



HAL
open science

MoTie: Monitoring of mobile Things with Intelligent and embedded adaptations for secure services

Mohand Hamadouche, Catherine Dezan, David Espes, Kalinka Regina Lucas
Jaquie Castelo Branco

► To cite this version:

Mohand Hamadouche, Catherine Dezan, David Espes, Kalinka Regina Lucas Jaquie Castelo Branco. MoTie: Monitoring of mobile Things with Intelligent and embedded adaptations for secure services. Compas 2023: Conférence francophone d'informatique en Parallélisme, Architecture et Système, Jul 2023, Annecy, France. hal-04394631

HAL Id: hal-04394631

<https://hal.science/hal-04394631v1>

Submitted on 15 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

MoTie : Monitoring of mobile Things with Intelligent and embedded adaptations for secure services.

Mohand HAMADOUCHE

Supervisors : Catherine Dezan (UBO), David Espes (UBO), Kalinka Branco (USP, Brazil)

Financed by : CDE / SAMM (ARED)

Term of the thesis : Oct. 2019 - 2023

Contact : mohand.hamadouche@univ-brest.fr



I. Context and problem statement

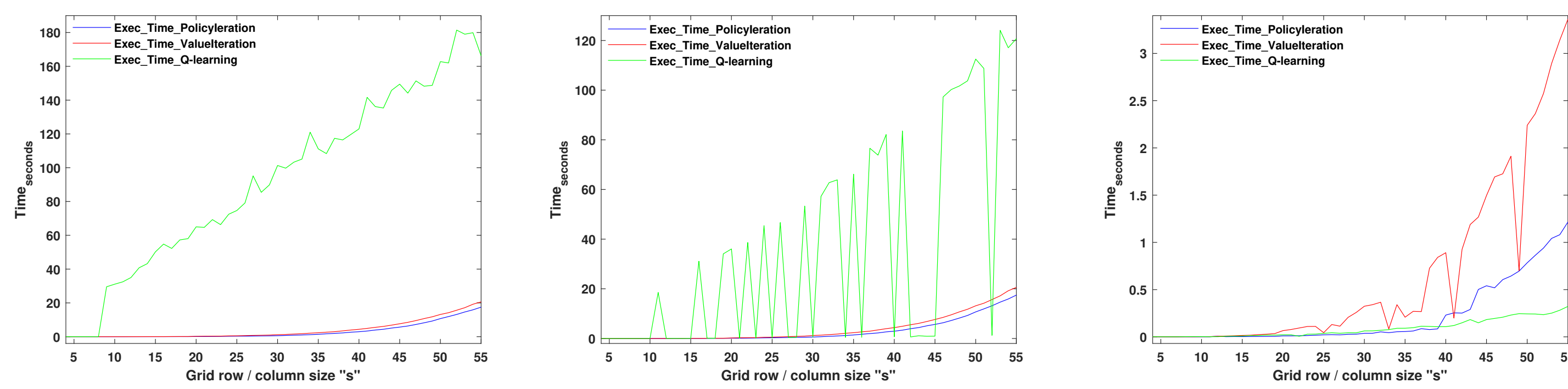
In the context of maritime mission,

- ❖ Cooperation with a UAV or Multi-UAV system improves mission quality, productivity, and safety.
- ❖ The UAVs provide a complementary viewpoint and the maritime drones are used as a 'Mothership'.
- ❖ Aerial drones have to face hazards (malfunctioning of the system/sensors, cyber-attacks e.g. spoofing).
- ❖ Many challenges in multi-UAV systems:
 - ➔ Distributed decision-making, taking into account conflicts;
 - ➔ Failures and safety threats in multi-UAV systems.



II. Comparison of fundamental methods for mission planning [1]

- ❖ Markov decision process (MDP) approach is the most common for modeling UAV missions.
- ❖ Fundamental methods for solving it:
 - ➔ **Value Iteration** and **Policy Iteration** which are Dynamic programming (DP) methods.
 - ➔ **Q-Learning** which is a temporal-difference (TD) method.



Execution time for size 4×4 to 55×55 : (a) Version 1-a. (b) Version 1-b. (c) Version 2.

TABLE III: Summary of 100 executions of the three methods of each version to solve the irregular grid.

100 Executions	Version 1-a			Version 1-b			Version 2			
	Value Iteration	Policy Iteration	Q-Learn.	Value Iteration	Policy Iteration	Q-Learn.	Value Iteration	Policy Iteration	Q-Learn.	
Iterations	$Iter_{min}$	1,375	3	1,000,000	1,375	3	6	1,375	3	11
	$Iter_{max}$	1,375	3	1,000,000	1,375	3	1,000,000	1,375	3	1,000,000
	Average Iter	1,375	3	1,000,000	1,375	3	890,009	1,375	3	855,015
Time	$Exec.time_{min}$	0.03433	0.00050	11.21280	0.03444	0.00032	0.00012	0.03474	0.00032	0.00029
	$Exec.time_{max}$	0.07609	0.16397	14.00078	0.05924	0.01831	10.42652	0.05743	0.02480	10.35899
	Average $Exec.time$	0.03608	0.00232	11.48608	0.03676	0.00081	7.25192	0.03801	0.00091	7.36136
Speed up Q-L / DP Methods		318	4951	1	197	8953	1	194	8089	1

TABLE VI: Summary of 300 executions of the Dynamic Programming methods and the Q-Learning method to solve the decision-making problem of a UAV mission planning.

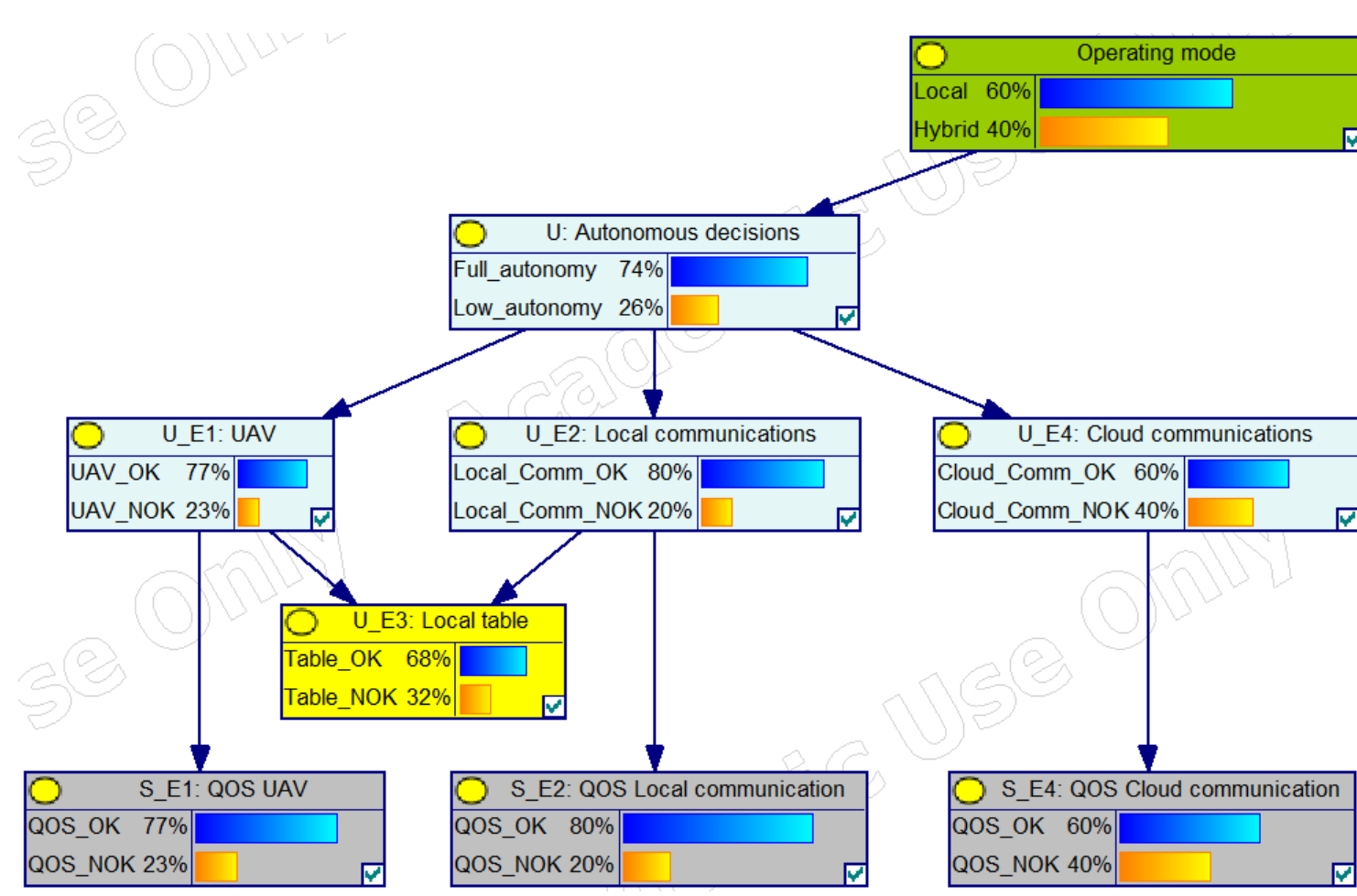
Execution time in seconds			Speed up Q-L / DP Method
Using V.I.	Average(T_{Nav})	0.0018	7.8
			1052
Using P.I.	Average(T_{Nav})	0.00019	73.7
			996.7
Using Q-L	Average(T_{Nav})	0.0140	1
	$iter_{max} = 1,000$		
	$iter_{max} = 100,000$	1.8937	1

Mission planning of a UAV:

- ➔ Sophisticated case with irregular states.
- ➔ Actions are non-similar in each state of the problem.
- ➔ It is more difficult to set up the Q-Learning method, and so, DP methods are more efficient.

IV. Adaptation with cooperation through the cloud/the local network

- ❖ Cooperation within a UAV team can be necessary in case of UAV malfunctioning.
- ❖ Mode switching evaluated by BN:
 - ➔ **Local mode**: local data (+ neighbor data) are sufficient;
 - ➔ **Hybrid mode**: local data are insufficient or communications are compromised.
- ❖ Data used:
 - ➔ QoS UAV
 - ➔ QoS Local communication
 - ➔ QoS Cloud communication



References

- [1] Mohand Hamadouche, Catherine Dezan, David Espes, and Kalinka Branco. Comparison of value iteration, policy iteration and q-learning for solving decision-making problems. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*.
- [2] Mohand Hamadouche, Catherine Dezan, and Kalinka RLJC Branco. Reward tuning for self-adaptive policy in mdp based distributed decision-making to ensure a safe mission planning. In *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pages 78–85. IEEE, 2020.
- [3] Mohand Hamadouche, Catherine Dezan, David Espes, and Kalinka Branco. Online reward adaptation for mdp-based distributed missions. In *2023 ICUAS*.

III. Self adaptation based rewards tuning for mission planning [2, 3]

- ❖ Policy π is a (solution of an MDP) mapping from state space (S) to action space (A).
- ❖ Policy π is used to determine the optimal action given the current state.
- ❖ Change in action reward modifies the action associated with a state.
- ❖ Modification of the policy requires updating the action rewards.
- ❖ **Method principle**:

1. Definition of conflicts and constraints.
2. Definition of priorities (action/UAV).
3. Conflict management:
 - ✳ Reward computation:

$$Q^* = \begin{pmatrix} Q_{s_1, a_1} & Q_{s_1, a_2} & \dots & Q_{s_1, a_m} \\ \vdots & \vdots & \ddots & \vdots \\ Q_{s_n, a_1} & \dots & \dots & Q_{s_n, a_m} \end{pmatrix}$$

$$\pi^*(s) = \arg \max_a Q^*(s, a) \quad \forall s \in S$$

Constraint: can't perform action α in State s_i

Execute another action β ($A \setminus \{\alpha\}$).

$$V(s_i) = \max (Q(s_i, \cdot)^*) = Q(s_i, a_\alpha) \rightarrow V(s_i) = \max (Q(s_i, \cdot)^*) = Q(s_i, a_\beta) \rightarrow Q(s_i, a_\beta) > Q(s_i, a_\alpha)$$

$$Q^*(s, a) = R(s, a) + \gamma \sum_{s' \in S} T(s, a, s') V^*(s') \rightarrow R(s_i, a_\beta) + \gamma \times T(s_i, a_\beta, s') \times V^*(s') > Q(s_i, a_\alpha)$$

$$\rightarrow \text{New Reward } R(s_i, a_\beta) \geq Q(s_i, a_\alpha) - \gamma \times T(s_i, a_\beta, s') \times V^*(s') \rightarrow \text{New Policy } \pi^*$$

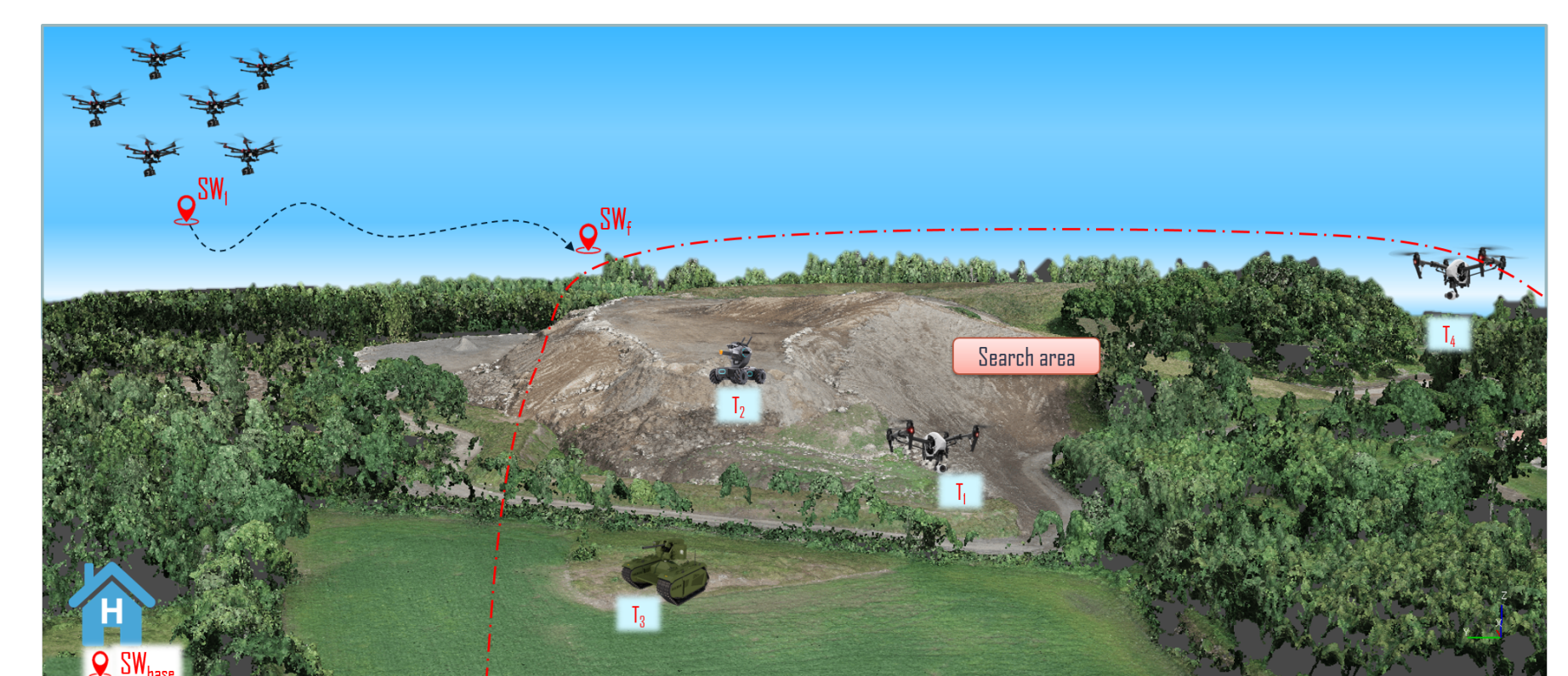
(a) **UAV level**:

- ➔ Executing simultaneous actions can lead to conflicts.
- ➔ Identification of conflicting states with antagonistic actions.
- ➔ Action-states with conflicts are adjusted by modifying the rewards of conflicting actions.

(b) **Multi-UAV level**:

- ➔ UAV decisions are unique and optimal at their level.
- ➔ Decisions can be contradictory and do not meet the mission's objectives of the mission.

- ➔ The UAV favors an alternative action to solve the conflict in the team.
- ➔ The method is efficient in terms of both latency ($\mu(\text{time})=2\text{ms}$) and energy consumption.



V. Conclusion

- ❖ Contributions:
 - ➔ Comparative study of methods for solving MDPs for mission planning.
 - ➔ Proposal of a systematic method for resolving conflicts at UAV and Multi-UAV team level.
 - ➔ Simulation of conflict resolution scenarios using the CoppeliaSim simulator.
 - ➔ Proposal of a method for adapting the UAV operating mode for conflict resolution by using the Cloud or the local network.

