Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 7873–7879

Automated Planning for Inspection and Maintenance operations using Unmanned Ground Vehicles ⋆

M.A. Hinostroza * Anastasios M. Lekkas *
Aksel A. Transeth ** Bjørnar Luteberget **
Christian de Jonge * Svein Ivar Sagatun *****

* *Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway.*
(e-mail: {miguel.hinostroza, anastasios.lekkas} @ntnu.no)
** *SINTEF Digital, Mathematics and Cybernetics, Trondheim, Norway.*
(e-mail: {aksel.a.transeth, bjornar.luteberget} @sintef.no)
*** *Equinor, Norway.*
(e-mail: {chjo; svesa} @equinor.com)

Abstract:
Offshore oil and gas industry has a strong incentive to improve its traditional operations and move towards more remote controlled and automated installations. This allows for improved efficiency, reduced cost and improved quality, and safety by removing personnel out of harm's way. The use of Unmanned Ground Vehicles (UGVs) in these upcoming platforms, is relevant for Inspection and Maintenance (I&M) operations. Traditionally, UGVs are used only for pre-defined tasks and have no capabilities for replanning, if a new task is required or any unexpected event occurs. This paper presents a novel concept for I&M operations using automated planning for UGVs. The automated planner is based on a temporal planning algorithm, and considers actions related to, for example, visiting a specific waypoint, inspect a sensor or manipulate an actuator. Also, the proposed system allows to perform replanning in case of any specific location needs to be revisited or a path is blocked. In addition, we couple the mission planner with a UGV guidance, navigation and control system, which has path planning, path following and control capabilities. To assess the performance of the proposed system, an use case for I&M operations on board of an oil and gas platform was simulated and promising results were obtained.

Keywords: Automated planning, maintenance and inspection, oil and gas platform, unmanned ground vehicle.

1. INTRODUCTION

Offshore oil and gas platforms are often located in remote and distant places and may pose a challenging environment for personnel due to the exposure to potential hazardous or harmful chemicals, work in areas exposed for weather and on smaller installations with hydrocarbons under pressure (Chen et al., 2014; Transeth et al., 2010). Consequently, increased use of automation and remote control represent a desirable development (Tan et al., 2020). Currently, various equipment and instruments that are installed on an offshore facility may already be automatic.

Inspection and Maintenance operations (I&M) are crucial to ensure a high production efficiency, asset integrity and for safe operations of offshore platforms. These tasks are:

- periodic or on-demand visual inspection, of equipment, structure or surfaces;
- periodic or on-demand acoustic inspection using directional sound looking for anomalies or vibrations;
- thermal (using infrared) inspection of electrical equipment, process equipment and heated surfaces to look for leaks, anomalies in temperature;
- thermal (using infrared) for detection of small (fugitive) gas leaks and monitoring of these;
- taking sample of production fluids for analysis;
- calibrating and verify operations of flame and gas sensors;
- inspection of process equipment, structures, surfaces or containers with respect to corrosion using non-destructive testing (e.g., ultrasound or eddy current).

Unmanned robot solutions for offshore platforms have traditionally been designed for subsea tasks, (Bengel et al., 2009), and performed by remotely operated vehicles (ROVs) or autonomous underwater vehicles (AUVs). More recently, the idea of using mobile ground vehicles in topside offshore applications for I&M purposes have been introduced in Soldan et al. (2012), however, no implementations of mobile robots in real offshore environments could be observed so far in the literature. Nevertheless,

⋆ This work was funded by the Research Council of Norway (RCN) project "Autonomous robot missions with AI-based planning and acting"(ROBPLAN), project number 32274.

10.1016/j.ifacol.2023.10.022

using robots for these types of inspection is attractive since robots enable for repetitive, accurate, geotagged and low-cost inspection. These ideas and challenges are currently being studied in Olsen et al. (2023); Solem et al. (2023).

Unmanned Ground Vehicles (UGVs) have appeared since the 60s. The first major mobile robot development effort was "Shakey", developed in the late 1960s to serve as a testbed for AI work at Stanford Research Institute (Nilsson et al., 1984). Nowadays, the use of autonomous and semi-autonomous vehicles to perform inspection tasks enhances the efficacy and the safety of the operation; however, many technical problems, such as those pertaining to the precise positioning and path following of the vehicles, robust obstacle detection, and intelligent control, are still under development. Also, the use of UGVs is only for predefined tasks and has no option to replan if a new task is desired or any unexpected event occurs.

To overcome this limitation, the use of automated planning or AI planning onboard of UGVs is a promising solution (Transeth et al., 2022). AI planning, also called automated planning and scheduling, is a sub-area of artificial intelligence (AI) that studies the deliberation process in a computational way to be applicable to intelligent agents and robots (Ghallab et al., 2004). The field of automated planning has contributed significantly to advanced fields like Space exploration (Rabideau et al., 1999). The first AI planning algorithm, *STRIPS*, was developed in 1971 at Stanford University to solve a basic planning problem for a mobile robot, and since then the field has progressed significantly and contributed to applications such as space-exploration (Bernard et al., 1999), inspection (Streich and Adria, 2004), prevention of disasters, rescue operations and operations with robots and autonomous systems (Pinto et al., 2012; Xue and Lekkas, 2020; Hinostroza and Lekkas, 2022). In addition to STRIPS, other fundamental AI planning algorithms are *hierarchical task network (HTN)* (Erol et al., 1994) and *GraphPlan* (Blum and Langford, 1999). The main difference between these fundamental methods is the space where the plan is searched, STRIPS and GraphPlan are state-space search methods and HTN is a task-network based search. In recent years, with the advances in microchips technologies and increasing of the computational power,a new generation of high-efficiency and complex AI planners have been developed, for example to include temporal constrains (Barreiro et al., 2012) and/or system dynamics (Piotrowski et al., 2016). Temporal planning refers to the process of creating a plan that takes into account the order and duration of actions, as well as their relationship to time. In temporal planning, a plan is represented as a sequence of actions and the time at which each action should occur. The goal of temporal planning is to find the best plan that satisfies temporal constraints, such as deadlines and resource availability, and accomplishes the desired goals in a timely manner.

This paper presents a novel concept for I&M operations on topside of oil and gas offshore platforms implementing automated planning for UGVs. The objective of this work is to increase the autonomy's level of traditional offshore oil and gas platform using UGVs. The system integrates a temporal planner into a traditional guidance navigation and control system (GNC) for mobile robots. The main idea is to build an hierarchical architecture where the AI planner (placed at top-level) based on a pre-known problem domain, actions and goal computes a temporal plan, which will be transferred to the GNC system (low-level) to be executed. The temporal AI planner chosen for this study is the Simultaneous temporal planning (STP), this method is chosen due to its robustness and good performance. The STP was the runner-up in the 2018 International Planning Competition (Furelos Blanco et al., 2018). The GNC system is composed of path planning based on a hybrid A-star algorithm, path-following based on Line-of-sight. In order to assess the performance of the proposed system, numerical simulations were carried out for a real offshore facility, with static obstacles, valves and pumps for an inspection mission. The main contributions of this paper can be summarized in the following list:

- Modelling a typical oil and gas offshore environment for a high-level mission planning purposes using UGVs;
- Implementation of a real-time automated planner running simultaneously with a traditional low-level control for UGVs;

2. METHODOLOGY AND SYSTEM OVERVIEW

2.1 Proposed System Architecture

The system architecture proposed in this work is presented in Fig. 1. Each rounded square represents a module of the system. In this paper we are focusing on the mission planning and UGV interface modules. Detailed information about the integration of automated planning and GNC system can be found in Hinostroza and Lekkas (2022), about the path-following control module in Hinostroza et al. (2020). The modules are:

- *UGV interface* - ROS interface between the Gazebo simulator and inputs from the GNC module.
- *Control system* - From the temporal plan, this module computes the robot attitude set-points.
- *Kalman Filter Estimator* - Module for estimating the actual-state of the system.
- *Mission planner* - From a problem domain definition computes a temporal plan, is composed of two sub-modules.
    - · *Problem domain refiner* - Sub-module to refine the input problem domain definition.
    - · *STP planner* - Computing the temporal plan composed of a sequence of actions.
- *System dynamics update* - Reading the user operator commands and unexpected events.
- *Output saver* - Saves estimation outputs for later use and analysis.

2.2 Mission Planner

The mission planner is based on the Simultaneous Temporal Planner (STP) (Furelos Blanco et al., 2018). This algorithm was the runner-up algorithm in the 2018 International Planning Competition (IPC). It relies on a transformation from temporal planning to classical planning, constructing a temporal plan by finding a sequence of classical actions that solve the problem while satisfying a given set of temporal constraints. The main contribution

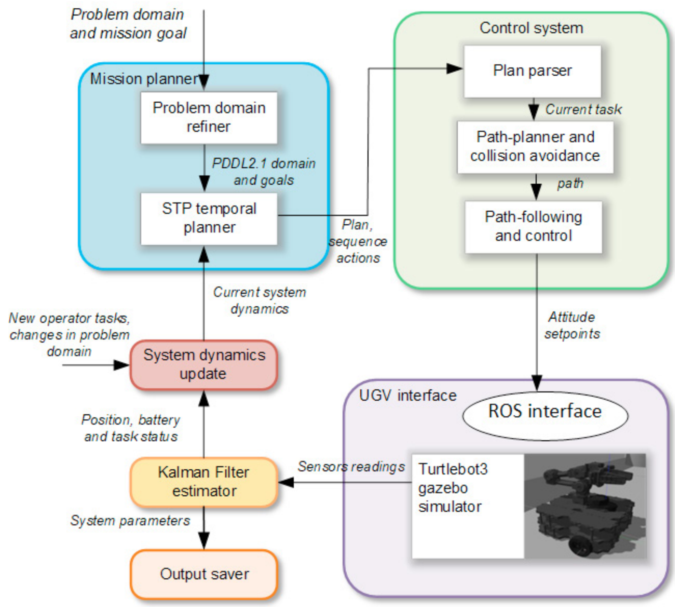

Fig. 1. Software architecture describing high level interactions between the different modules in the system.

is that it can solve problems that require simultaneous events, i.e. the temporal actions have to be scheduled in such a way that two or more of their effects take place concurrently. To do so, STP separates each event into three phases: one phase in which temporal actions are scheduled to end, one phase in which simultaneous effects take place, and one phase in which temporal actions are scheduled to start.

In this work, the STP planner was chosen over other well-known planners such as ROSPLAN (Cashmore et al., 2015), Europa (Barreiro et al., 2012), and T-rex because of its simplicity in being integrated into a GNC system. The STP planner is coded in a recent version of Python and Linux, making it possible to run simultaneously with a GNC system programmed in Matlab. This is not the case with the other planners. For example, ROSPLAN is a black box that is not easy to add new blocks or extra code to, and the Europa planner has been discontinued since 2012 and is not compatible with current versions of ROS and Python. Moreover, this work implements a temporal planner because actions and effects take place over a period of time, and the possibility of taking actions may depend on events and other actions taking place simultaneously, i.e., durative actions do not sequentially follow each other, but may temporally overlap and interfere.

2.3 Refinement Block and Battery Modelling

The initial plan calculated by the STP planner, is computed based on preliminary data from previous measurements, Kalman filter estimations, statistical data. In order to obtain a more accurate plan, the plan refinement block is in charged to revise each individual action to verify if this action is feasible to execute by the robot. Thus, this module analyzes each action of the plan in combination with the current status and system dynamics to determine if the action is feasible to be executed or needs to be refined. The refinement process is done iteratively until a convergence is reached. The STP planner has no capabilities to include numerical constraints such as battery level, normally, this characteristics are implemented in hybrid planners (Ghallab et al., 2004). Also, the inclusion of battery level will considerably increase the computational cost of the STP planner. In this work, the battery level is modelled in the GNC module and is updated during task execution, when the battery level is below 30% a replanning command is sent to the STP planner.

The threshold of 30% for battery level has been taken from use cases produced by Equinor, Norway. These use cases, along with the challenges of offshore robot deployment and ways to solve them, are currently being studied in Olsen et al. (2023).

2.4 I&M Use Cases

There are three main categories of robot missions:

- scheduled missions e.g., the robot undertakes a standard inspection round,
- on-demand regular missions such as ad-hoc inspection tasks of some selected equipment, and
- emergency handling missions.

In this article the focus is on use cases belonging to the two first categories.

2.5 Algorithm of the Overall System

The algorithm of the overall system, automated planner plus mission planner and UGV interface, is presented in Fig. 2. The algorithm starts reading information about the oil and gas offshore facilities domain, allowed actions, initial states and mission goals. Then, the temporal automated planning module computes an initial temporal plan. This temporal is refined by the block of plan refinement. This new refined temporal plan is parsed into a simple sequence of task and duration. These individual tasks are translated to the control system of UGV to be executed by the low level control system. When a task is finished the algorithm updates the actual state of the system, problem domain, remaining goals, etc. in the Kalman filter estimator module. If any additional tasks need to be performed or an unexpected event occurs later on, we update the problem domain and perform replanning. Note that if replanning is necessary, the current problem domain and remaining goals will be transferred back to the temporal STP planner, and the algorithm will be restarted.

3. NUMERICAL SIMULATIONS AND RESULTS

In order to assess the performance of the overall system (mission planning, path planning, path-following, control and UGV interface) in an realistic inspection and maintenance scenery, a case-study for a real oil and gas offshore platform is presented. The physical place is the "Huldra" offshore platform and the UGV considered for simulations is the "Turtlebot3" from Robotis company.

3.1 Place for Simulations: Huldra Oil and gas offshore platform

The plant chosen for simulations is the Huldra Oil and Gas offshore platform from Equinor, Norway, see Fig. 3a.

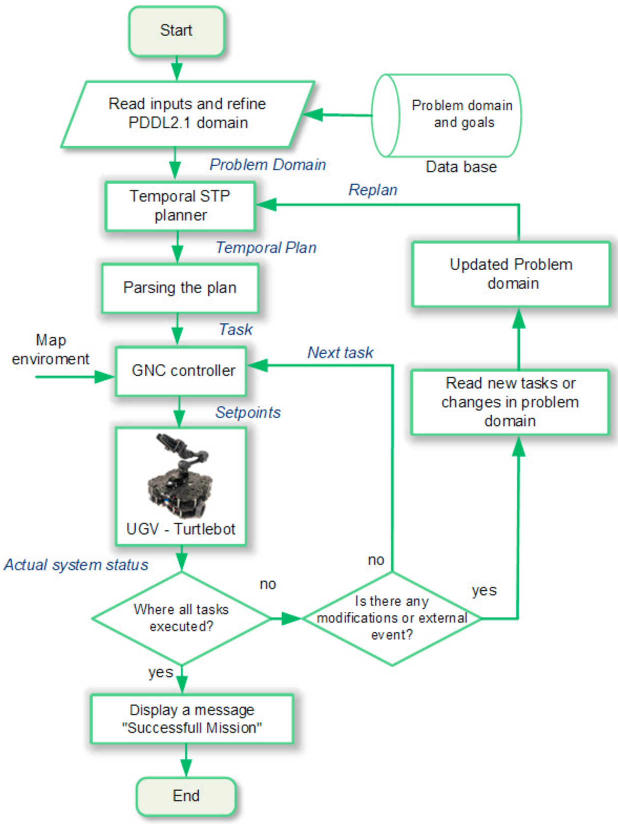

Fig. 2. Algorithm flow chart of proposed system

The 3D model and plant description was recently released under open-source license by Equinor[1] for research and innovation developments. In order to perform numerical simulations, the plant was simplified as can be seen in Fig. 3b, additionally a Gazebo map was created in Fig. 3c to perform simulations in ROS, where 1 grid map is equal to 1m.

3.2 Vehicle: Turtlebot3 UGV

The UGV considered for the simulations is the Turtlebot3, Fig. 4, from ROBOTIS[2] company. TurtleBot3 is a small, affordable, programmable, ROS-based mobile robot for use in education, research, hobby, and product prototyping. The work in this paper is based on the "waffle-pi" model, also the TurtleBot3 can be used as a mobile manipulator capable of manipulating an object by attaching a manipulator like OpenManipulator-X. The Turtlebot3 includes a ROS-gazebo simulator which was used in this work.

3.3 Problem domain definition

In this subsection the problem domain definition for the use case of I&M operations in an offshore platform is presented. The waypoints, actions and agents are presented in Fig. 5. The domain considers 8 waypoints, 3 valves, 2 pumps and 3 charge stations and static obstacles.

Variables: The variables and agents for this preliminary study are shown in Table 1. Thus, one robot, battery,

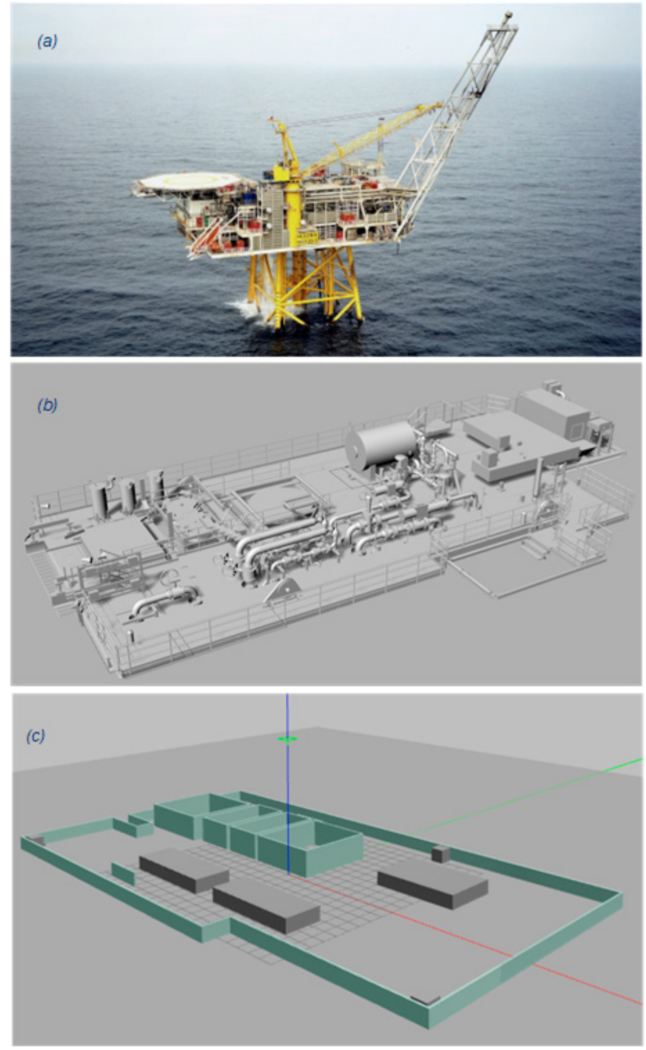

Fig. 3. (a) Huldra oil and gas offshore platform (Courtesy of Equinor), (b) Upper-layer of Huldra, (c) Simplified ROS gazebo map.

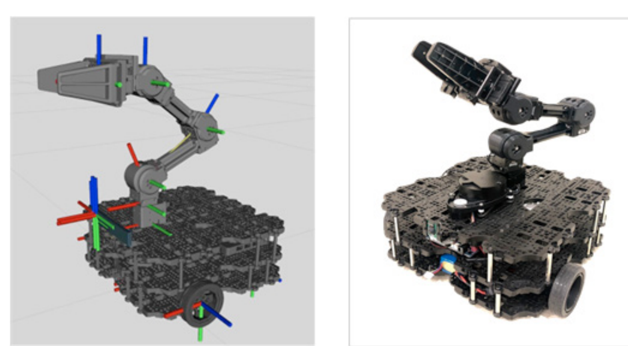

Fig. 4. Turtlebot3 waffle-pi with OpenManipulator-X.

8 way points, 3 valves, 2 pumps and 3 battery charger stations are considered. Additionally, the status of the battery level is defined as *full-battery* and *low-battery*. All the connections between way-points are assumed (except when the operator reports some blocked path). Note that valves, pumps and chargers stations are located in a prior-defined fixed waypoint.

[1] https://data.equinor.com/
[2] https://emanual.robotis.com/

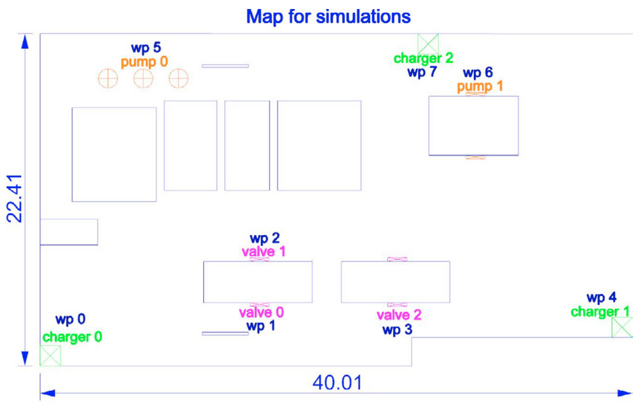

Fig. 5. Problem Domain map, waypoints and location of valves, pumps and charger stations.

Table 1. STP planner domain variables.

Agent	Variable	Notation
robot	turtlebot0	?V-vehicle
battery	battery0	?B-bat
valves	valve0,valve1,valve2	?C-valve
pumps	pump0,pump1	?P-pump
charger stations	$charger_i$, i = 0,1,2	-
way points	$waypoint_j$, j= 0,1,2,..,7	?W-wp

Actions: Based on traditional tasks in I&M onboard of oil and gas production offshore platform, presented in section 2.4, the actions considered for this preliminary study are presented in Table 2. There are mainly 5 kinds, move the UGV between waypoints, take an infrared (IR) picture, take an electro-optical (EO) picture, go to the battery charge station and charge the battery. The type of actions are modelled as durative-actions, the duration for move actions is based on the distance between waypoints and UGV speed, where ΔT is the simulation time, in seconds.

Table 2. Problem Domain - durative actions.

Action	Parameters	Duration (ΔT)
durative-action move-turtlebot	?V - vehicle ?O - location ?L - location ?R - route	-
durative-action picture-ir	parameters (?V - robot ?L - location ?G - camera-ir ?P - pump ?T - battery)	12
durative-action picture-eo	(?V - robot ?L - location ?G - camera-eo ?B - valve ?T - battery)	10
durative-action charge-battery	(?V - robot ?L - location ?B - battery ?C - charger)	10
durative-action move-charger	(?V - robot ?O - location ?L - location ?R - route ?B - battery)	-

Operator additional tasks and external events: The presented system has capabilities to include additional operator tasks, such as revisiting a way-point, or inform of unexpected events such as blocked paths during mission execution. These additional events can be obtained from an Analytics Engine (Transeth et al., 2010). The robot's Analytics Engine analyses the onboard sensor data to look for anomalies in the process plant (e.g., corrosion, leakages, etc.). An Analytics Engine could be distributed into two parts, 1) an online/real time version running on the robot giving immediate analysis, and 2) an offboard version running "in the cloud" giving analysis results with a delay, and also has central information about "everything" going on in the facility (including information about other robots).

3.4 Simulated Use Case and Results

Three events which may occur during a mission with one or more robots in I&M operations are scheduled missions (e.g., a standard inspection round) and emergency handling missions and other "on-demand missions". System setup description UGV with three charging stations and Robot battery deplete when not charging. Mission description (a) Take EO and IR pictures of seals on valves and pumps at 8 waypoints, Minimum 30 % battery capacity at mission end.

Possible events that could occur during the mission: (1) Operators can add new tags during the mission, (2) Battery becomes lower than expected while the robot is on its way to a waypoint, (3) Higher consumption than initial planned and/or faulty battery reduced max capacity to 50 %, (4) Blocked path is detected by the robot while moving between waypoints, so the path planner finds an alternative route which makes it more efficient to first visit near waypoint and before current waypoint. Table 3 presents the PDDL2.1 problem domain considered for the numerical simulation, presented in this work.

Table 3. Initial state and goal for simulation.

Parameter	PDDL representation
Initial state	(no-seals-check valve0) (no-seals-check valve1) (no-seals-check valve2) (no-seals-check valve3) (no-temperature-check pump0) (no-temperature-check pump1) (no-temperature-check pump2) (at turtlebot0 waypoint0) (full-bat battery0)
Goal	(at turtlebot0 waypoint7) (seals-check valve0) (seals-check valve1) (seals-check valve2) (temperature-check pump0) (temperature-check pump1) (full-bat battery0)
Blocked paths	d12, waypoint 1 - waypoint 2

Simulation results: The numerical simulations were carried out using a hybrid code running, simultaneously, in MATLAB 2021b, Python 3.8.10 and ROS noetic, to share input/outputs between these three programs the functions **pyrun** and **sys** and ROS a publisher/subscriber were used. Thus, the STP planner algorithm is running in a Matlab script, the GNC system is coded in Python and the UGV simulator is running in ROS Gazebo. The whole code is running in a Intel Corei7-7th generation CPU, 2.8GHz and 16GB RAM memory. The computational time to calculate the plan was less than a second for the use case with 8 waypoints. This is a great finding because it will allow the developed real-time application using the STP algorithm.

Table 4 presents the plans calculated for use case. This table presents the initial plan (tasks 1-6a) and the replanned task (task 6b-12). The plan starts at the charger battery station (waypoint 0) and move to the *valve 1* located at waypoint 1. Then, the IR picture of *valve 1* is taken. The plan continues normally, until action 6a, where the system requests a replanning, because the battery is low (30%), and to perform a replanning. This new plan starts with an action move to the closest charger station, *charger 0*. Then the battery is recharged, and the UGV resumes his pendings mission goals. The plan resumed with move to *pump 0* and take EO picture. At the end the last action is to move the robot to the waypoint7.

Table 4. Output of the STP planner including replanning.

n.	T(s)	Action	bat.
1:	0	move turtlebot0 waypoint0 waypoint1 d01	85
2:	63	picture eo turtlebot0 waypoint1 valve0	75
3:	73	move turtlebot0 waypoint1 waypoint3 d13	60
4:	114	picture eo turtlebot0 waypoint3 valve2	50
5:	124	move turtlebot0 waypoint3 waypoint2 d32	35
6(a):	171	picture eo turtlebot0 waypoint2 valve1	25
6(b):	181	move charge waypoint2 waypoint0 d20	-
7:	250	charge battery waypoint0 battery0	100
8:	260	move turtlebot0 waypoint0 waypoint5 d05	85
9:	357	picture ir turtlebot0 waypoint5 pump0	75
10:	369	move turtlebot0 waypoint5 waypoint6 d56	60
11:	471	picture ir turtlebot0 waypoint6 pump1	50
12:	483	move turtlebot0 waypoint6 waypoint7 d67	35

Fig. 6 presents the outputs of the complete system for the use case. Fig. 6a present the initial plan, the plan is composed of 11 tasks. Fig. 6b presents the re-computed plan, this is activated when the battery level is below than 30%, the plan is computed by 7 tasks. Fig. 6c presents a black and white map plot, where it is possible to see the trajectory of the UGV during mission execution, the desired trajectory is plotted in green and the initial point is in red. Fig. 6d shows a shot picture of the ROS gazebo mission execution.

Finally, it is important to remark that although STP can deal with many kinds of temporal problems, such as sequential problems with single hard envelopes and simultaneous events, it is not complete (Furelos Blanco et al., 2018) and does not always provide an optimal solution. The reason for this depends on the number of concurrent actions allowed in its formulation. Thus, STP scales with the number of actions, and as the number of concurrent actions increases, the problem of finding an optimal solution becomes more complex. One strategy to solve this limitation consists of starting with low values of concurrent actions and increasing them while the solution is not found. Moreover, as was mentioned in Cushing et al. (2007), it is important to distinguish between temporal planning and classical planning, as some temporal planners are an extension of classical planners. The STP uses the Fast Downward planning system with lifted temporal states as part of the search nodes which allows concurrent actions. In this preliminary study, simultaneous events were not simulated, but the system is prepared to handle concurrent actions and multi-robot problems. These scenarios will be presented in a future publication, which will include physical experiments with industrial robots.

4. CONCLUSIONS AND FUTURE WORK

A novel approach for inspection and maintenance operations using automated planning for UGVs was presented. The application of automated planning in traditional I&M operations for oil and gas offshore platforms seems to be

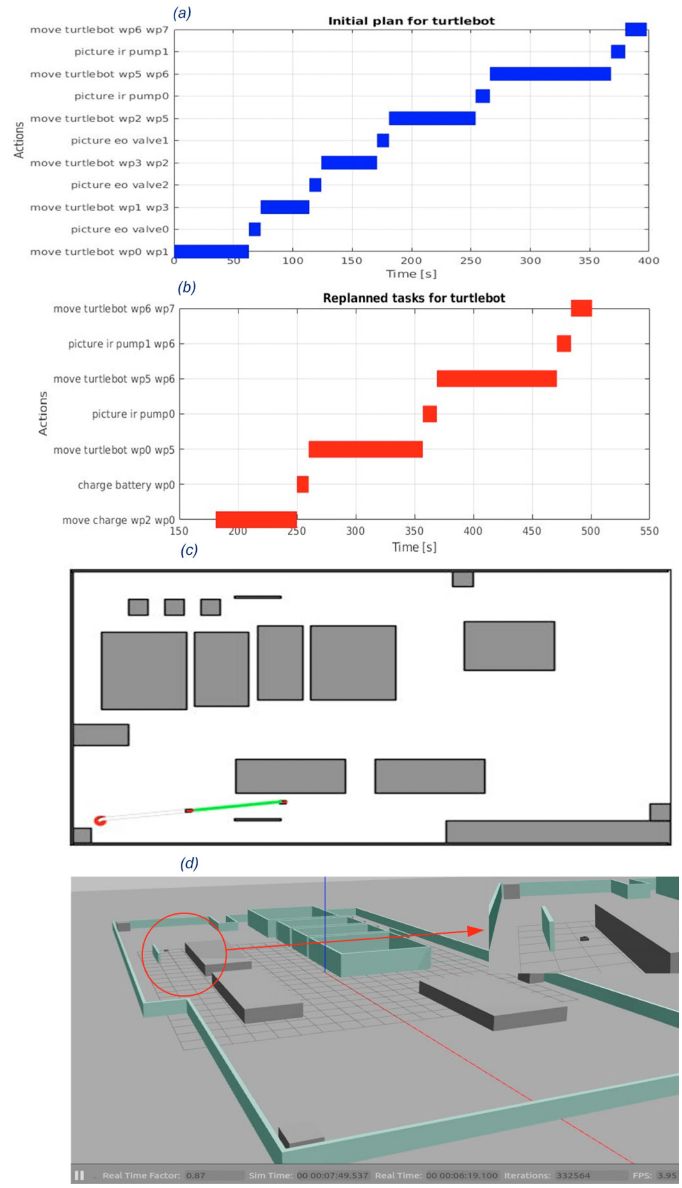

Fig. 6. (a) Initial plan computed using STP planner, (b) Modified plan, (c) GNC execution control, (d) ROS Gazebo mission execution.

positive and significantly increases the range of autonomy and number of potential operations. The temporal STP planner is an adequate method to compute fast and reliable plans for an UGV, navigating in a busy offshore platform environment including several static obstacles. The planner is able to generate sequences of actions (plans) that accomplish the mission goals, and does so fast enough to allow for real-time implementation. The GNC system, composed of a path-planning, path-following and control system has demonstrated a good synergy with the new planning module and has shown a good performance executing a calculated plan. From numerical simulations to good capabilities of the overall system was demonstrated. Additional future work pertains to including more realistic scenarios, with more sensors and tasks, implementation of ROS Simultaneous localization and mapping (SLAM) navigation, for an autonomous problem domain definition, and simulations with multi-robots.

REFERENCES

Barreiro, J., Boyce, M., Do, M., Frank, J., Iatauro, M., Kichkaylo, T., Morris, P., Ong, J., Remolina, E., Smith, T., and Smith, D. (2012). Europa: A platform for ai planning, scheduling, constraint programming, and optimization. *4th International Competition on Knowledge Engineering for Planning and Scheduling (ICKEPS).*

Bengel, M., Pfeiffer, K., Graf, B., Bubeck, A., and Verl, A. (2009). Mobile robots for offshore inspection and manipulation. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3317–3322. IEEE.

Bernard, D., Dorais, G., Gamble, E., Kanefsky, B., Kurien, J., Man, G., Millar, W., Muscettola, N., Nayak, P., and Rajan, K. (1999). Spacecraft autonomy flight experience: The ds1 remote agent experiment.

Blum, A.L. and Langford, J.C. (1999). Probabilistic planning in the graphplan framework. In *European Conference on Planning*, 319–332. Springer.

Cashmore, M., Fox, M., Long, D., Magazzeni, D., Ridder, B., Carrera, A., Palomeras, N., Hurtos, N., and Carreras, M. (2015). Rosplan: Planning in the robot operating system. In *Proceedings of the international conference on automated planning and scheduling*, volume 25, 333–341.

Chen, H., Stavinoha, S., Walker, M., Zhang, B., and Fuhlbrigge, T. (2014). Opportunities and challenges of robotics and automation in offshore oil & gas industry. *Intelligent Control and Automation*, 5, 136-145.

Cushing, W., Kambhampati, S., and Weld, D.S. (2007). When is temporal planning really temporal? In *Proceedings of the 20th international joint conference on Artifical intelligence*, 1852–1859.

Erol, K., Hendler, J., and Nau, D.S. (1994). Htn planning: Complexity and expressivity. In *AAAI*, volume 94, 1123–1128.

Furelos Blanco, D., Jonsson, A., Palacios Verdes, H.L., and Jiménez, S. (2018). Forward-search temporal planning with simultaneous events. In *13th Workshop on Constraint Satisfaction Techniques for Planning and Scheduling, Jun 24-29; Delft, the Netherlands.*

Ghallab, M., Nau, D., and Traverso, P. (2004). *Automated Planning: theory and practice*. Elsevier.

Hinostroza, M.A., Xu, H., and Guedes Soares, C. (2020). Motion Planning, Guidance, and Control System for Autonomous Surface Vessel. *Journal of Offshore Mechanics and Arctic Engineering*, 143(4).

Hinostroza, M. and Lekkas, A.M. (2022). A rudimentary mission planning system for marine autonomous surface ships. *IFAC-PapersOnLine*, 55(31), 196–203.

Nilsson, N.J. et al. (1984). Shakey the robot.

Olsen, S., Lossius, V., Røyrøy, A., Stene, A., and Sagatun, S. (2023). Challenges with offshore robot deployment and ways to solve them. In *SPRINT Robotics Asia-Pacific Summit, Singapore, March 21-23.*

Pinto, J., Sousa, J., Py, F., and Rajan, K. (2012). Experiments with deliberative planning on autonomous underwater vehicles. In *IROS 2012 Workshop on Robotics for Environmental Monitoring, Vilamoura, Portugal.*

Piotrowski, W.M., Fox, M., Long, D., Magazzeni, D., and Mercorio, F. (2016). Heuristic planning for pddl+ domains. In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence.*

Rabideau, G., Knight, R., Chien, S., Fukunaga, A., and Govindjee, A. (1999). Iterative repair planning for spacecraft operations using the aspen system. In *Artificial Intelligence, Robotics and Automation in Space*, volume 440, 99.

Soldan, S., Bonow, G., and Kroll, A. (2012). Robogasinspector-a mobile robotic system for remote leak sensing and localization in large industrial environments: Overview and first results. *IFAC Proceedings Volumes*, 45(8), 33–38.

Solem, E., de Jonge, C., Røyrøy, A., Stene, A., and Sagatun, S. (2023). Anymal x qualification - testing and integration in an onshore gas processing facility. In *SPRINT Robotics Asia-Pacific Summit, Singapore, March 21-23.*

Streich, H. and Adria, O. (2004). Software approach for the autonomous inspection robot makro. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, volume 4, 3411–3416. IEEE.

Tan, J.H., Roberts, B., Sundararaju, P., Sintive, C., Facheris, L., Vanden Bosch, J., Lehning, V., and Pegg, M. (2020). Transforming offshore oil and gas production platforms into smart unmanned installations. In *Offshore Technology Conference Asia*. OnePetro.

Transeth, A.A., Schjølberg, I., Anastasios M. Lekkas, P.R., Ahmed Mohammed, Martin Skaldebø, B.O.A.H.M.B., Tsiourva, M., and Py, F. (2022). Autonomous subsea intervention (seavention). In *14th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles.: DTU Kongens Lyngby, Denmark.*

Transeth, A.A., Skotheim, , Schumann-Olsen, H., Johansen, G., Thielemann, J., and Kyrkjebø, E. (2010). A robotic concept for remote maintenance operations: A robust 3d object detection and pose estimation method and a novel robot tool. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5099–5106.

Xue, L. and Lekkas, A.M. (2020). Comparison of ai planning frameworks for underwater intervention drones. In *Global Oceans 2020: Singapore – U.S. Gulf Coast, Biloxi, MS, USA*, 1–9.