Multi-agent reinforcement learning for wind farm control

Claire Bizon Monroc, Ana Bušić, Donatien Dubuc, and Jiamin Zhu

Plan

1. Wind farm power production optimization
2. Delay-aware MARL algorithms for the Wind Farm Control Problem
3. Convergence analysis
4. WFCR benchmark for WFCP
5. Conclusion

Wind farm power production optimization

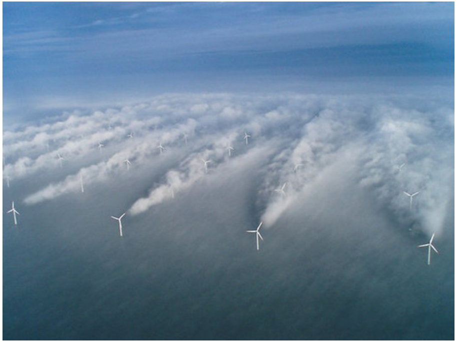


Figure: Horns rev offshore wind farm, Vattenfall, 2008

Wake effects

- Upstream turbines create sub-optimal wind conditions for downstream turbines.
- This decreases the total amount of power produced.

Wake steering with yaw control

- Yaw: angle between the rotor plane and the direction of the incoming wind
- Increasing the yaw of an upstream turbine deflects its wake away from downstream turbines.

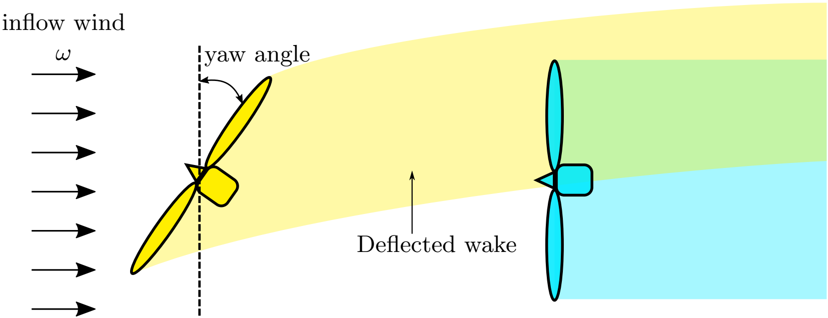


Figure: Wake deflection with yaw control

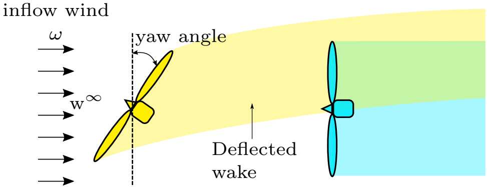
WFCP: Wind Farm Control Problem

Goal

Maximize the total power output of a wind farm with M turbines

Controls

Yaws $\gamma = (\gamma^1, \ldots, \gamma^M)$



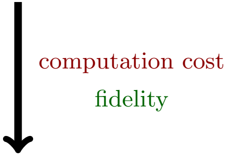
Measurements

- w^∞: free-stream wind conditions (direction ϕ^∞ and speed u^∞)
- $\mathcal{P}_{1,t} \ldots, \mathcal{P}_{M,t}$ individual productions at any time t
- $\mathcal{P}_{farm,t} = \sum_{i=1}^{M} \mathcal{P}_{i,t}$ total power output

Challenges

Wind farm simulators

- Steady-state models: FLORIS (NREL)
- 2D Navier-Stokes: WFSim (TUDelft)
- Dynamic Wake Meandering model: FAST.Farm (NREL)
- Large eddy simulations: SOWFA (NREL)

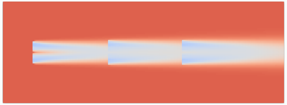


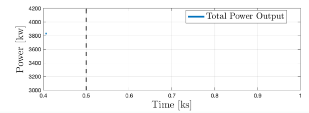
Challenges

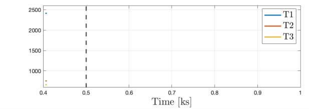
Modelling errors

- Steady-state models
- 2D Navier-Stokes
- Dynamic Wake Meandering model
- Large eddy simulations

Wake propagation







Scaling to large farms

Yaw optimization

Static Problem

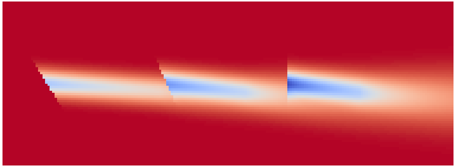
- Wind conditions are constant in time

$$\max_{\gamma} \mathcal{P}_{farm}$$

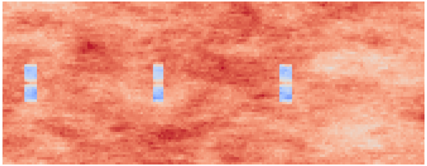
Dynamic Problem

- Wind conditions change at every time-step $(0 < \beta < 1)$

$$\max_{\gamma_0,\dots,\gamma_\infty} \sum_{k=0}^{\infty} \beta^k \mathcal{P}_{farm,k}$$



(a) Static simulation: FLORIS (NREL)

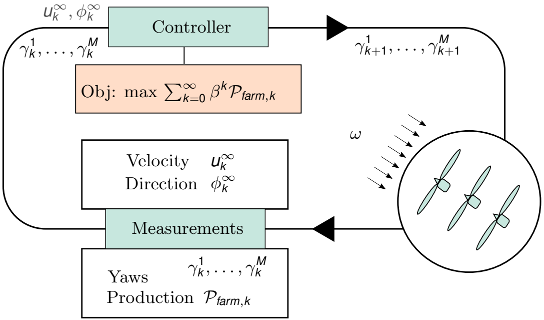


(b) Dynamic simulation: FAST.Farm (NREL)

A model-free approach

Adapt policies in the real system !

Design data-driven methods that learn from observing control inputs and output measurements collected on the wind farm.



$u_k^\infty, \phi_k^\infty$

Controller

$\gamma_k^1, \dots, \gamma_k^M$

$\gamma_{k+1}^1, \dots, \gamma_{k+1}^M$

Obj: max $\sum_{k=0}^\infty \beta^k \mathcal{P}_{farm,k}$

ω

Velocity u_k^∞
Direction ϕ_k^∞

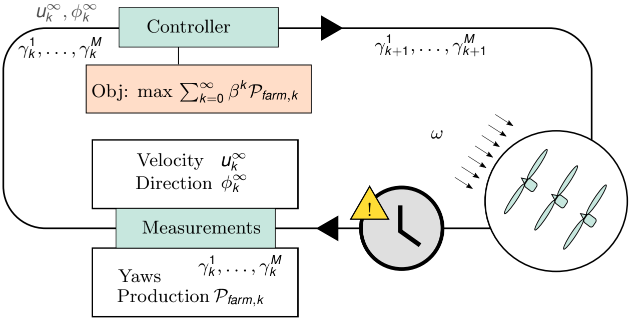
Measurements

Yaws $\gamma_k^1, \dots, \gamma_k^M$
Production $\mathcal{P}_{farm,k}$

... that is robust to wake propagation dynamics

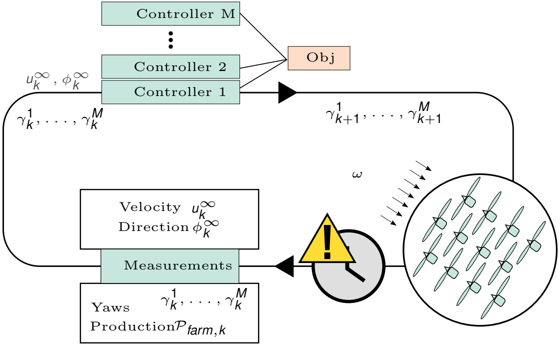
Account for wake propagation times

Wake propagation dynamics create a time delay between a change of yaws and a measurable impact on the energy production of the wind farm.



$u_k^\infty, \phi_k^\infty$
Controller
$\gamma_k^1, \dots, \gamma_k^M$
$\gamma_{k+1}^1, \dots, \gamma_{k+1}^M$
Obj: max $\sum_{k=0}^{\infty} \beta^k \mathcal{P}_{farm,k}$
ω
Velocity u_k^∞
Direction ϕ_k^∞
!
Measurements
Yaws $\gamma_k^1, \dots, \gamma_k^M$
Production $\mathcal{P}_{farm,k}$

A cooperative multi-agent reinforcement learning (MARL) approach



Controller M
Obj
Controller 2
Controller 1
$u_k^\infty, \phi_k^\infty$
$\gamma_k^1, \ldots, \gamma_k^M$
$\gamma_{k+1}^1, \ldots, \gamma_{k+1}^M$
ω
Velocity u_k^∞
Direction ϕ_k^∞
Measurements
Yaws $\gamma_k^1, \ldots, \gamma_k^M$
Production$\mathcal{P}_{farm,k}$

MARL approach

Decentralized learning of local policies:

- M agents
- 1 shared reward
- Each agent i receives a partial observation of the system and learns a policy π^i

Used in static and quasi-static simulations (Graf et al. 2019; Xu et al. 2020; Stanfel et al. 2021)

Contributions

- Delay-aware MARL algorithms for the WFCP
- Exploiting model knowledge with imitation
- Theoretical analysis of independent learners
- A multi-agent RL benchmark environment

Delay-aware MARL algorithms for the WFCP

WFCP as a delayed problem

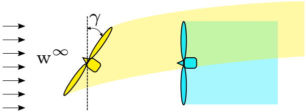
- WFCP as a Delayed Dec-MDP
- Local reward functions based on delay estimation for decentralized learning

Design delay-aware independent learners

- Independent learners for the dynamic WFCP
- Evaluation on FAST.Farm and WFSim with both turbulent and changing wind conditions

1. Bizon Monroc, Bouba, Bušić, Dubuc, and Zhu Delay-aware decentralized q-learning for wind farm control., CDC 2022
2. Bizon Monroc, Bušić, Dubuc, and Zhu Actor critic agents for wind farm control., ACC 2023

WFCP as a delayed Dec-MDP problem



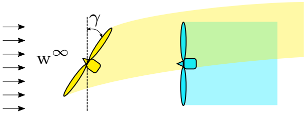
The multi-agent WFCP

Objective:

$$\max_{\pi^1,\dots,\pi^M} \mathbb{E}\left[\sum_{k=0}^{\infty} \beta^k \sum_{i=1}^{M} P_{i,k}\right]$$

- S : full state space (unobserved)
- $\Omega_{i,1 \leq i \leq M}$: M observation spaces with γ^i and w^∞
- $A_{i,1 \leq i \leq M}$: M action spaces representing the change in local yaw $\Delta\gamma^i$ between two time-steps
- $r : S \times A \times S \to [0, R]$: shared reward function

WFCP as a delayed Dec-MDP problem



The multi-agent WFCP : a delayed approach

Objective:

$$\max_{\pi^1,\dots,\pi^M} \mathbb{E}\left[\sum_{k=0}^{\infty} \beta^k \sum_{i=1}^{M} P_{i,k}\right]$$

- $S_{i,1\leq i\leq M}$: M local state spaces with γ^i and w^{∞}
- $S = \Pi_i^M S_i$: global space as factorization of local spaces
- $A_{i,1\leq i\leq M}$: M action spaces representing the change in local yaw $\Delta\gamma^i$ between two time-steps
- $r : S \times A \times S \rightarrow [0, R]$: shared reward function
- A delay $d \in \mathbb{N}$: the number of time-steps after which the reward can be collected

WFCP as a delayed Dec-MDP problem

The multi-agent WFCP: a delayed approach with local rewards

Objectives:

$$\max_{\pi^1} \mathbb{E}\left[\sum_{k=0}^{\infty} \beta^k \sum_{i=1}^{M} r_k^1\right] \quad \dots \quad \max_{\pi^M} \mathbb{E}\left[\sum_{k=0}^{\infty} \beta^k \sum_{i=1}^{M} r_k^M\right]$$

- $S_{i,1\leq i\leq M}$: S_i: M local state spaces with γ^i and w^∞
- $S = \Pi_i^M S_i$: global space as factorization of local spaces
- $A_{i,1\leq i\leq M}$: M action spaces representing the change in local yaw $\Delta\gamma^i$ between two time-steps
- $r^i : S_i \times A_i \times S_i \rightarrow [0, R]$: M local reward functions
- M delays : the number of time-steps after which each local reward can be collected

Delay-aware MARL algorithms for the WFCP

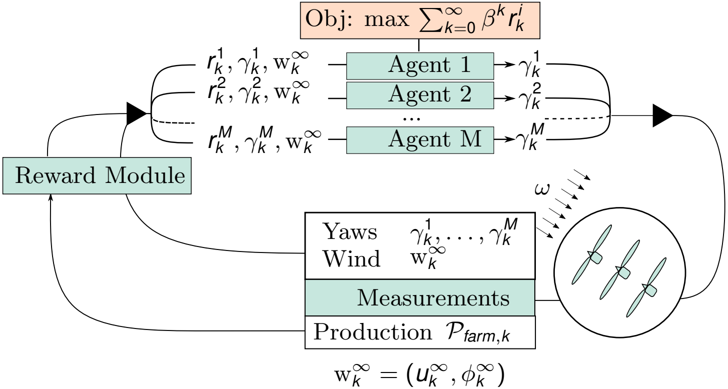
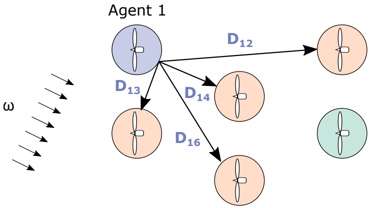

Obj: max $\sum_{k=0}^{\infty} \beta^k r_k^i$
$r_k^1, \gamma_k^1, \mathrm{w}_k^\infty$
$r_k^2, \gamma_k^2, \mathrm{w}_k^\infty$
$r_k^M, \gamma_k^M, \mathrm{w}_k^\infty$
Agent 1
Agent 2
...
Agent M
γ_k^1
γ_k^2
γ_k^M
Reward Module
ω
Yaws $\gamma_k^1, \dots, \gamma_k^M$
Wind w_k^∞
Measurements
Production $\mathcal{P}_{farm,k}$
$\mathrm{w}_k^\infty = (u_k^\infty, \phi_k^\infty)$

Figure: WFCP-MARL: our multi-agent formulation of the dynamic WFCP as a Delayed MDP

Local reward functions



Delay Matrix

	D_{12}	D_{13}	D_{14}		D_{16}	Agent 1
			D_{34}		D_{36}	
	D_{42}			D_{45}	D_{46}	
				D_{65}		Agent 6

Delays approximated with Taylor's frozen wake hypothesis (Taylor 1938):

$$D_{ij} \propto \frac{\text{distance}(i \to j)}{u_\infty}$$

Reward functions

$$r_{i,k} = \begin{cases} 1 & \text{if} \quad \frac{V_2^i - V_1^i}{V_1^i} > \Delta \\ -1 & \text{if} \quad \frac{V_2^i - V_1^i}{V_1^i} \leq -\Delta \\ 0 & \text{otherwise} \end{cases}$$

$$\text{with} \quad V_1^i = \sum_{j=1}^{M} P_{j,k} \quad V_2^i = \sum_{j=1}^{M} P_{j,k+D_{i,j}}$$

$$\Delta > 0$$

Empirical results

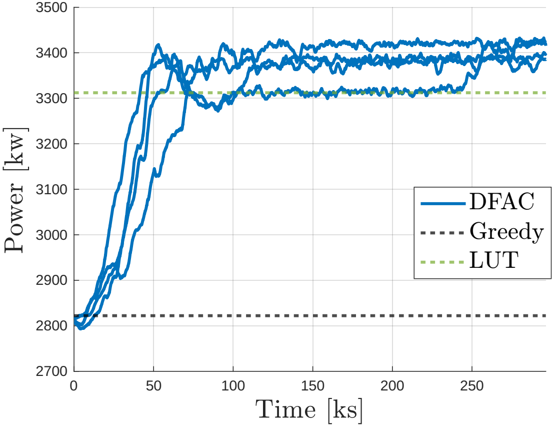


Figure: Average energy produced during 1h

DFAC: Delay-Aware Fourier Actor Critic Agents

- Row of 3 turbines
- Turbulent stationary wind inflow
- Turbulence intensity: 8%
- Average wind speed: 8 m/s
- Simulated on FAST.Farm

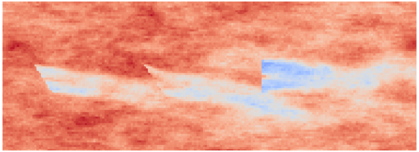


Figure: Row of 3 turbines simulated in FAST.Farm

Empirical results

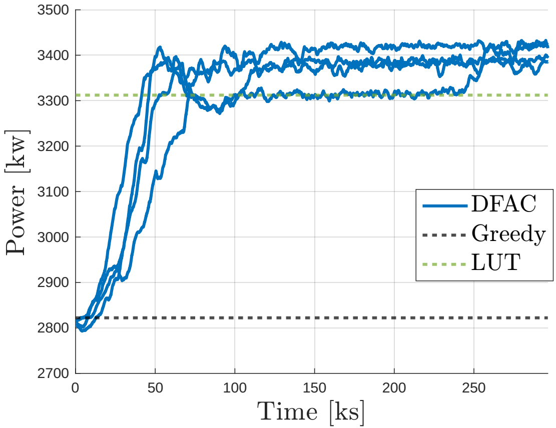


Figure: Average energy produced during 1h

Scaling to larger wind farms:

Layout	DFAC Energy [MWh]	Over baseline (%)	Rise Time (ks)
Layout 1 (3T)	9.34	45.83	22
Layout 3 (7T)	23.02	9.35	38
Layout 4 (16T)	31.44	10.99	302
Layout 5 (32T)	47.90	37.02	-

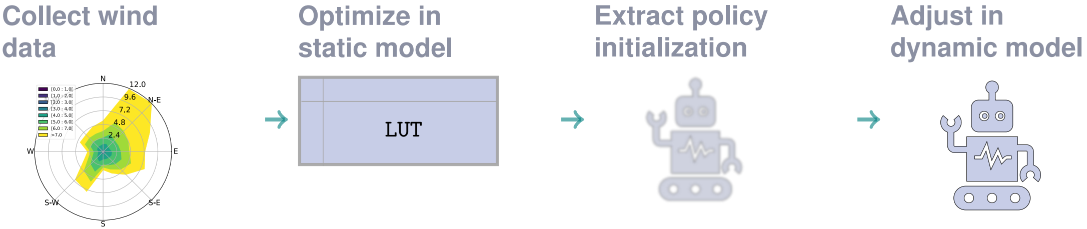
Table: DFAC on WFSim: scaling experiments on layouts with 3, 7, 16 and 32 turbines. Results after 200ks (Layout 1, 3) and 600ks (Layout 4, 5).

Exploiting model knowledge with imitation

Imitate optimal policies from model-based optimization

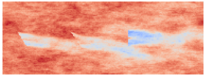
- Use optimal policies learned with static simulator FLORIS to guide online learning with dynamic simulator FAST.Farm
- Evaluation on WFSim and FAST.Farm under both changing and turbulent wind conditions

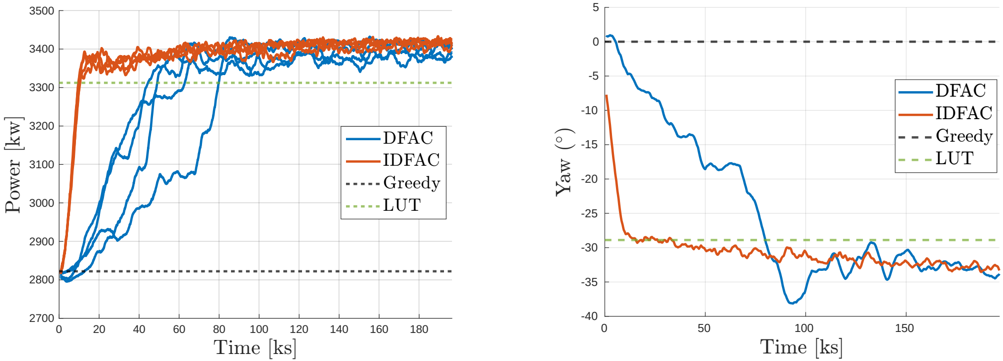
Bizon Monroc, Bušić, Dubuc, and Zhu, Towards fine tuning wake steering policies in the field: an imitation-based approach., TORQUE 2024



Collect wind data → Optimize in static model → Extract policy initialization → Adjust in dynamic model

Imitation



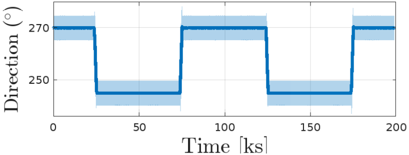


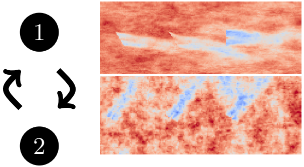
(a) Power output

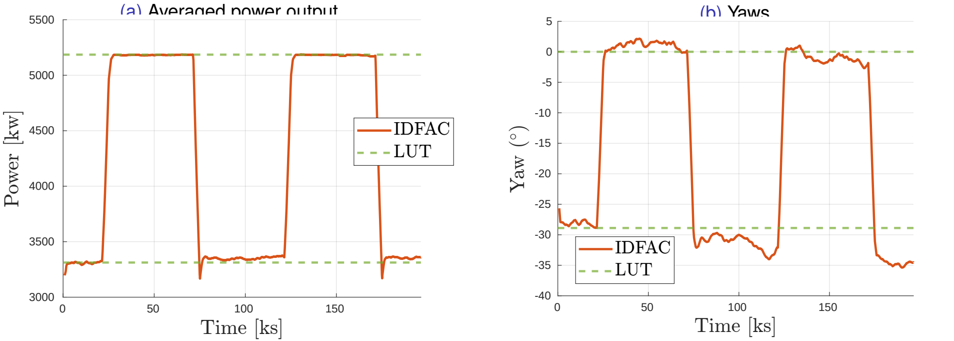
(b) Yaw of the first turbine on 1 run

Figure: DFAC and IDFAC algorithms - 1h average of total power output (a) and yaws during the first experiment of each algorithm (b) on 56h of simulation.

Imitation: turbulent wind step






(a) Averaged power output

(b) Yaws

Towards theoretical guarantees

The WFCP without delay is a Transition-Independent Dec-MDP

- No-delay WFCP can be framed as a transition independent Dec-MDP (Becker et al. 2004; Allen and S. Zilberstein 2009)
- ... where the interdependence of agents dynamics can be represented by a directed acyclic graph (DAG).

A multi-scale approach for independent learners

- Multi-scale approach and learning rates attribution procedure
- Independent Q-learners converge (under assumptions) to an equilibrium of best responses
- Proof is based on results from the stochastic approximation theory (Borkar 1997; Borkar and Meyn 2000; Kushner and Yin 2003)

Bizon Monroc, Bušić, Dubuc, and Zhu Multi-agent reinforcement learning for partially observable cooperative systems with acyclic dependence structure., ARLET Workshop at ICML 2024

Transition Independent Dec-MDP

TI Dec-MDP

The transition kernel can be decomposed into M local transition functions $P^i : S_i \times A_i \times S_i \to [0,1]$:

$$P(s,a,s') = \Pi_i P^i(s^i, a^i, s'^i)$$

- d^π: invariant distribution over S under π
- $0 < \beta < 1$: discount factor

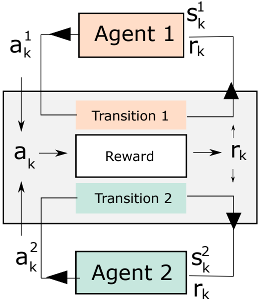


Figure: TI Dec-MDP

Local q-values $Q^{\pi^{-i}}_{\pi^i}(s^i, a^i) = \mathbb{E}_{s_0 \sim d^\pi, a_k \sim (\pi^i, \pi^{-i}), s_k \sim P}\left[\sum_{k=0}^{\infty} \beta^k r(s_k, a_k) \mid s^i_0 = s^i, a^i_0 = a^i\right]$

Wind farm control as a Transition Independent Dec-MDP

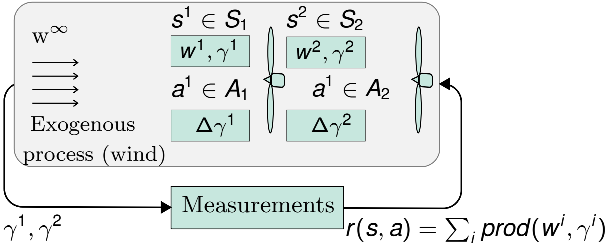
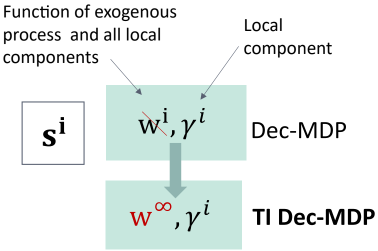
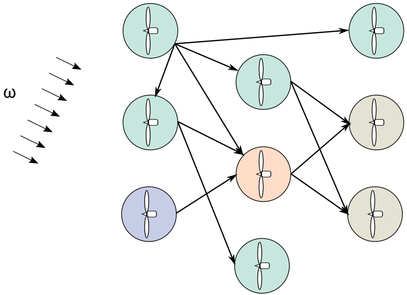


Figure: The multi-agent wind farm control loop



Agents have a DAG interaction structure



There is a directed acyclic graph (DAG) $\mathcal{G}$ such that for every agent i in $\mathcal{G}$ with ancestors $\mathcal{NA}(i)$ and descendants $\mathcal{ND}(i)$, if:

1. Policies of agents in $\mathcal{NA}(i)$ stay fixed
2. i changes its policy
3. Agents in $\mathcal{ND}(i)$ sequentially adapt their policies

the change in reward is bounded by the change in i's policy.

Learning Rates Attribution

All M agents locally update their q-values estimates according to iterates:

$$\hat{Q}^i_{k+1,c} = \hat{Q}^i_{k,c} + \alpha^i_k(s^i_k, a^i_k)\left[\bar{r}^i_k + \beta[\phi(\hat{Q}^i_k)(s^i_{k+1})]^T \hat{Q}^i_k(s^i_{k+1}, \cdot) - \hat{Q}^i_{k,c}\right]$$

with neighbors $\mathcal{N}(i)$, $\bar{r}^i_k = \sum_{j \in \{i, \mathcal{N}(i)\}} r^j(s^j_k, a^j_k, s^{U^j}_k, a^{U^j}_k)$, $\phi(Q^i)(\cdot|s^i) \propto exp(\frac{Q^i(s^i,a^i)}{\tau})$

Learning Rates Attribution

For any node:

- (A) Ancestors have strictly inferior ranks
- (B) Ancestors have strictly different ranks between themselves

Learning Rates Attribution

All M agents locally update their q-values estimates according to iterates:

$$\hat{Q}^i_{k+1,c} = \hat{Q}^i_{k,c} + \alpha^i_k(s^i_k, a^i_k)\left[\bar{r}^i_k + \beta[\phi(\hat{Q}^i_k)(s^i_{k+1})]^T \hat{Q}^i_k(s^i_{k+1}, \cdot) - \hat{Q}^i_{k,c}\right]$$

with neighbors $\mathcal{N}(i)$, $\bar{r}^i_k = \sum_{j \in \{i, \mathcal{N}(i)\}} r^j(s^j_k, a^j_k, s^{U^j}_k, a^{U^j}_k)$, $\phi(Q^i)(\cdot|s^i) \propto exp(\frac{Q^i(s^i, a^i)}{\tau})$

Learning Rates Attribution

For any node:

- (A) Ancestors have strictly inferior ranks
- (B) Ancestors have strictly different ranks between themselves

Theorem

The q-value estimates $\hat{Q}^i_{k+1}$ will converge towards the smoothed best-responses for local q-values $Q^{\pi^{-i}}_{\pi^i}$.

Experiments

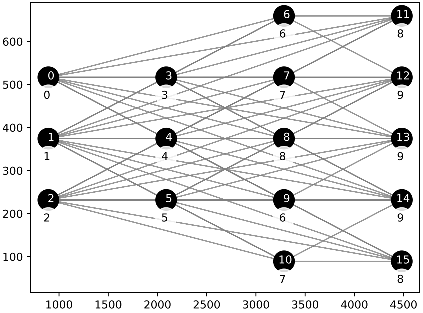


Figure: Agent interaction DAG.

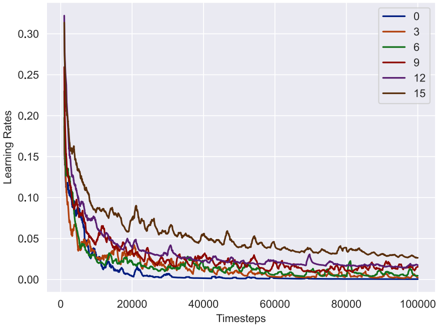


Figure: Evolution of learning rates

Experiments

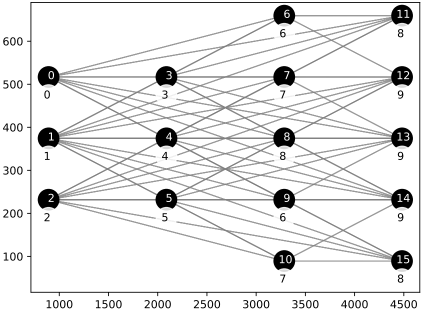


Figure: Agent interaction DAG.

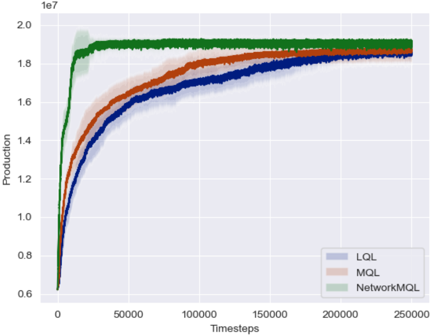


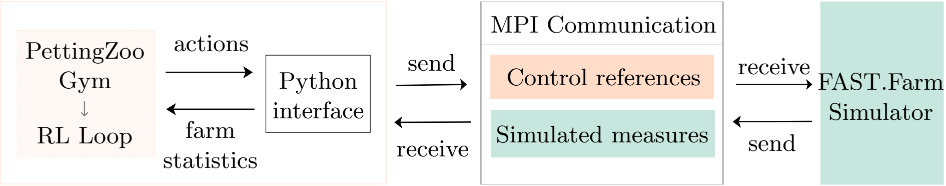
Figure: Evolution of power output

WFCRL: MARL environments for the WFCP

Bridging the wind energy and reinforcement learning communities

- Interfaces any control algorithm with FLORIS and FAST.Farm
- MARL environments with real-world layouts
- Benchmark example for both production maximization and load minimization

Bizon Monroc, Bušić, Dubuc, Zhu, NeurIPS 2024 Datasets and Benchmarks Track
`https://github.com/ifpen/wfcrl-benchmark`



Conclusion

- Wind farm control as a delayed MARL probem
- Experimentally validated algorithms on state-of-the-art dynamic simulators
- Use of static models to guide learning in dynamic conditions
- A convergence analysis of independent learners in a certain class of Dec-MDP
- A new benchmark and interfacing library to bridge RL and wind energy communities

Perspectives

Towards applications on real systems

- Validation on high-fidelity simulations
- Considering loads in a constrained Dec-POMDP
- Tracking a production signal for integration in the grid

Multiscale Q-learning: interdependence dynamics and delays

- Convergence with delayed updates
- Convergence under a single time-scale
- Gap between equilibrium and optimal policy ?

WFCRL: future developments as an open-source library

Based on publications

1. Bizon Monroc C., Bouba E., Bušić A., Dubuc D., and Zhu J. Delay-aware decentralized q-learning for wind farm control. In 2022 IEEE 61st Conference on Decision and Control (CDC).
2. Bizon Monroc C., Bušić A., Dubuc D., and Zhu J. Actor critic agents for wind farm control. In 2023 American Control Conference (ACC).
3. Bizon Monroc C., Bušić A., Dubuc D., and Zhu J. Towards fine tuning wake steering policies in the field: an imitation-based approach.. TORQUE 2024.
4. Bizon Monroc C., Bušić A., Dubuc D., and Zhu J. Multi-agent reinforcement learning for partially observable cooperative systems with acyclic dependence structure.. Presented at ARLET Workshop, ICML 2024
5. Bizon Monroc C., Bušić A., Dubuc D., and Zhu J. WFCRL: A Multi-Agent Reinforcement Learning Benchmark for Wind Farm Control, NeurIPS 2024 Datasets and Benchmarks Track

Deposits

Patent
Bouba E., Dubuc D., Zhu J., Bizon Monroc C. and Bušić A., (2023). Method of controlling a wind farm using a reinforcement learning method, Patent Application FR3142782A1, French Patent Office.

Code deposits

- WFCRL - Wind Farm Control Reinforcement Learning (Python library).
  Deposit N°: IDDN.FR.001.250031.000.S.C.2024.000.30705
- Farm2Python - Interfacing tool for research in wind farm control.
  Deposit N°: IDDN.FR.001.250029.000.S.C.2024.000.30705

Bibliography I

- Graf, Peter et al. (2019). “Distributed Reinforcement Learning with ADMM-RL”. In: *2019 American Control Conference (ACC)*, pp. 4159–4166.
- Stanfel, Paul et al. (Aug. 2021). “Proof-of-concept of a reinforcement learning framework for wind farm energy capture maximization in time-varying wind”. In: *Journal of Renewable and Sustainable Energy* 13.4.
- Taylor, G. I. (1938). “The Spectrum of Turbulence”. In: *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences* 164.919, pp. 476–490.
- Xu, Zhiwei et al. (2020). “Model-Free Optimization Scheme for Efficiency Improvement of Wind Farm Using Decentralized Reinforcement Learning”. In: *IFAC-PapersOnLine* 53.2. 21st IFAC World Congress, pp. 12103–12108.

Thank you.

Appendix I: Imitation-based DFAC (IDFAC)

For any LUT mapping wind conditions to yaws $\mathcal{M}_{Farm} : (\mathrm{w}) \rightarrow (\gamma_1^{\mathcal{M}}, \ldots, \gamma_M^{\mathcal{M}})$.

$$\forall s_i = (\gamma_i, \mathrm{w}) \in S_i$$

$$\pi^{i\mathcal{M}}(s_i) = \begin{cases} \min(\Delta, \mathcal{M}_{Farm}(\mathrm{w})_i - \gamma_i) & \text{if } \gamma_i < \mathcal{M}_{Farm}(\mathrm{w})_i \\ \max(-\Delta, \gamma_i - \mathcal{M}_{Farm}(\mathrm{w})_i) & \text{if } \gamma_i > \mathcal{M}_{Farm}(\mathrm{w})_i \\ 0 & \text{otherwise} \end{cases}$$

$$V^{i\mathcal{M}}(s_i) = r_{ub} \sum_{k=0}^{T-1} \beta^k = r_{ub} \frac{1-\beta^T}{1-\beta} \quad \text{where } T = \left\lceil \frac{|\mathcal{M}_{Farm}(\mathrm{w})_i - \gamma_i|}{\Delta} \right\rceil$$

with Δ the upper bound on absolute change in the yaw command at every iteration.

Imitation-based DFAC (IDFAC)

Methodology

1. Generate a training dataset from the model-derived optimal policy and value functions $\pi^{i\mathcal{M}}$ and $V^{i\mathcal{M}}$
2. Offline: learn parameterized functions V_ν and π_θ approximating $\pi^{i\mathcal{M}}$ and $V^{i\mathcal{M}}$

$$\mathcal{L}_\theta = \sum_{i=0}^{N} \left[\pi_\theta(x_i) - \pi^{\mathcal{M}}(x_i)\right]^2, \qquad \mathcal{L}_{\nu_1} = \sum_{i=0}^{N} \left[V_\nu(x_i) - V^{\mathcal{M}}(x_i))\right]^2.$$

3. Offline: adjust estimate by minimizing the TD error with rewards estimated under stochastic policies

$$\delta(s, r, s') = r(s, a) + \beta V_\pi(s') - V_\pi(s), \qquad \mathcal{L}_{\nu_2} = \delta(s, r, s')^2$$

4. Online: initialize DFAC in the dynamic environment with the policies learned offline

Imitation: WFSim experiment on wind time series

- Farm of 7 turbines: layout inspired from Ablaincourt wind farm in France
- Non-turbulent stationary wind inflow
- Average wind speed: 10 m/s
- Simulated on WFSim

Method	Energy [MWh]	Over greedy baseline(%)
FLORIS	29.07	-4.91
Imitation only	29.55	-3.36
Imitation + Online Learning	31.32	2.43

Table: Average energy produced during 1h on the tracking experiment with LUT and pre-trained Fourier Actor-Critic on a 7 turbine layout

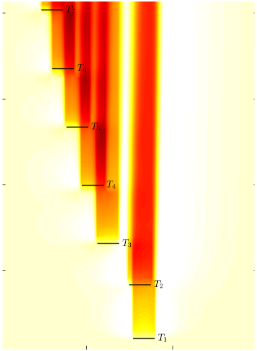


Figure: 7 turbines layout

Appendix II: Transition Independent Dec-MDP

TI Dec-MDP

A Transition Independent Decentralized Markov Decision Process (TI Dec-MDP) is a tuple $\{M, S, A, P, R\}$ with M agents and:

- $S = S_1 \times \cdots \times S_M$ a finite state space
- $A = A_1 \times \cdots \times A_M$ a finite action space
- R the reward function shared by all agents
- P the transition kernel

such that the transition kernel can be decomposed into M local transition functions $P^i : S_i \times A_i \times S_i \rightarrow [0, 1]$:

$$P(s, a, s') = \Pi_i P^i(s^i, a^i, s'^i)$$

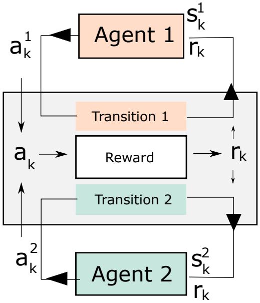


Figure: TI Dec-MDP

An important assumption on structure

Notations: $Q^{>i} = (Q^{i+1}, \ldots, Q^M) \quad Q^{<i} = (Q^1, \ldots, Q^{i-1})$.

Agent interdependence dynamics

For π, d^π with $\phi(Q) \cdot d^\pi$ the corresponding stationary distribution over global state-action pairs. For any M-uplets Q there is an associated reward expectation taken over the stationary distribution of state-action pairs. Let us take a set of q-tables Q with its corresponding global policy $\pi = \phi(Q)$. For any agent i:

- For $j \leq i$, Q^j is any q-table in $S_j \times A_j$
- For $j > i$, Q^j is a q-table of the smoothed best response to π^{-j} as introduced in. We write $Z^{\geq i+1}(Q^{<i+1})$ the $M - i$ q-tables $Q^{>i}$ thus defined.

Let $Q'^i \in [-D, D]^{|S_i| \times |A_i|}$ be a local perturbation to Q^i within the constraint set. Write $Q' = (Q^{<i}, Q'^i, Z^{\geq i+1}(Q^{<i}, Q'^i))$ and $\pi' = \phi(Q')$. There exists an ordering of agents $\{1, \ldots, M\}$ and $K \in (0, 1)$ such that for every agent i and its q-table Q^i, the reward function satisfies:

$$\|R_\pi(s) - R_{\pi'}(s)\|_1 \leq K\|Q^i - Q'^i\|_\infty$$

Convergence of local iterates

Let now all agents locally run a Q-learning update,

$$\begin{aligned}\hat{Q}^i_{k+1}(s^i, a^i) &= \hat{Q}^i_k(s^i, a^i) \\ &\quad + \alpha^i_k(s^i_k, a^i_k)\left[r_k + \beta[\phi(\hat{Q}_k)(s^i_{k+1})]^T \hat{Q}_k(s^i_{k+1}, \cdot) - \hat{Q}^i_k(s^i_k, a^i_k)\right] I_{k,s^i,a^i}\end{aligned} \tag{1}$$

with I_{k,s^i,a^i} the indicator of the event that the local state-action pair s^i, a^i is visited at timestep k.

Theorem

With proper initialization and:

- *Initial values $\hat{Q}^i_0 \in [-D, D]$ for $D > 0$ such that $D > \frac{R}{\beta}$*
- *Carefully chosen learning rate sequences*
- *A discount factor β satisfies $\beta \leq 1 - K$*
- ***Specific assumption on agent interdependence dynamics***

The q-value estimates $\hat{Q}^i_{k+1}$ will converge towards the smoothed best-response of q-values

$$Q^{\pi^{-i}}_{\pi^i}(s^i, a^i) = \mathbb{E}_{s_0 \sim d^{\pi}, a_k \sim (\pi^i, \pi^{-i}), s_k \sim P}\left[\sum_{k=0}^{\infty} \beta^k r(s_k, a_k) \mid s^i_0 = s^i, a^i_0 = a^i\right]$$