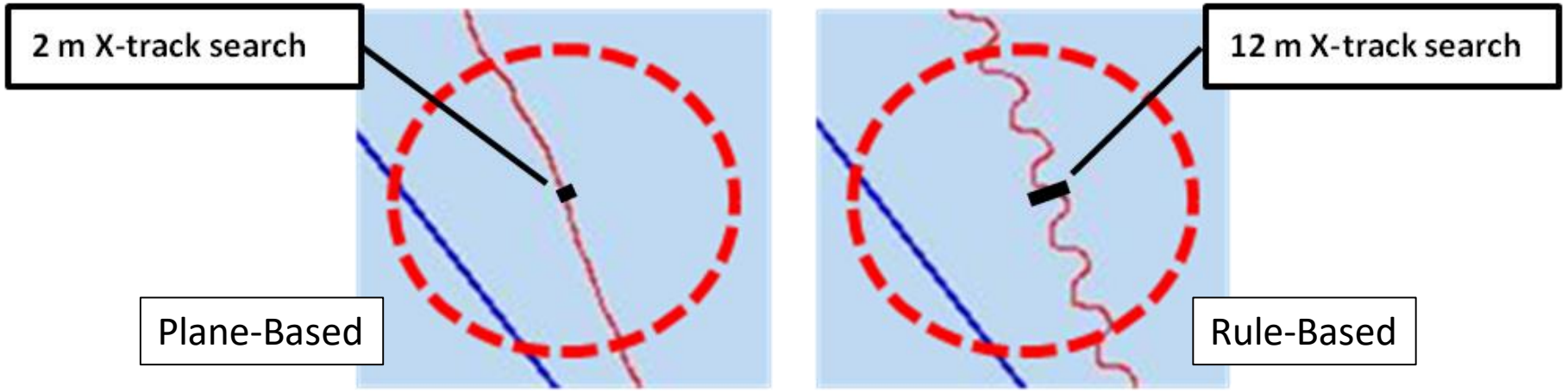


Plane-Based
3 of 3 runs with two different UUVs

Rule-Based
4 of 4 runs with two different UUVs

Data collected during Contested Urban Environment (CUE) 2021 – Portsmouth, UK

Comparison of Plane-Based vs Rule-Based Contour



- Reduces platform and operational risk
- Increases ONSTA time, transit opportunity, serial tasking, shared resources for dynamic re-tasking
- Improves accuracy with reduced vehicle dynamics
- Operates where standard mission profiles can not

Plane-Based Behavior: 30% increase in efficiency over Rule-Based (reduction in survey time)

Future Work

- Enable the contour behavior take a desired search area
  - Both behaviors currently
    - Only support an initial heading
    - Multi-contours are dependent on the initial heading
    - Require the use of BHV_OpRegion for safety
- Monte Carlo Behavior Evaluation
  - The new Plane-Based approach has addition input parameters that have been chosen based on manual tweaking
  - Utilizing a Monte Carlo simulation to evaluate the behavior parameters would allow for selecting the parameters that perform best in most environments
- Update the behavior to process live side-scan sonar bathymetry
- Collaborative multi-contour mapping

Backup

Multi-Domain Vehicle Simulators

Background: Our demonstrations often involve vehicles in the air domain. Creating a comprehensive simulation helps identify potential problems in the mission. However, uSimMarine and BHV_ConstantDepth prohibit robots from achieving negative depth.

Workaround:

- moos-ivp-umassd
  - uSimpleRobot: Basic vehicle simulator that allows for negative dept
  - BHV_SimpleDepth: Depth behavior that allows for negative depth
  - https://github.com/scottsideleau/moos-ivp-umassd
- pHelmIvPConfiguration: Modify depth domain to support negative depth:
  - `Domain = depth,-200:0:4000:optional`

Proposal:

1. Update manifest for moos-ivp-umassdand add it to moos-ivp.org
2. Solicit MOOS community for other open-source multi-domain simulators
3. Create a new simulator that supports multiple domains