MARIN

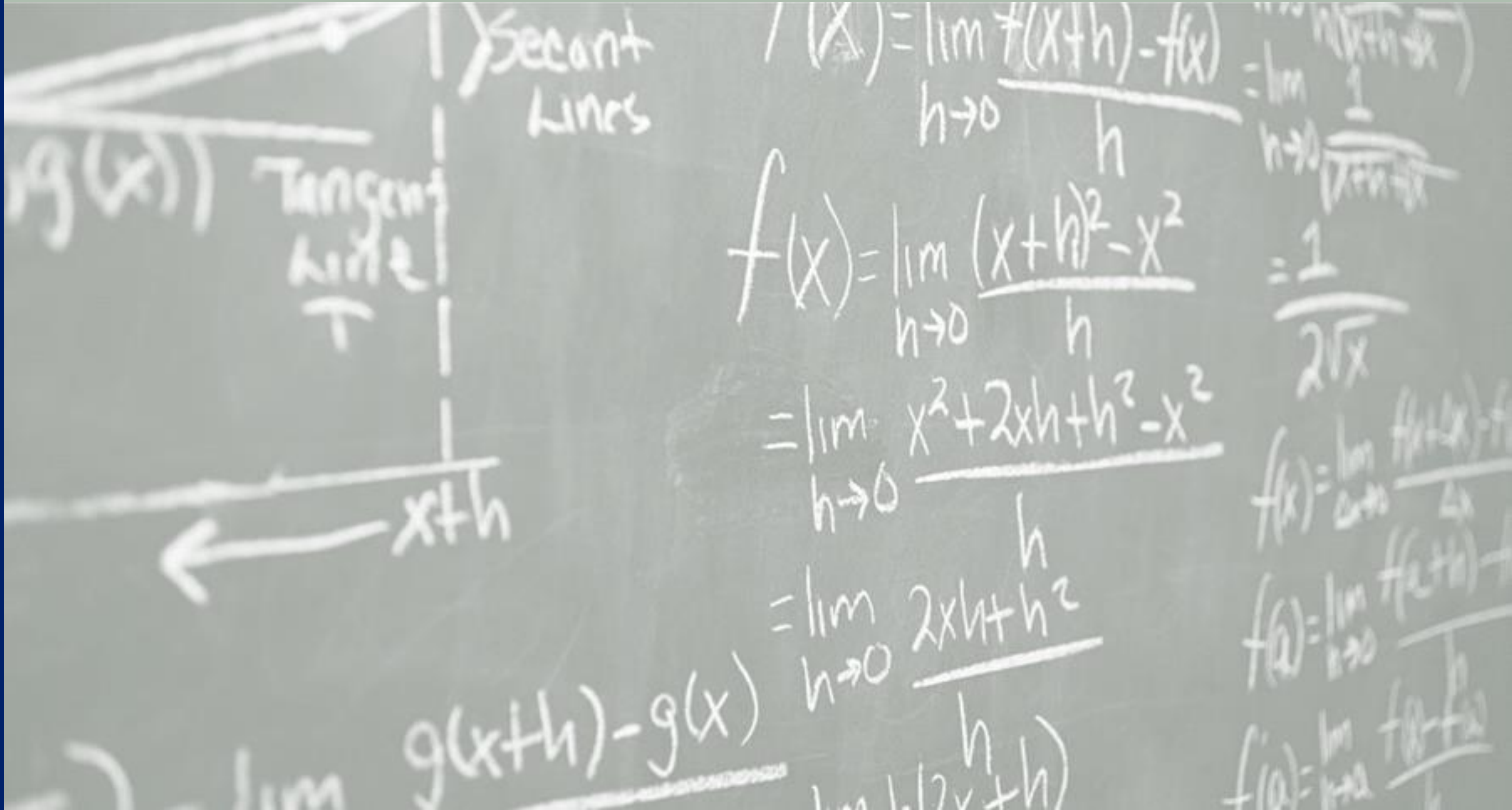
# CAPTAIN’S DECISION SIMULATION BY REINFORCEMENT LEARNING

Fanny REBIFFÉ – Technical presentation assessment – 13/10/2022

# SUMMARY

I. Personal background
II. Development process
III. Context and goal
IV. Data
V. Algorithm design

# I. PERSONAL BACKGROUND



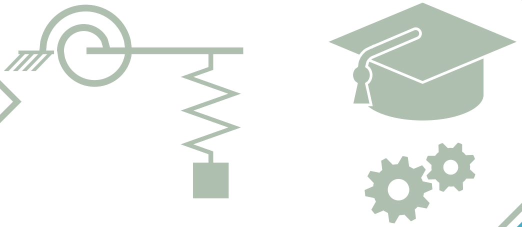
# ACADEMIC & PROFESSIONAL BACKGROUND

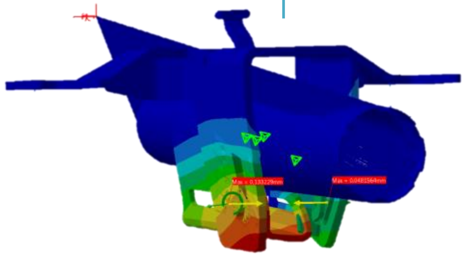
Jtekt – Automotive
FEM calculation
Combined physic &
data driven methods

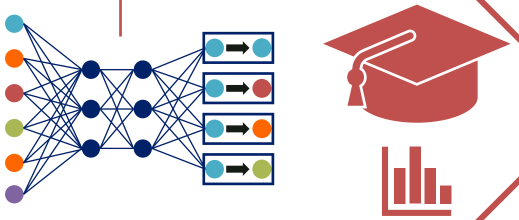
Tiama – Industrial automatons
Mass spring damper modelization
Time series analysis

Framatome – Nuclear
Combinatorial optimization
by reinforcement learning

2016 2017 2018 2019 2020 2021 2022





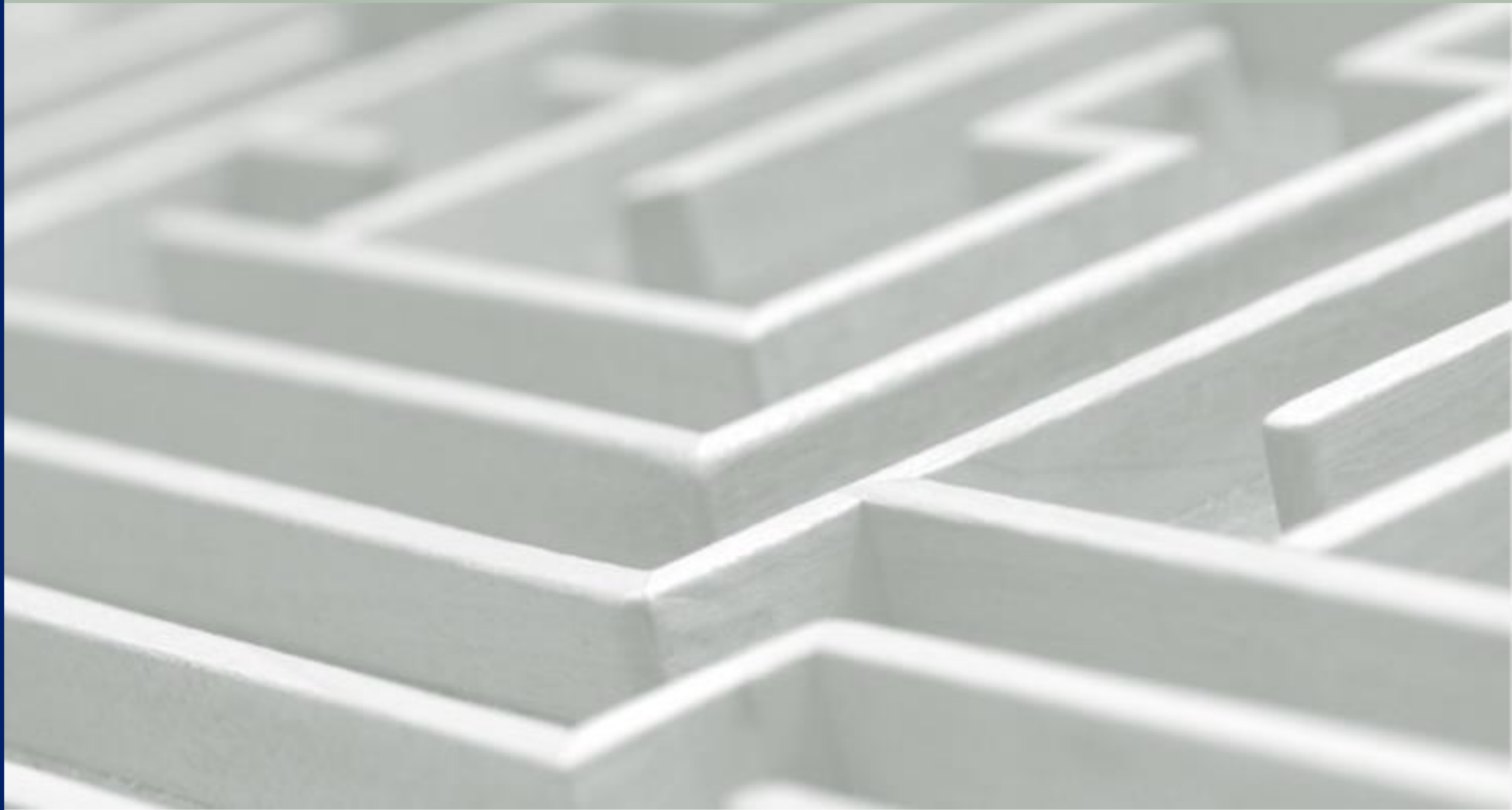


UTC – University of Technology of Compiègne
Engineer diploma in Mechatronics

UGA – University of Grenoble Alpes
Master degree in Statistics & data science

MARIN

# II. DEVELOPMENT PROCESS



# ALGORITHM DEVELOPMENT PROCESS – MLOPS

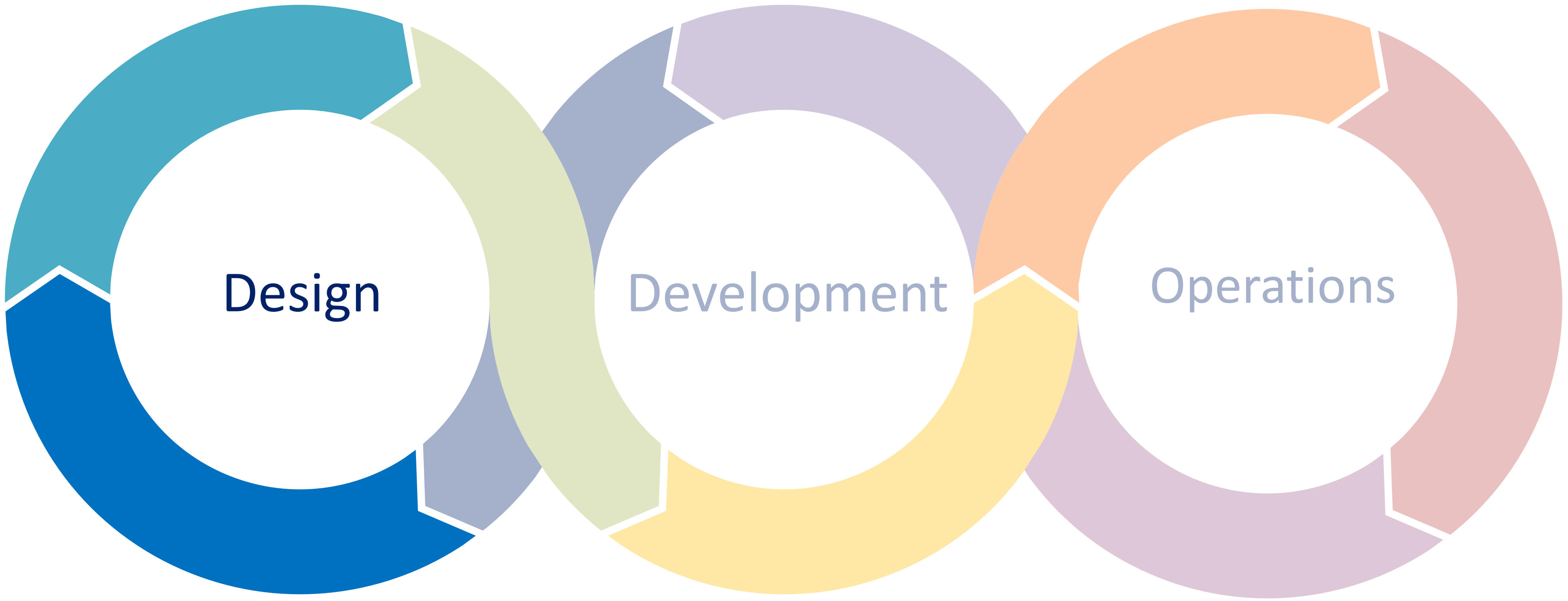

- Requirement engineering
- Features prioritization
- Data acquisition

- Data engineering
- Algorithm design
- Verification & validation

- Deployment
- Data feedback
- Performance monitoring

# III. CONTEXT AND GOAL

# ALGORITHM DEVELOPMENT PROCESS



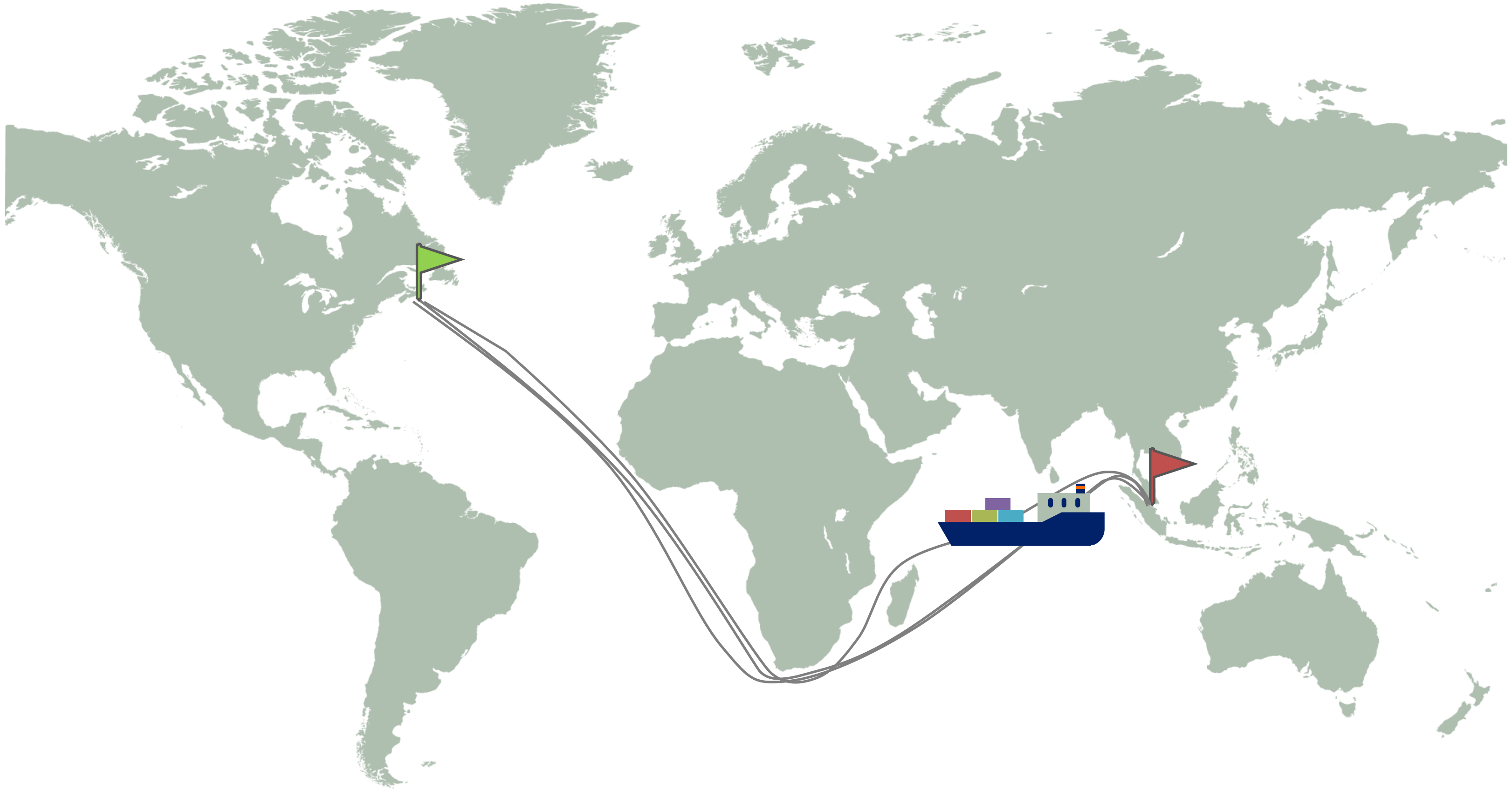

- Requirement engineering
- Features prioritization
- Data acquisition

- Data engineering
- Algorithm design
- Verification & validation

- Deployment
- Data feedback
- Performance monitoring

# CAPTAIN’S DECISION SIMULATION

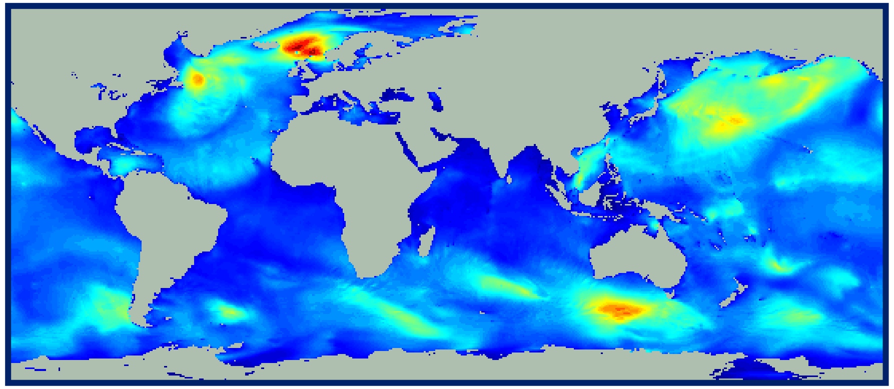
Operate a ship to take the best trajectory in a safe and efficient way



# CAPTAIN’S DECISION SIMULATION

Operate a ship to take the best trajectory in a safe and efficient way

- Choose actions that adapt ship heading and speed …
- in order to take the trajectory minimizing …
- motions, journey duration and emissions.
- knowing :
  - ship characteristics and loading conditions
  - wave conditions forecast



# REWARD FUNCTION

Shipping payment

Fuel price / Carbon emission tax

$$r = \text{final reward} - (\text{duration} + \text{motions} + CO_2)$$

Ship renting cost

Damage price

# IV. DATA

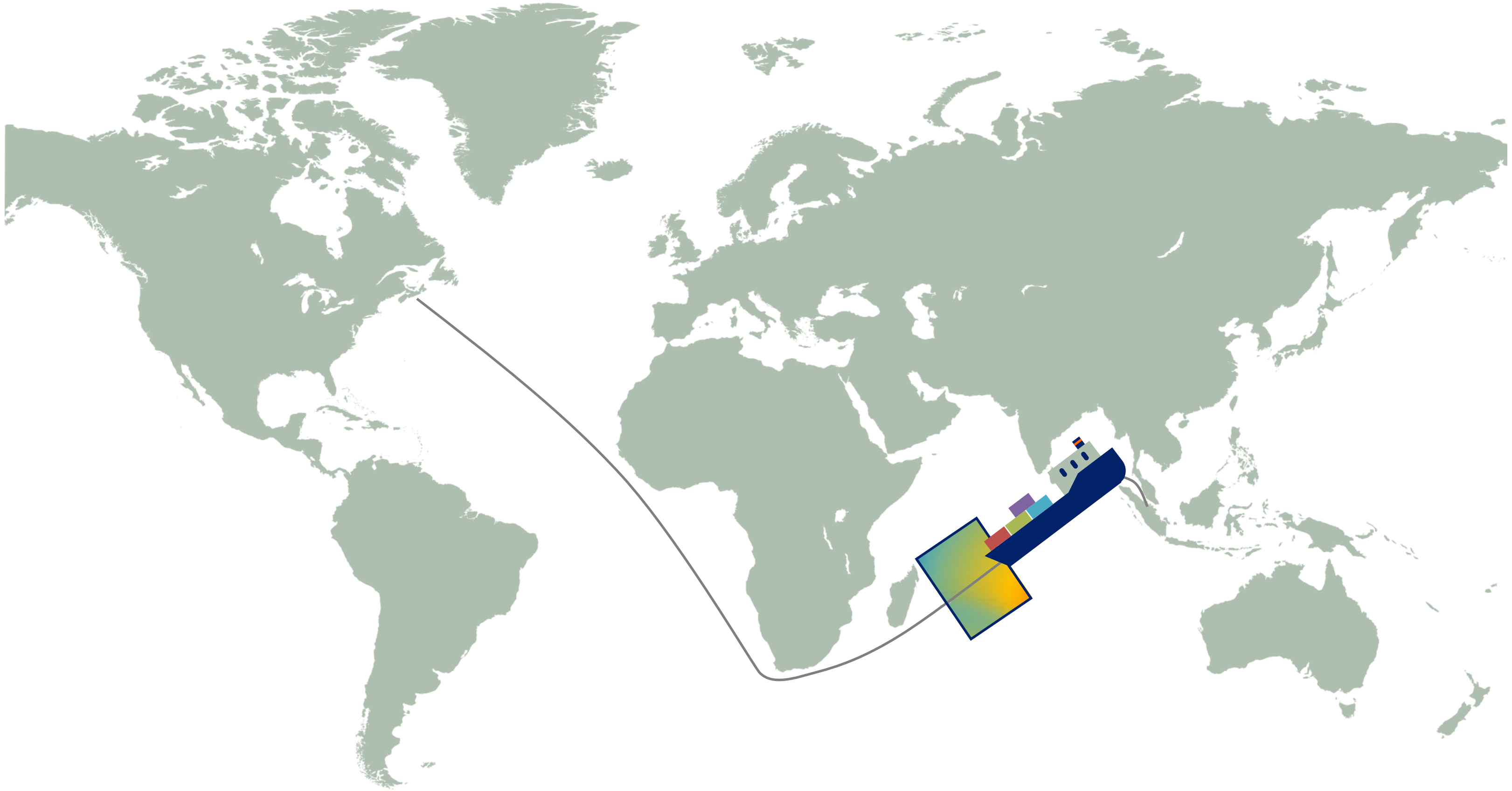
# ALGORITHM DEVELOPMENT PROCESS


- Requirement engineering
- Features prioritization
- Data acquisition

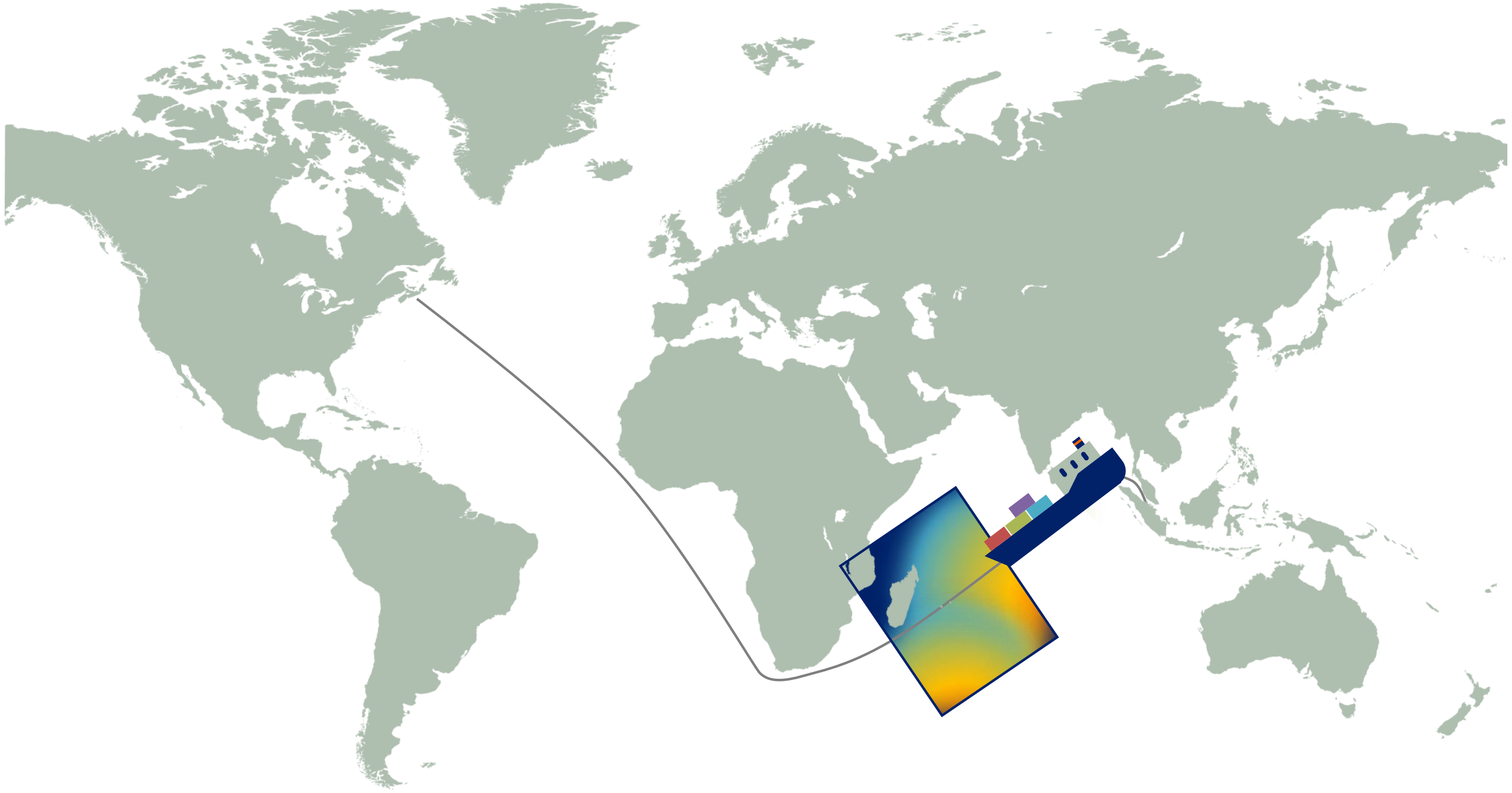
- Data engineering
- Algorithm design
- Verification & validation

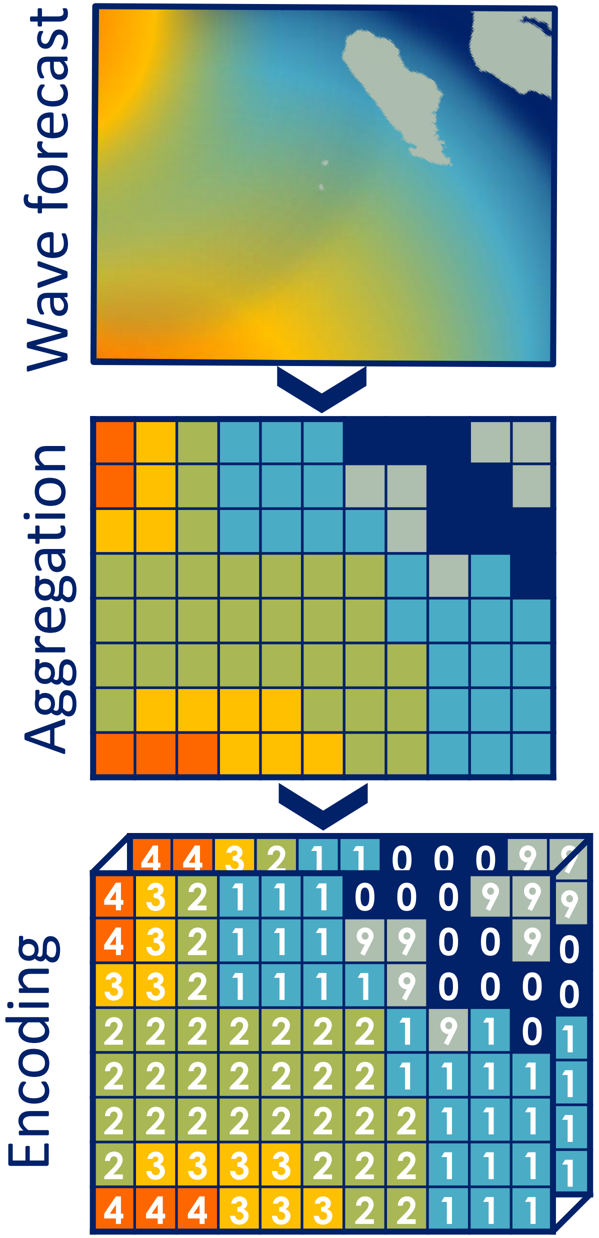
- Deployment
- Data feedback
- Performance monitoring

# WAVE CONDITIONS FORECAST



# WAVE CONDITIONS FORECAST






environment description in a given state

# ACTION SPACE

Heading actions

← ↑ →

Speed actions

extra slow slow normal

| Action space | Encoding |
|---|---|
| {←;extra slow} | 0 |
| {↑;extra slow} | 1 |
| {→;extra slow} | 2 |
| {←;slow} | 3 |
| {↑;slow} | 4 |
| {→;slow} | 5 |
| {←;normal} | 6 |
| {↑; normal} | 7 |
| {→; normal} | 8 |

# V. ALGORITHM DESIGN

# ALGORITHM DEVELOPMENT PROCESS


- Requirement engineering
- Features prioritization
- Data acquisition

- Data engineering
- Algorithm design
- Verification & validation

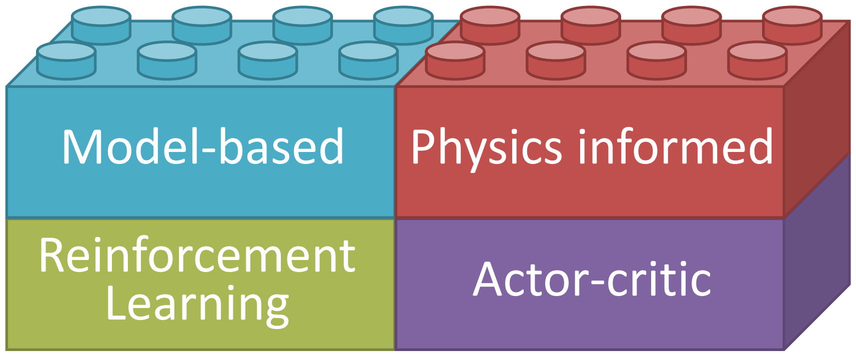
- Deployment
- Data feedback
- Performance monitoring

# CHOSEN ALGORITHM

Operate the ship to take the best trajectory in a safe and efficient way

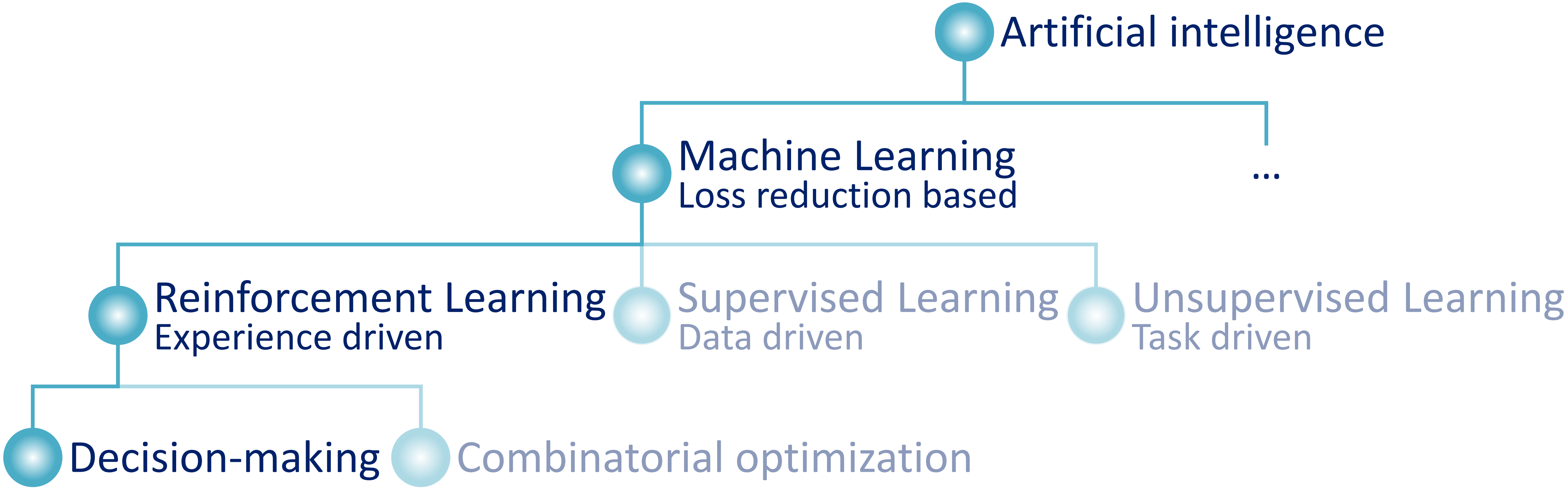
- Experience driven
- Stability criteria
- Sample efficient (Experience risk & cost)
- Existence of physics model

➔ Physics informed model based actor-critic reinforcement learning

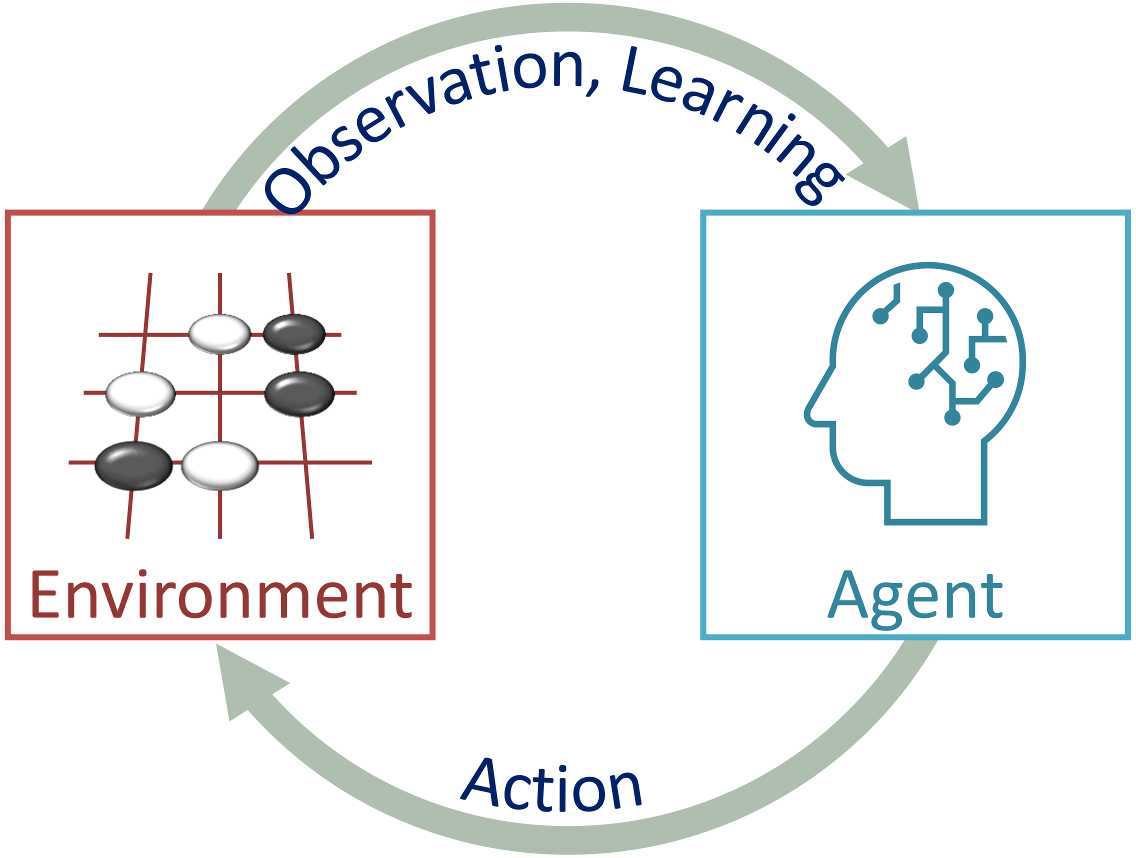


# FAMILY TREE

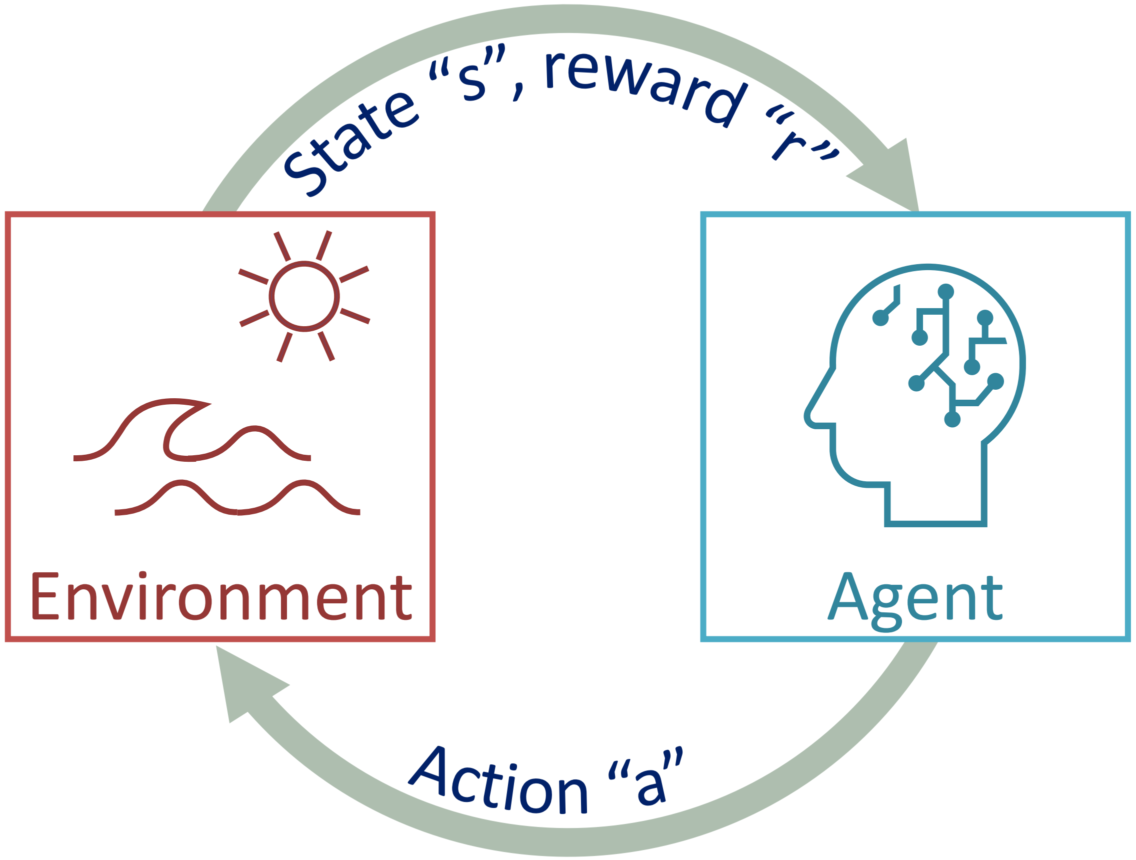

Reinforcement
Learning


Artificial intelligence
Machine Learning
Loss reduction based
...
Reinforcement Learning
Experience driven
Supervised Learning
Data driven
Unsupervised Learning
Task driven
Decision-making
Combinatorial optimization

# REINFORCEMENT LEARNING

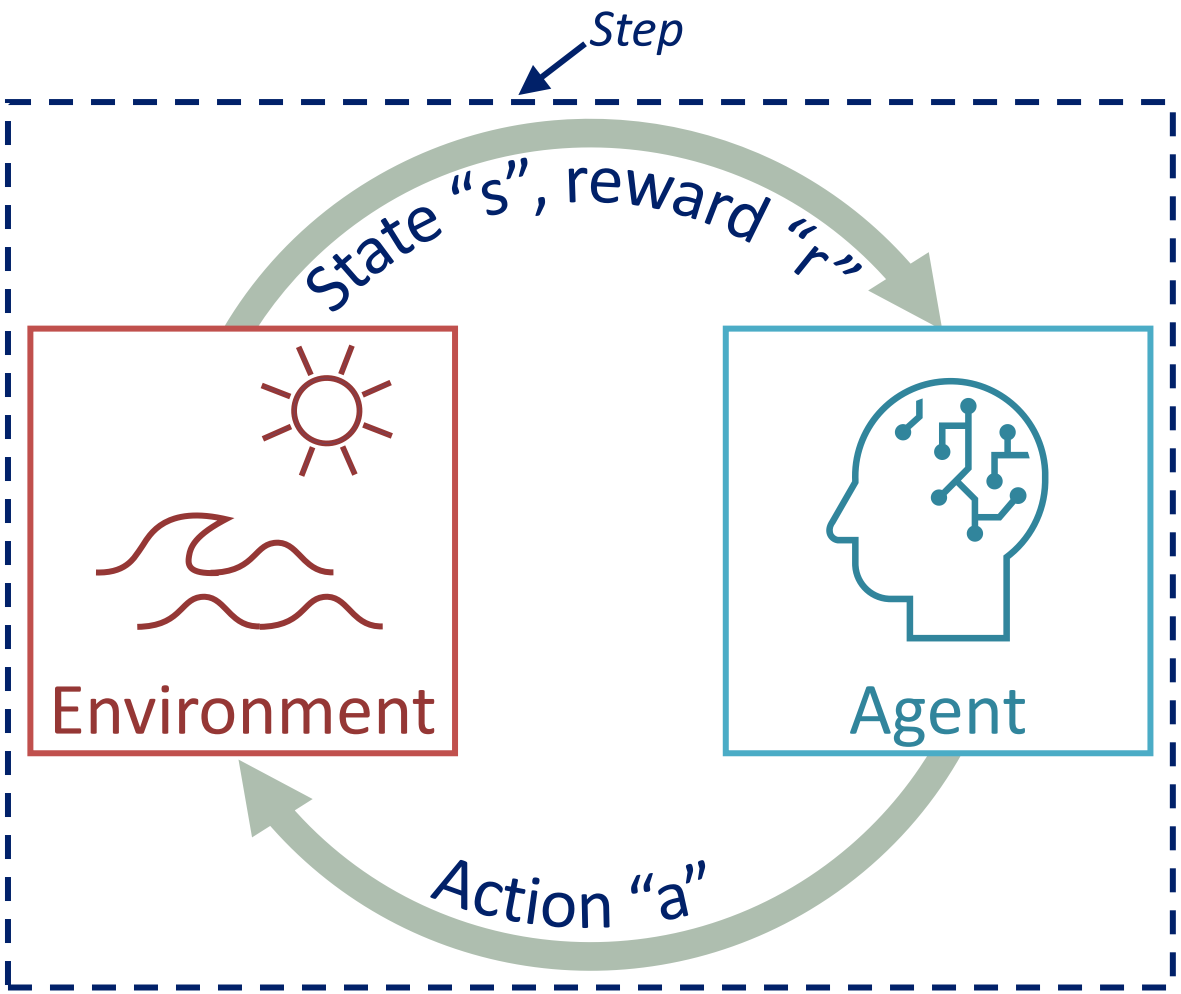



# REINFORCEMENT LEARNING

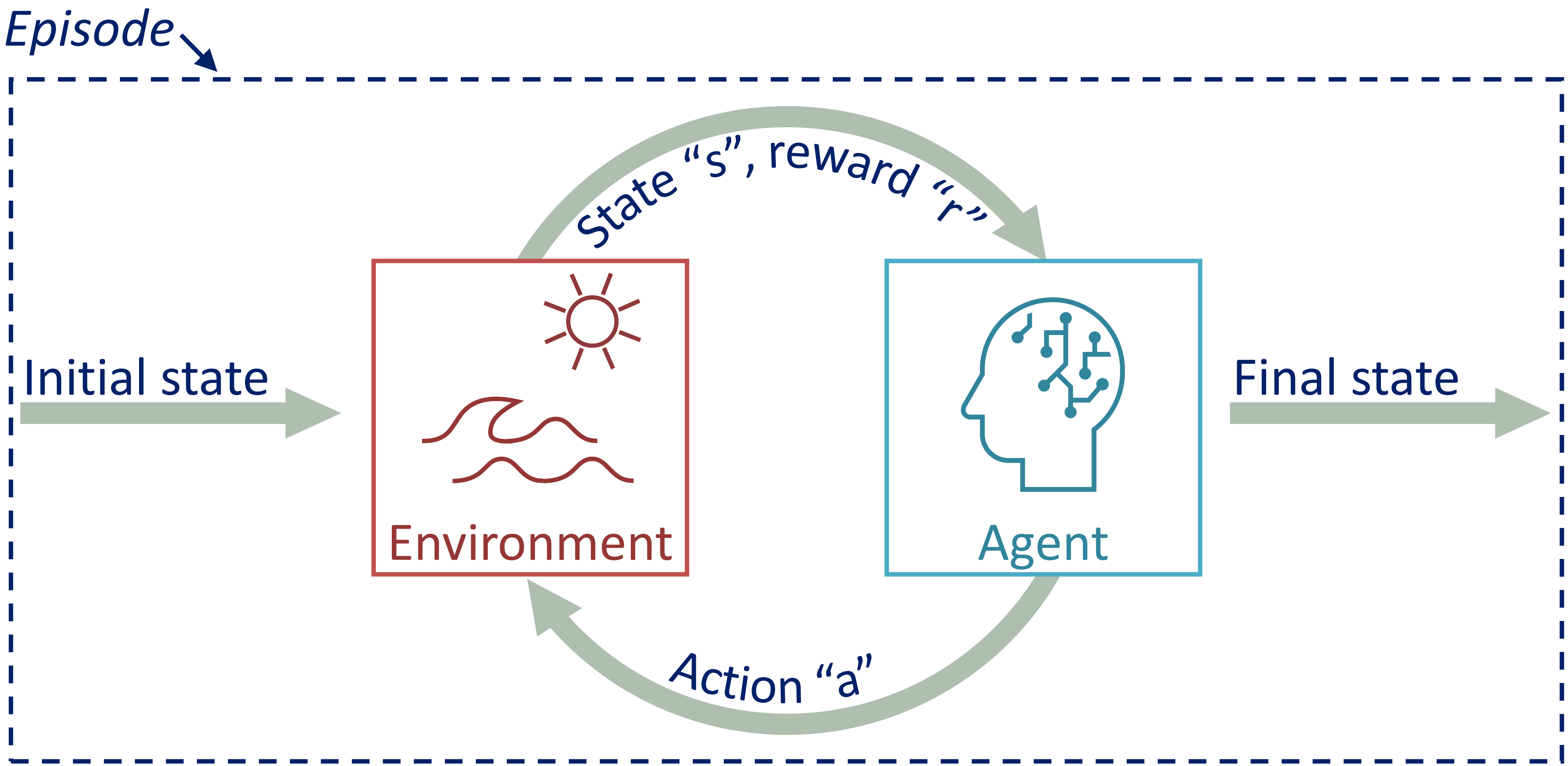
# REINFORCEMENT LEARNING

Reinforcement Learning

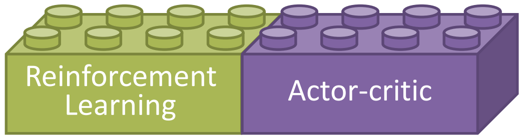


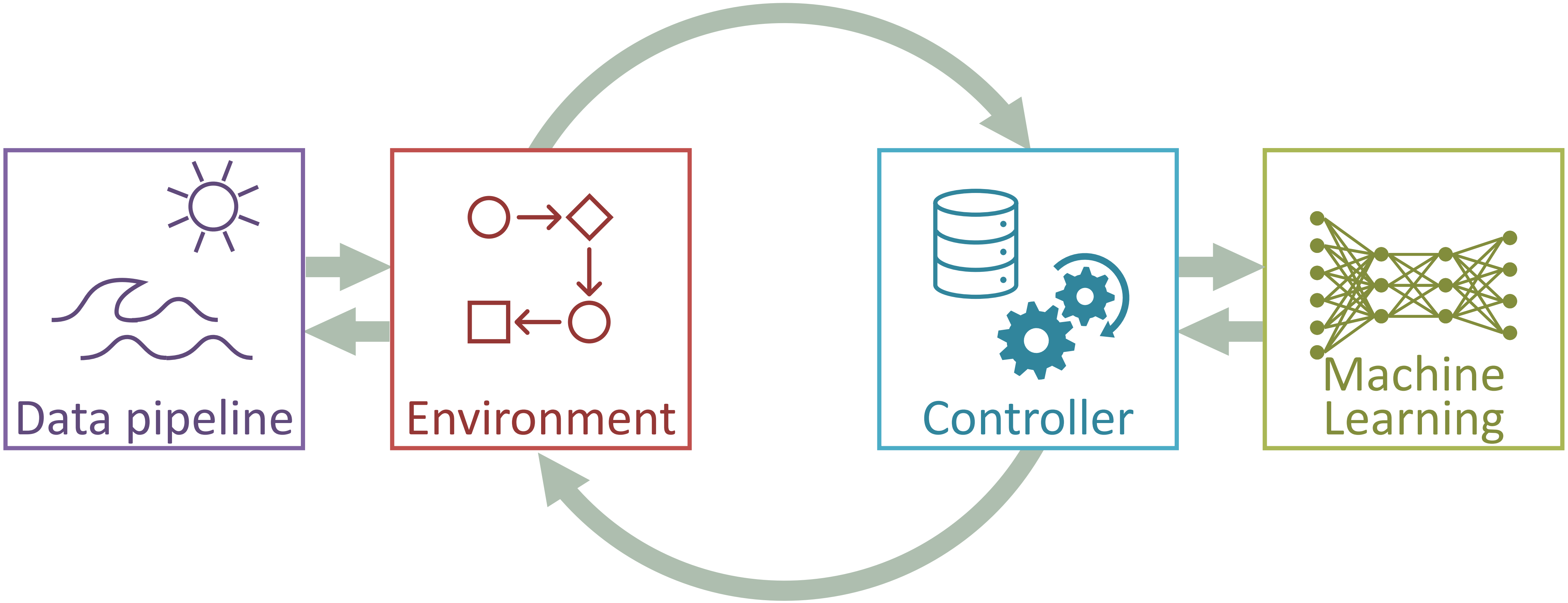
# REINFORCEMENT LEARNING

Reinforcement Learning

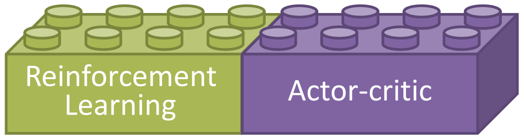


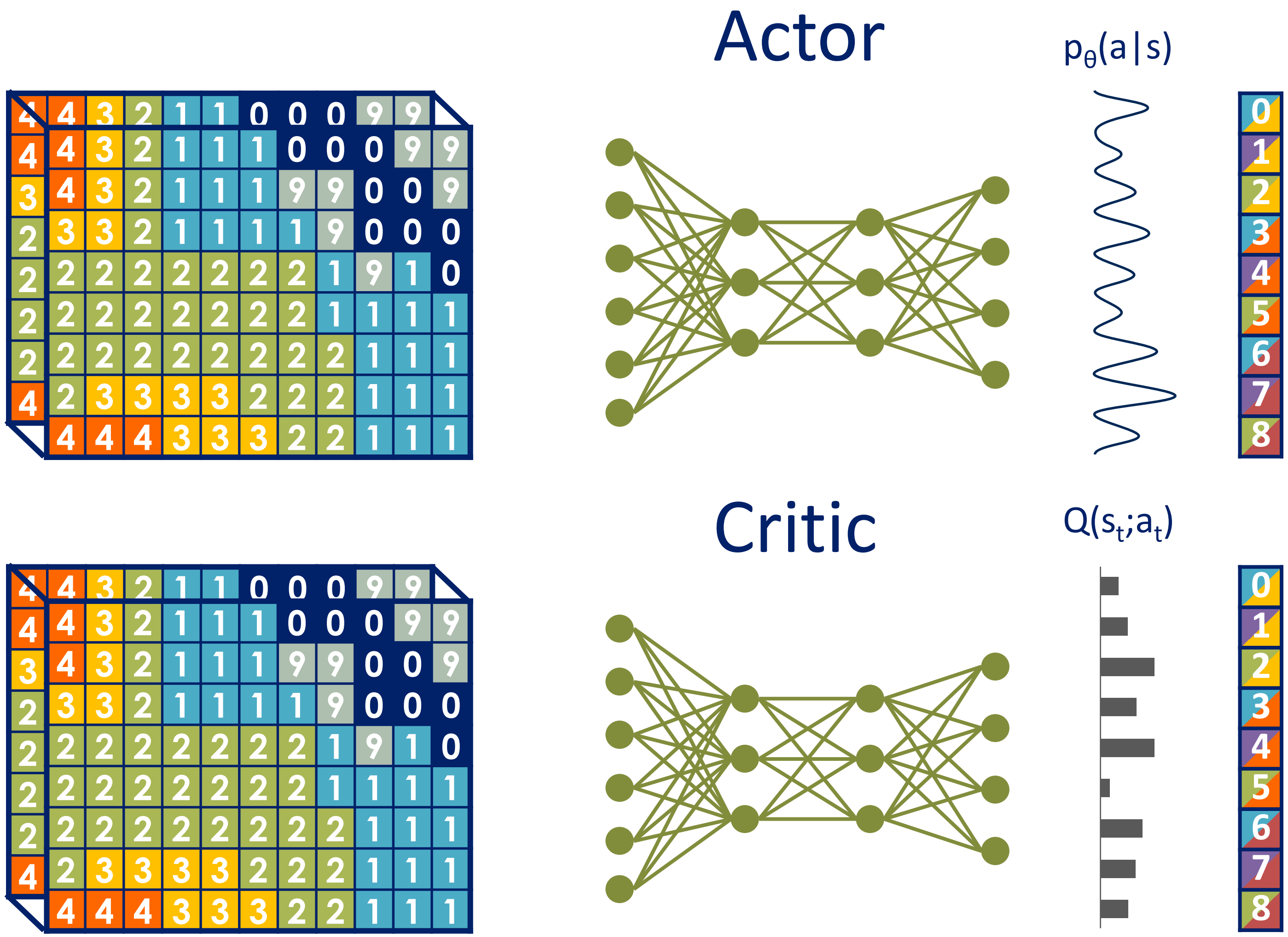
# REINFORCEMENT LEARNING



# ACTOR-CRITIC

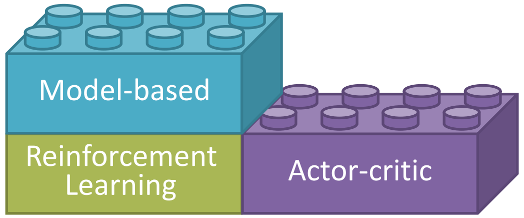


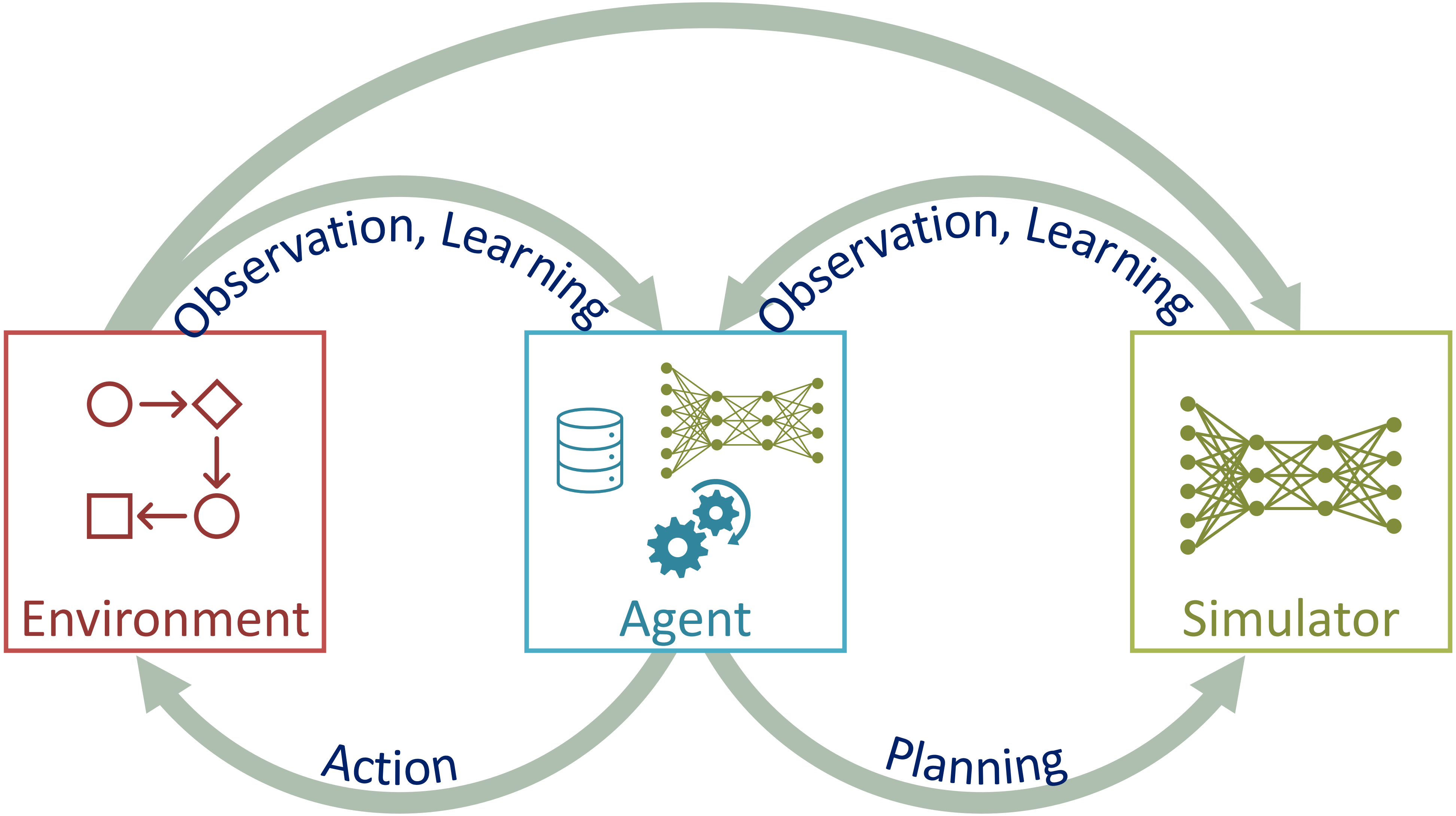

Prediction of policy
(probability to take an action)

Prediction of Q-value
(cumulative reward)

# MODEL-BASED

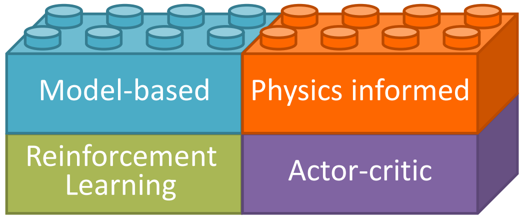


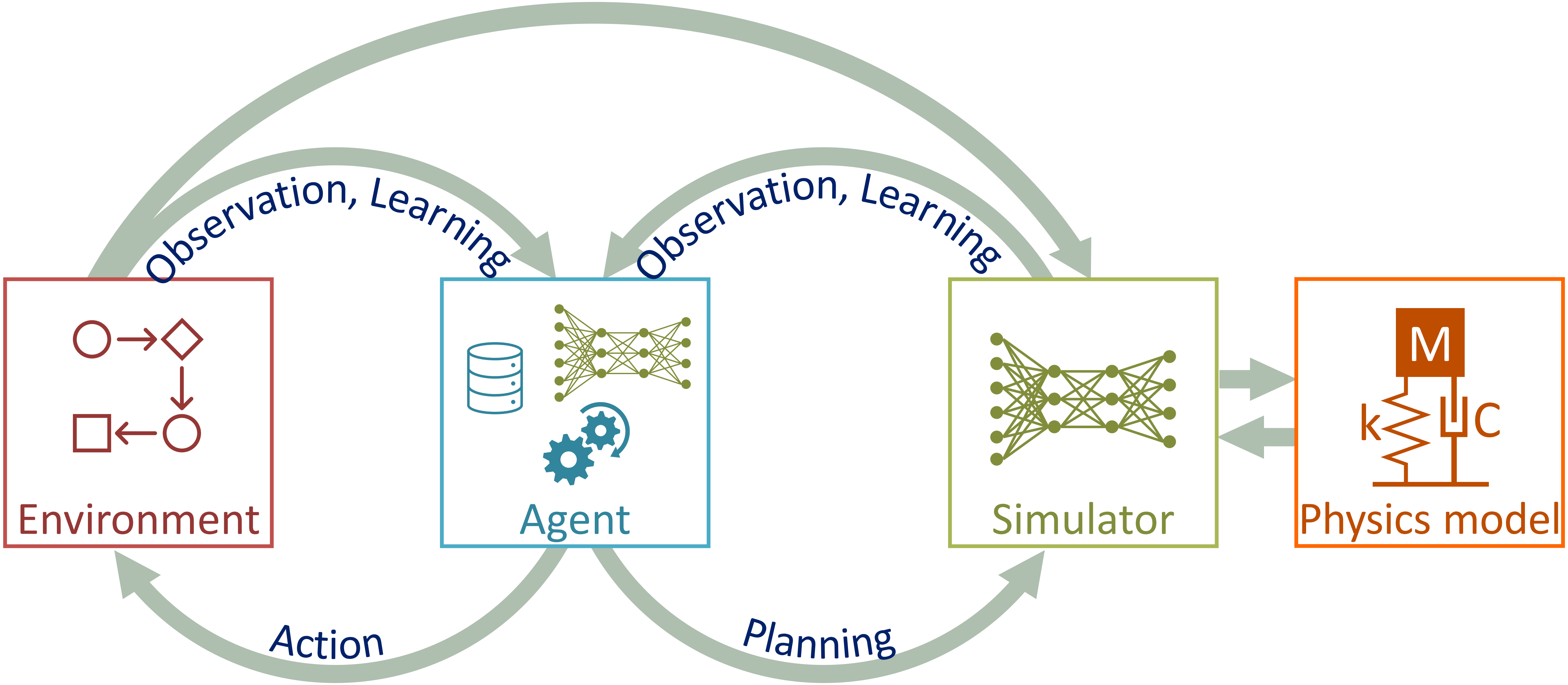
Model-based
Reinforcement Learning
Actor-critic

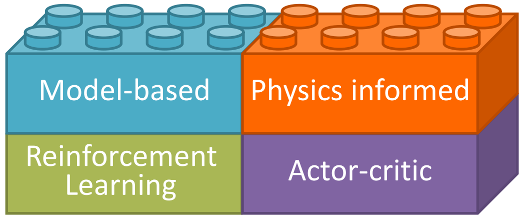
Observation, Learning
Observation, Learning
Environment
Agent
Simulator
Action
Planning

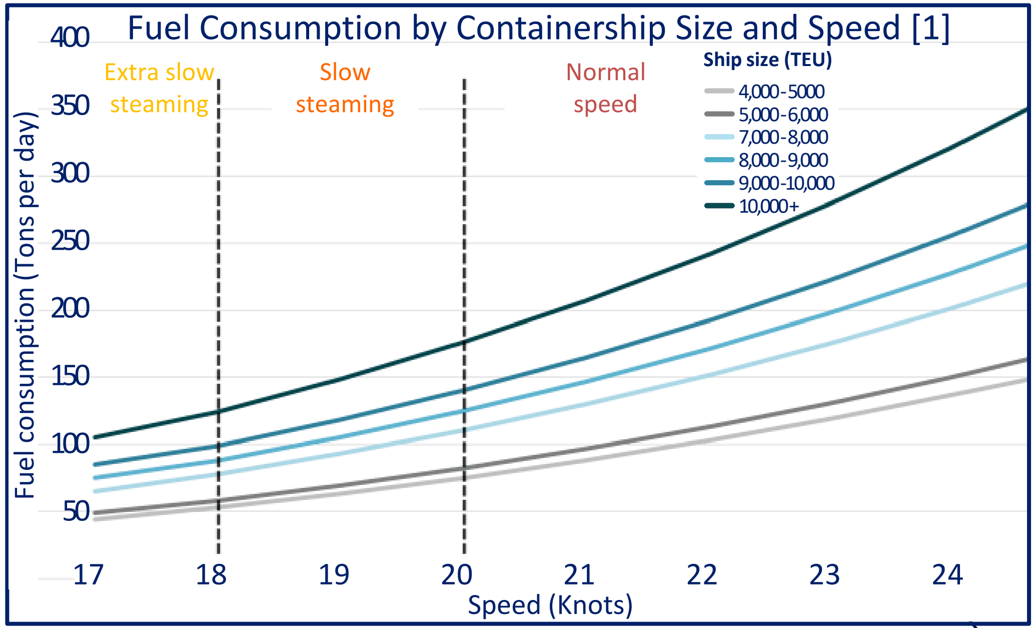
# PHYSIC-INFORMED



# PHYSICS MODEL



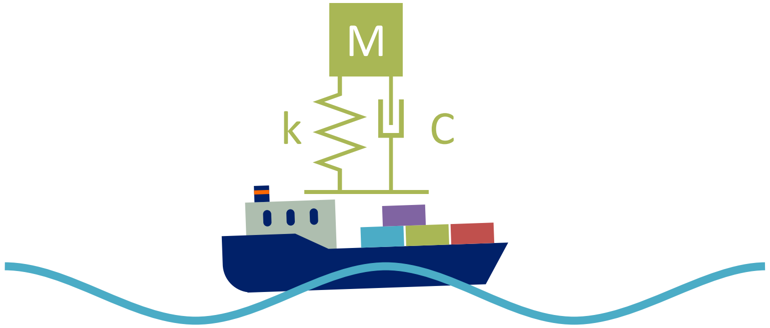




Shipping payment

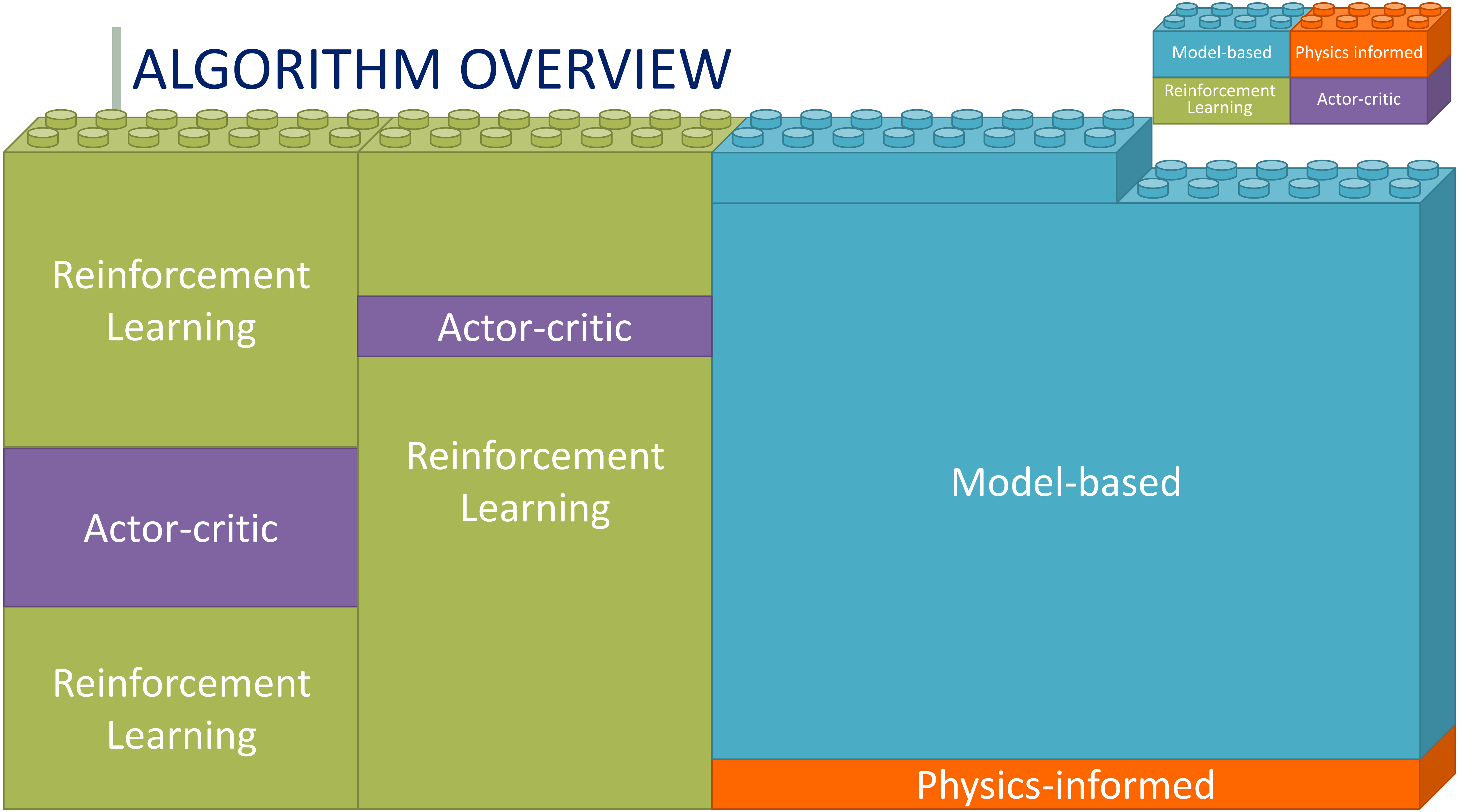
$$r = \text{final reward} - (\text{duration} + \text{motions} + CO_2)$$

*f (distance, speed)*

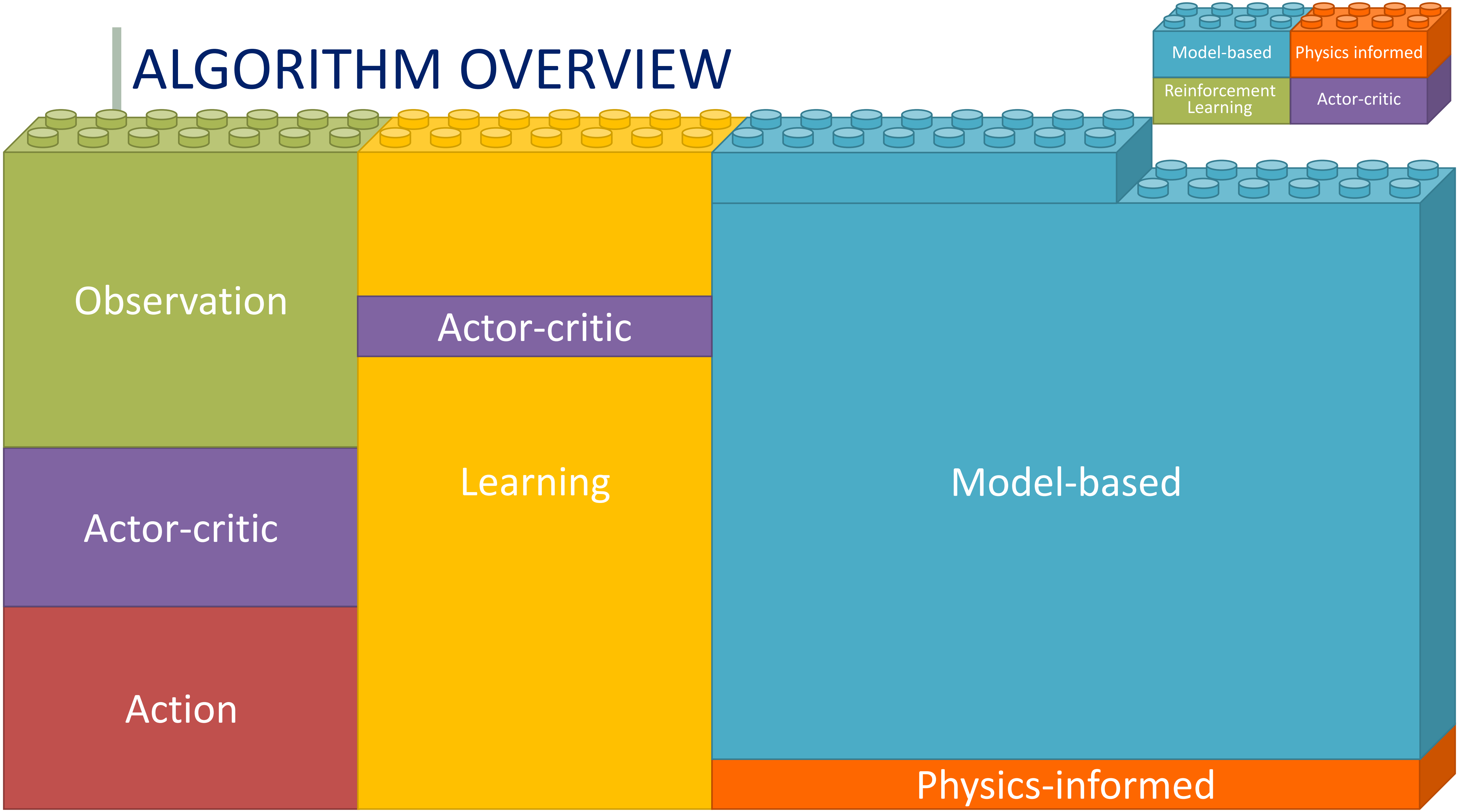
*f(t, a, ω, m, c, k)*



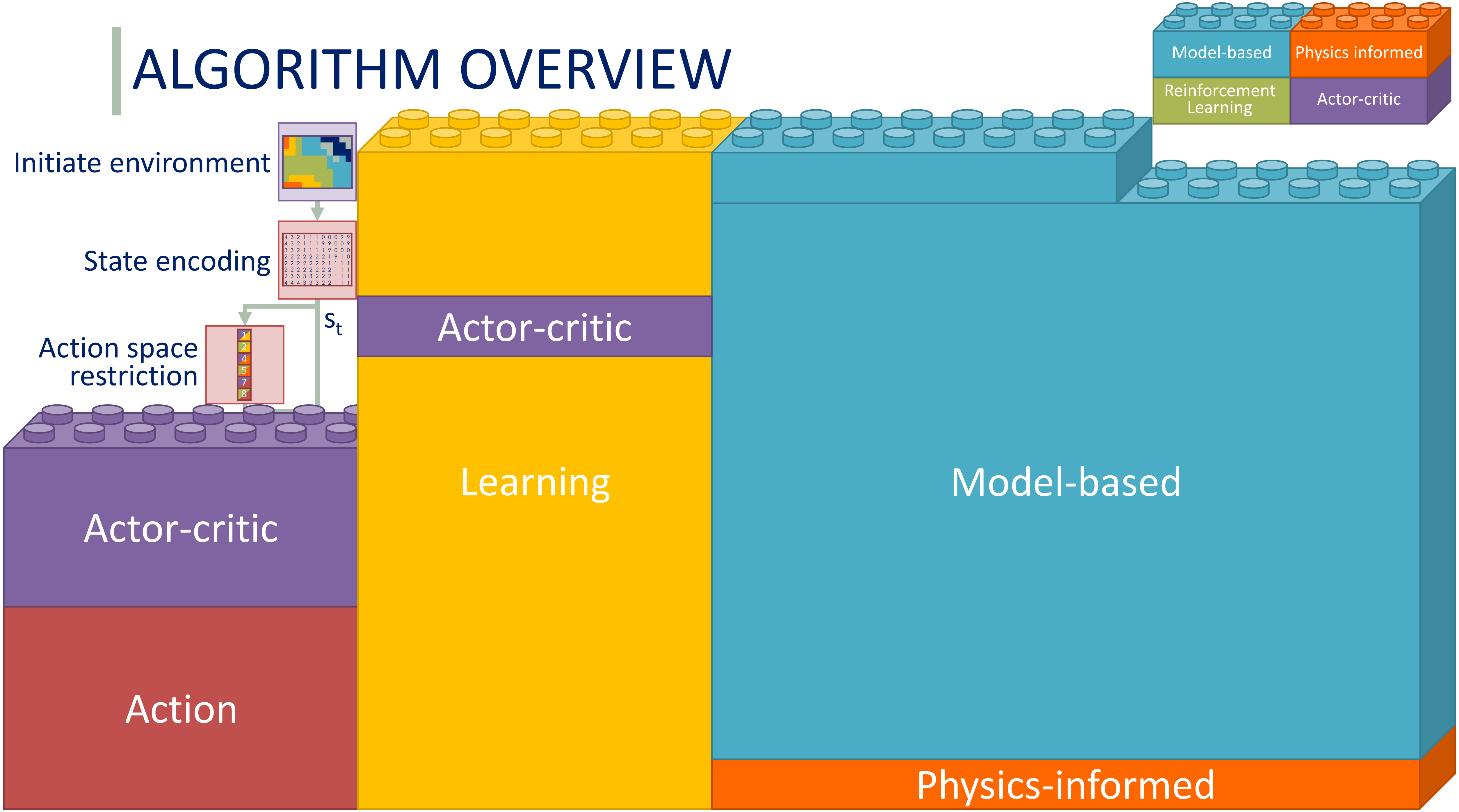
# ALGORITHM OVERVIEW



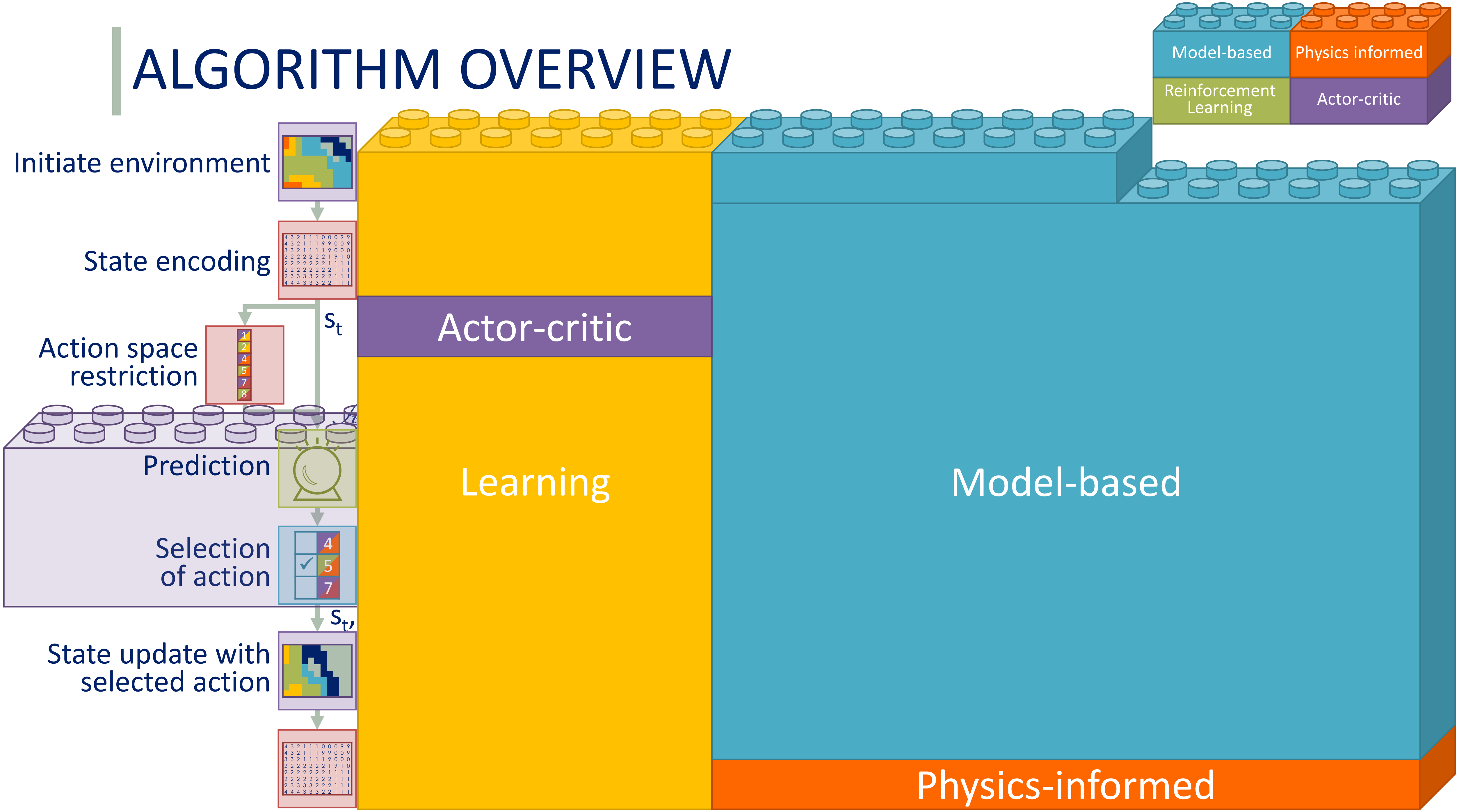
# ALGORITHM OVERVIEW



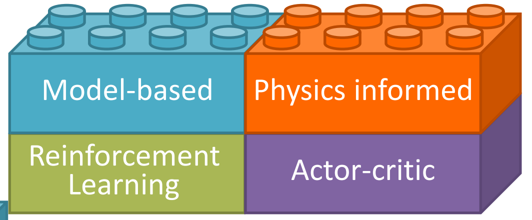
# ALGORITHM OVERVIEW

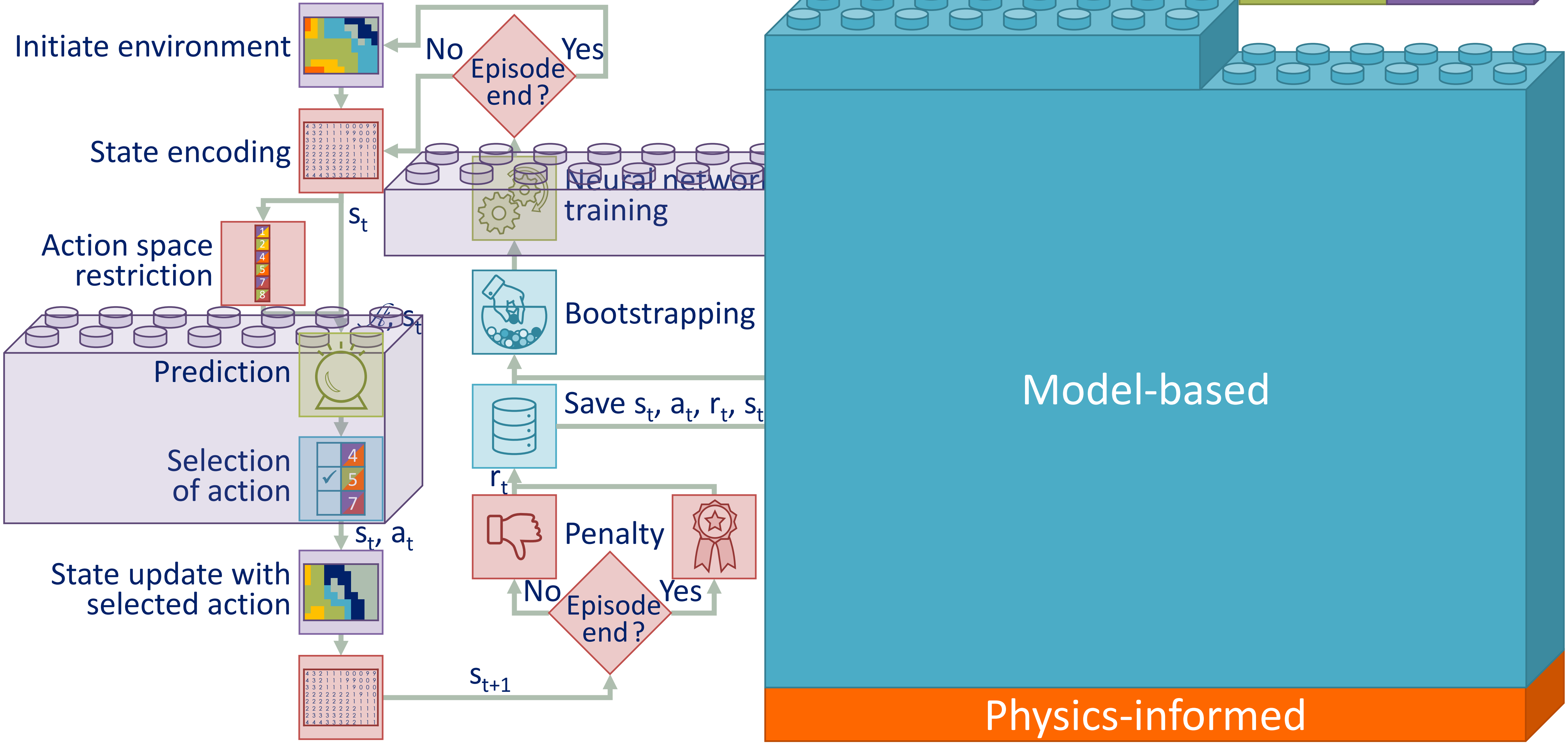


# ALGORITHM OVERVIEW

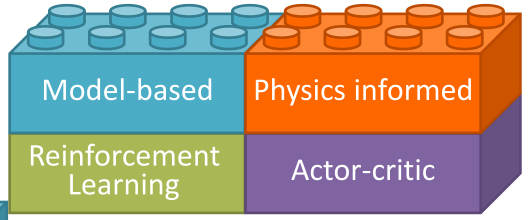


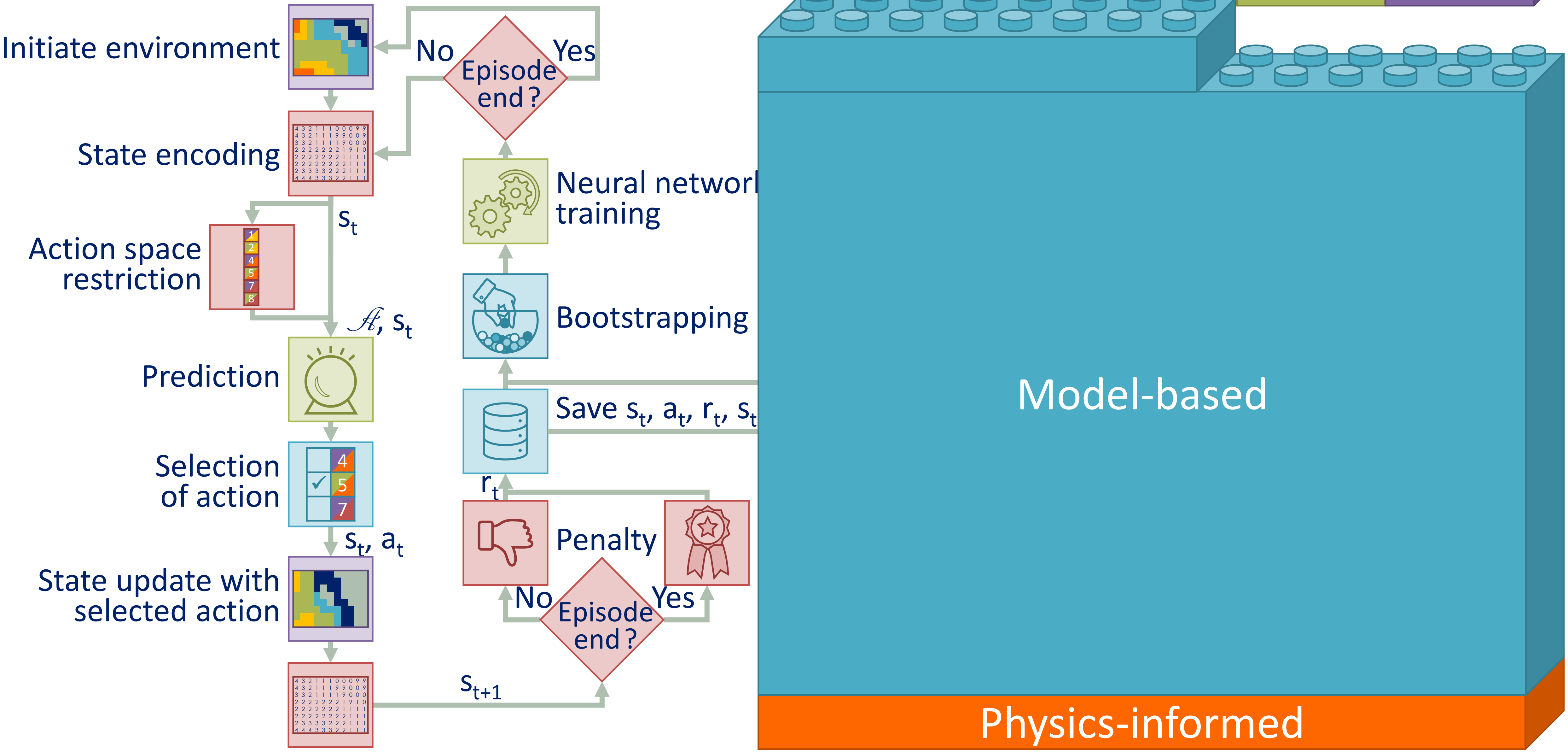
# ALGORITHM OVERVIEW



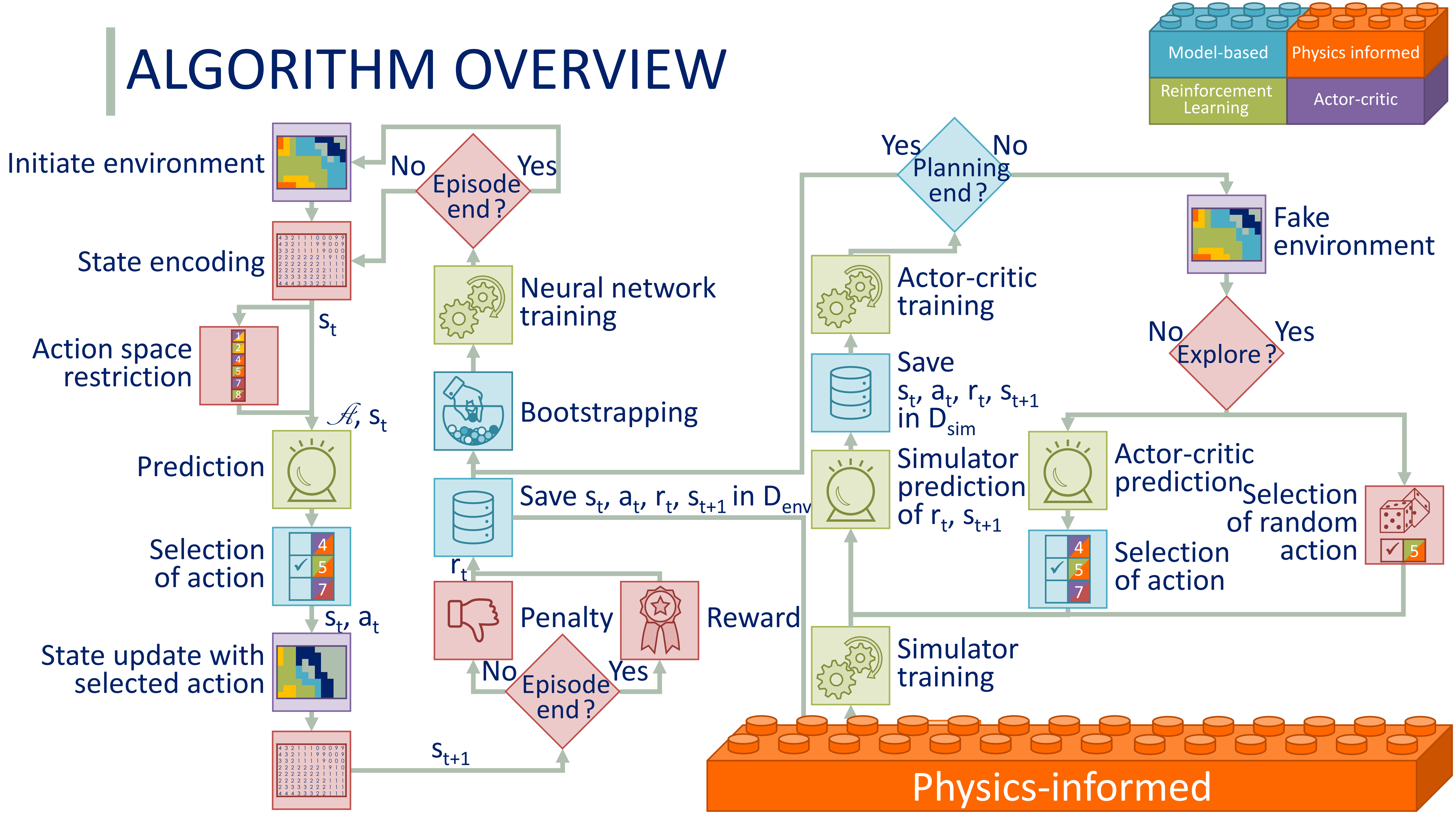



# ALGORITHM OVERVIEW

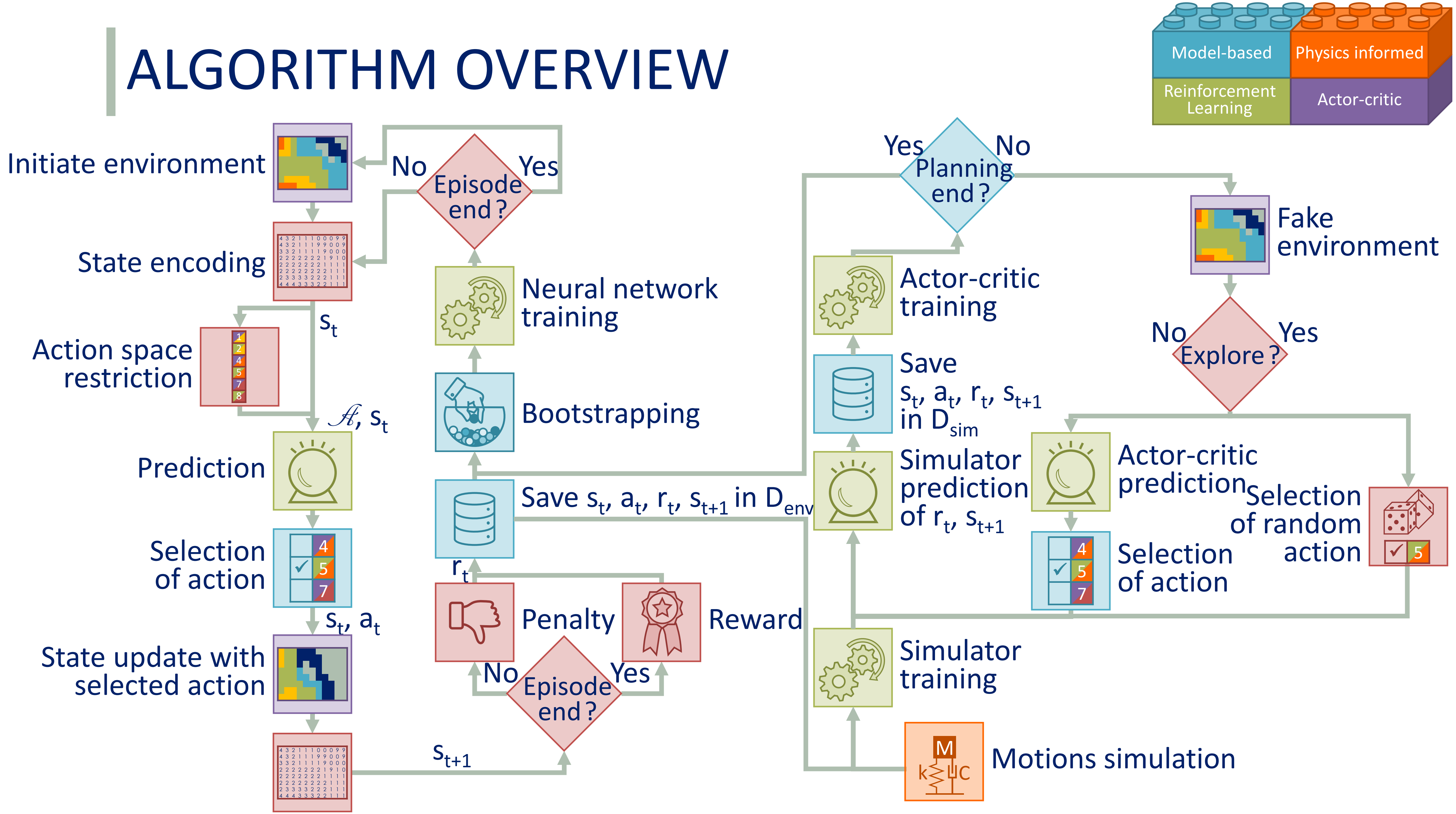


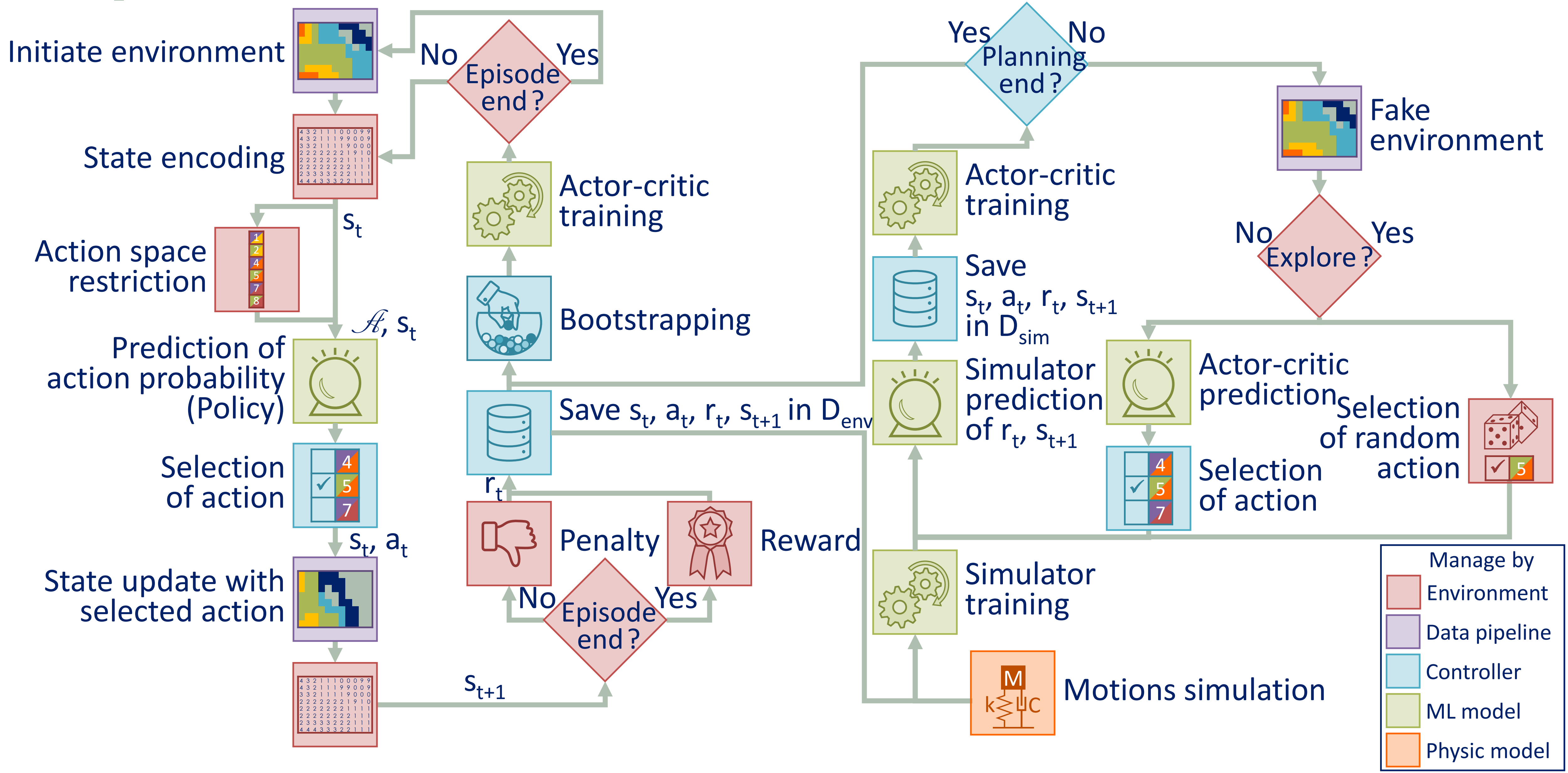
# ALGORITHM OVERVIEW



# ALGORITHM OVERVIEW



# ALGORITHM OVERVIEW



# ALGORITHM DEVELOPMENT PROCESS


- Requirement engineering
- Features prioritization
- Data acquisition

- Data engineering
- Algorithm design
- Verification & validation

- Deployment
- Data feedback
- Performance monitoring

MARIN

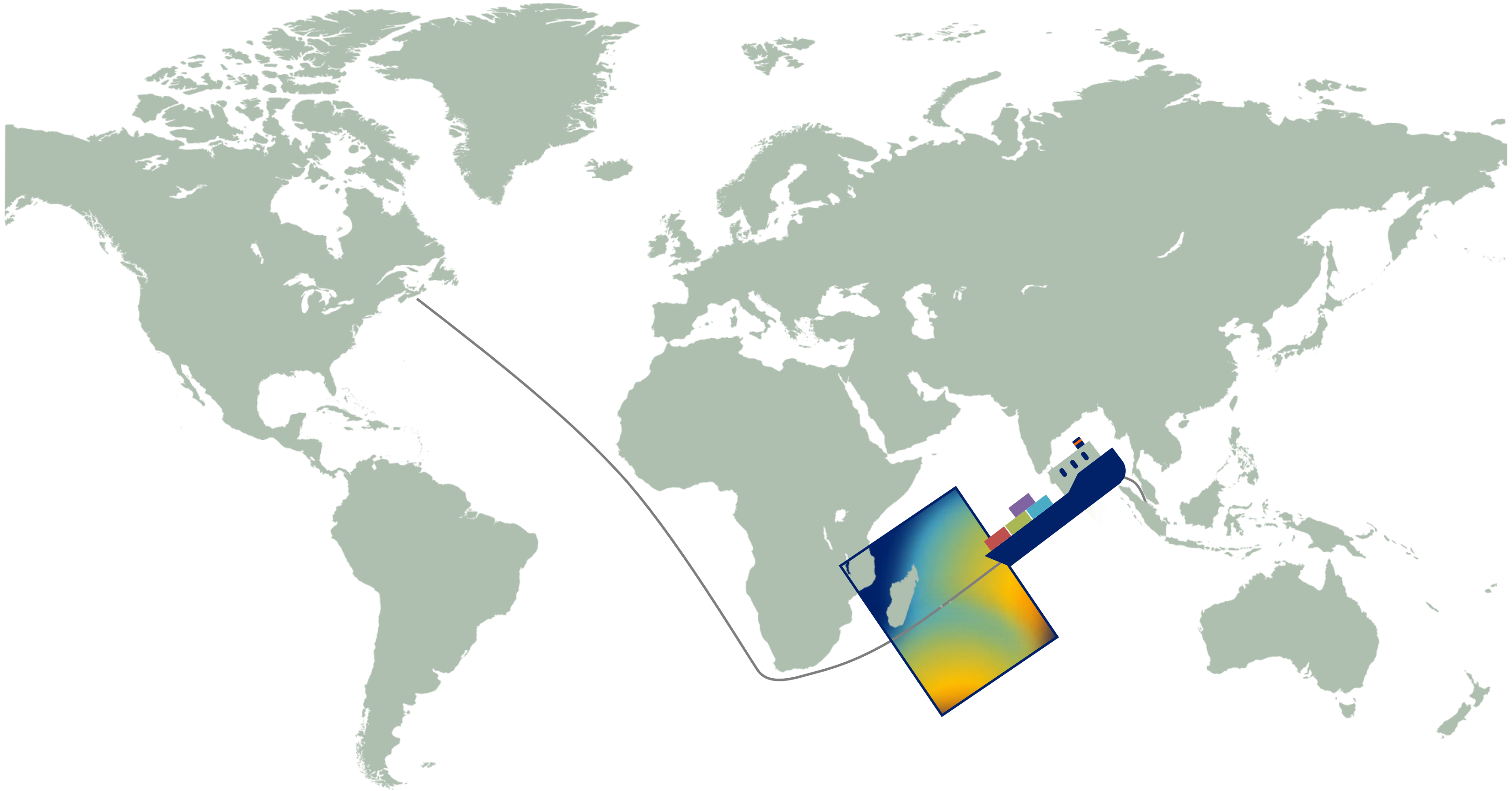
# THANK YOU

# APPENDICES

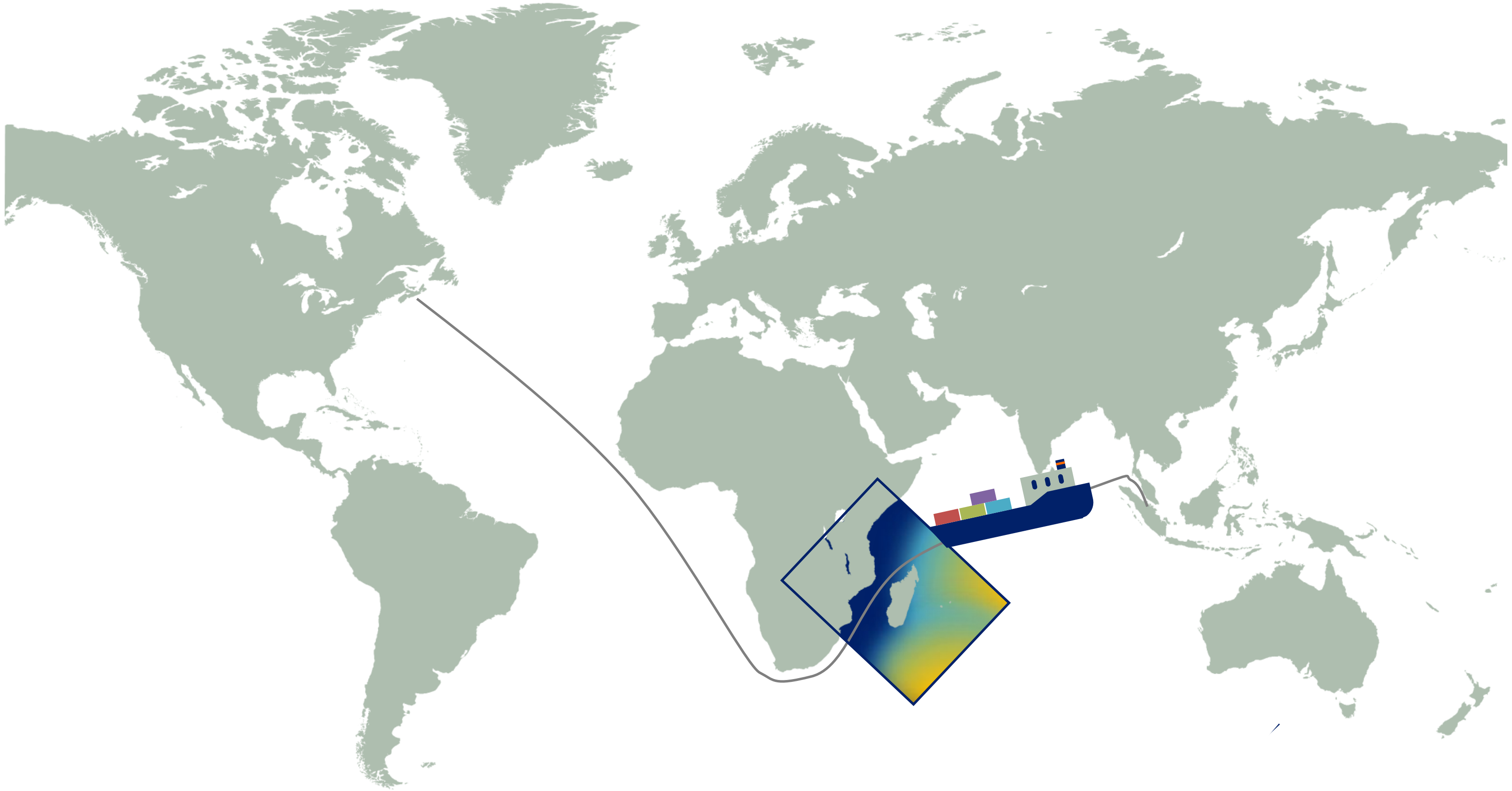
# SOURCES


- [1] The Geography of Transport Systems [On line] *Fuel Consumption by Containership Size and Speed*. https://transportgeography.org/contents/chapter4/transportation-and-energy/fuel-consumption-containerships
- [2] DeppMind [On line]. *AlphaGo*. https://www.deepmind.com/research/highlighted-research/alphago
- [3] Wang, T., Baol, X., Clavera, I., Hoang, J. Wen, Y., Langlois, E., Zhang, S., Zhang, G. Abbeel, P., Ba, J., Benchmarking Model-Based Reinforcement Learning. *arXiv: 1907.02057, 2017*. [On line] https://arxiv.org/pdf/1907.02057.pdf
- [4] Pascanu, R., Li, Y., Vinyals, O., Heess, N., Buesing, L., Racanière, S., Reichert, D., Weber, T., Wierstra, D., Battaglia, P., Learning model-based planning from scratch. *arXiv: 1707.06170, 2017*. [On line] https://arxiv.org/pdf/1707.06170.pdf
- [5] Liu, X. & Wang, J., Physics-informed Dyna-Style Model-Based Deep Reinforcement Learning for Dynamic Control. *arXiv: 2108.00128, 2021*. [On line] https://arxiv.org/pdf/2108.00128.pdf
- [6] Yu, C. & Rosendo, A., Risk-Aware Model-Based Control. *Frontiers in robotics and AI, 2021.* [On line] https://www.frontiersin.org/articles/10.3389/frobt.2021.617839/

# WAVE CONDITIONS FORECAST

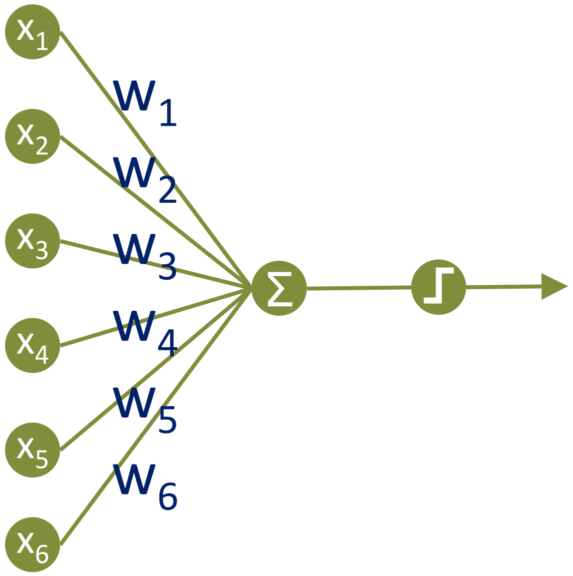


# WAVE CONDITIONS FORECAST

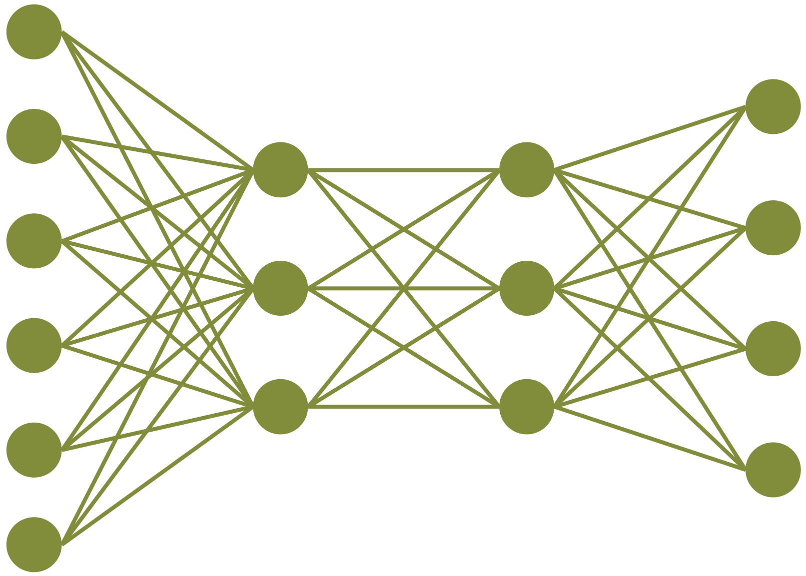


# NEURAL NETWORKS

A perceptron

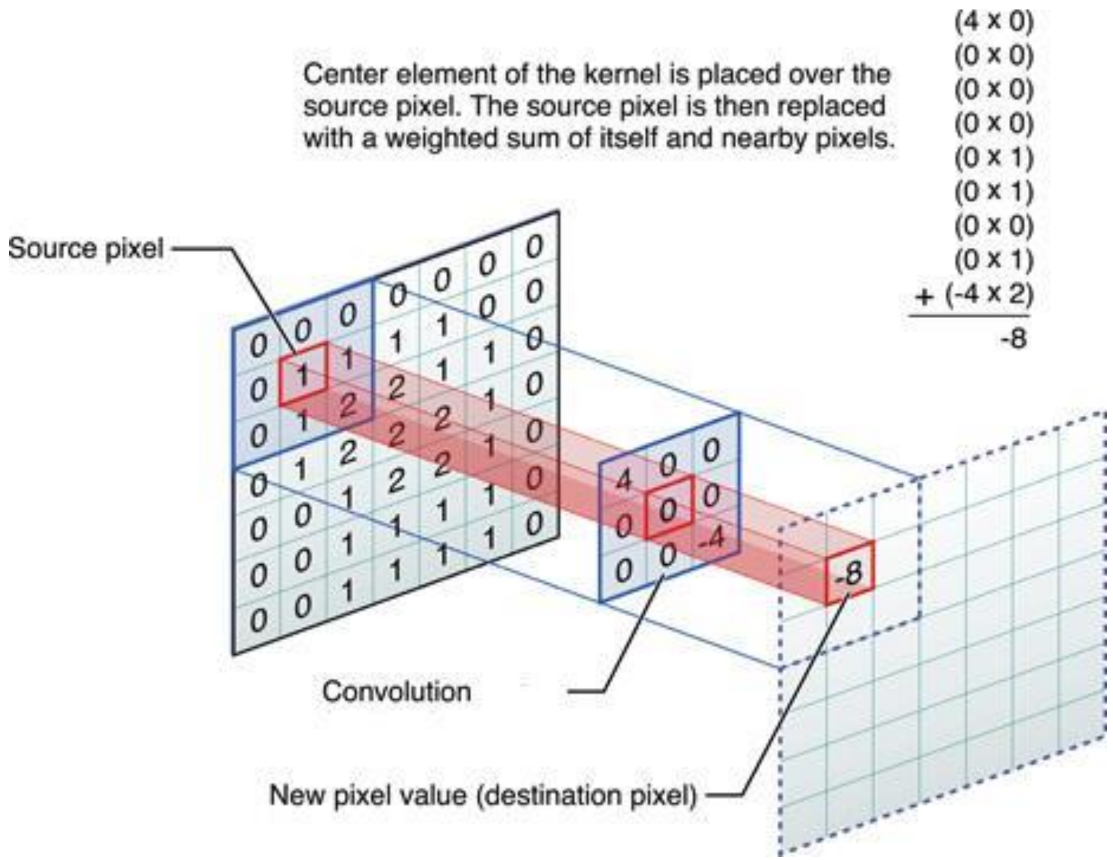



Multi layer perceptron



# CONVOLUTION

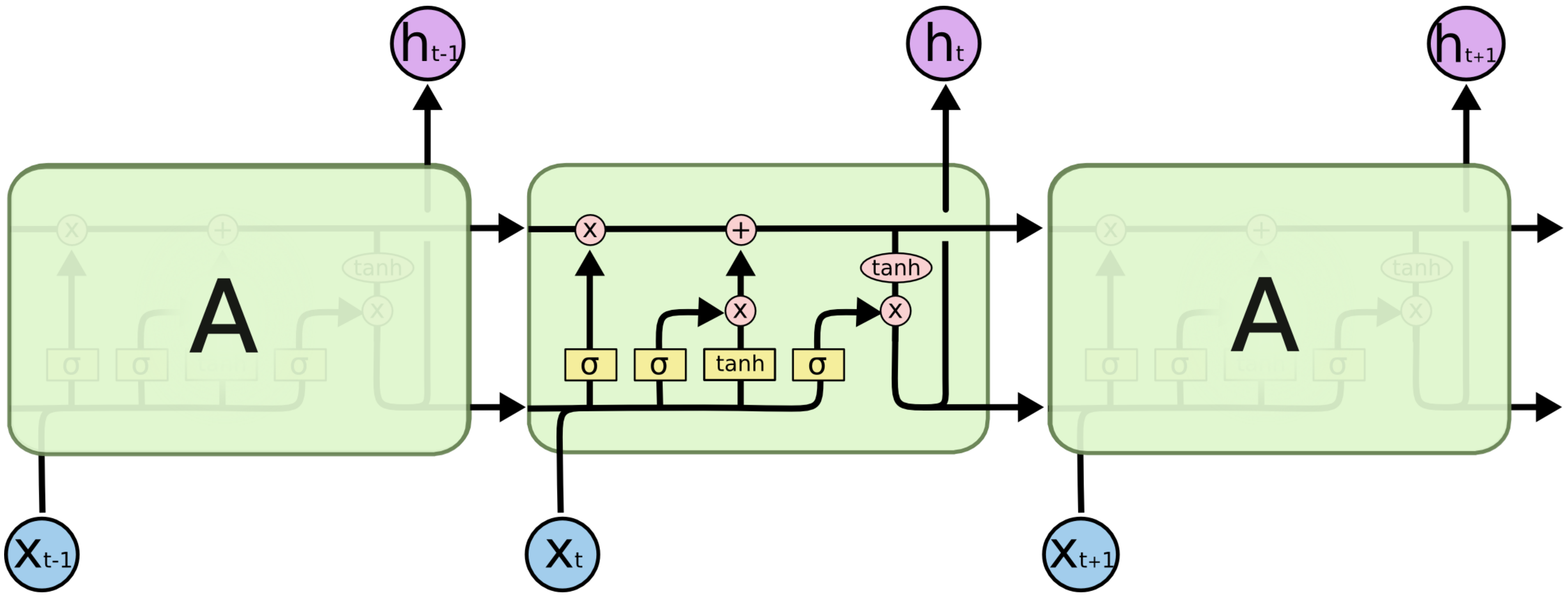
- Scan the matrix with a filter then gives a number corresponding to how much each sub section of the matrix match the filter




https://medium.com/@bdhuma/6-basic-things-to-know-about-convolution-daef5e1bc411

# LSTM

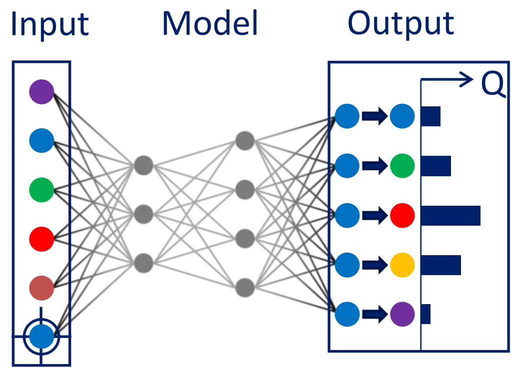
- Recursive layer to take time dependent information into account



# MODEL FITTING – CRITIC ONLY

NN inputs

- Wave encoded matrix
- Reward




NN output

- Vector of Q-value (estimated cumulative reward for each action)

Training

- Prediction $\hat{y} = Q(s_t,a_t)$
- Target $y = r_t+\gamma.\max_a Q(s_{t+1},a)$
- Loss function $MSE(y,\hat{y}) = \frac{1}{n}\sum(y-\hat{y})^2$

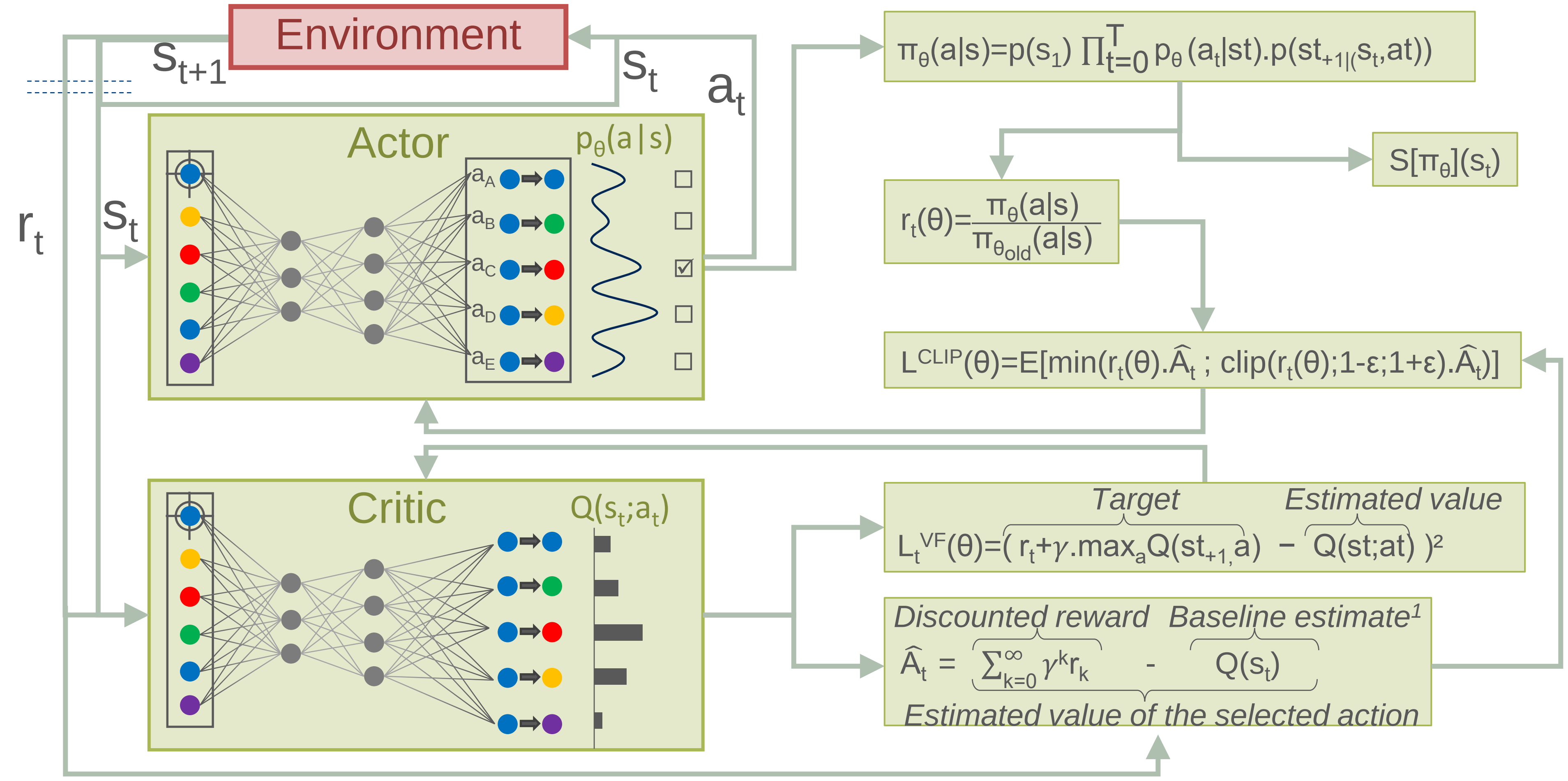
Estimated cumulative reward one step ahead
(lower uncertainty than ŷ)

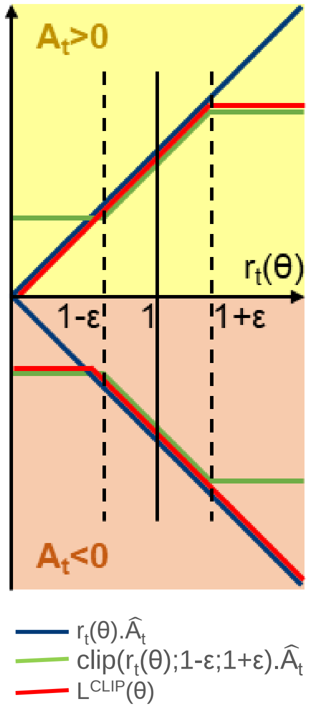
Action *a*    State *s*

$$Q(s_t,a;\theta)=E_{a\sim\pi_\theta(.|s)}[\sum_{k=0}^{\infty}\gamma^k r_k]$$

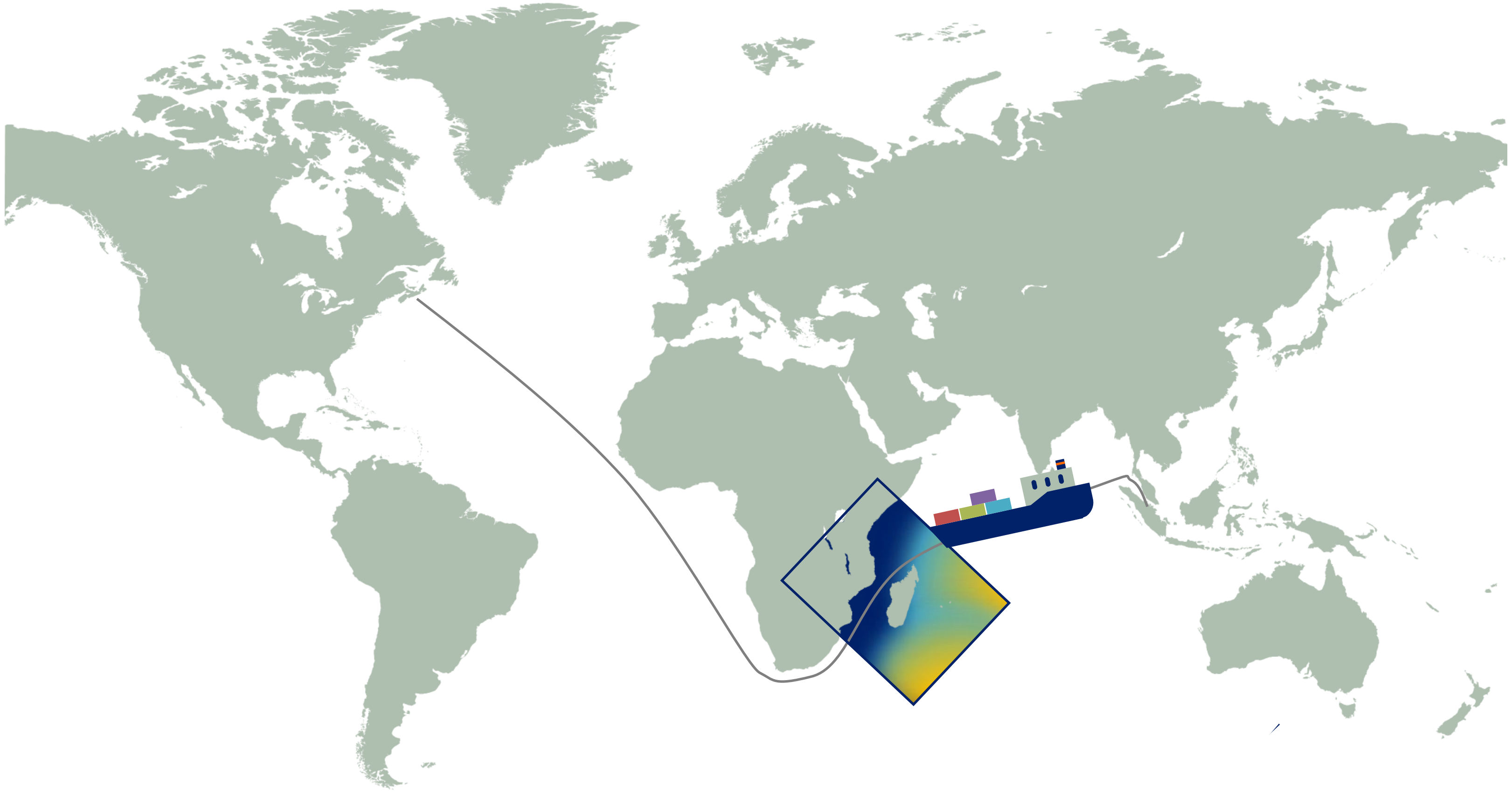
Discount factor *γ*    Reward *r*

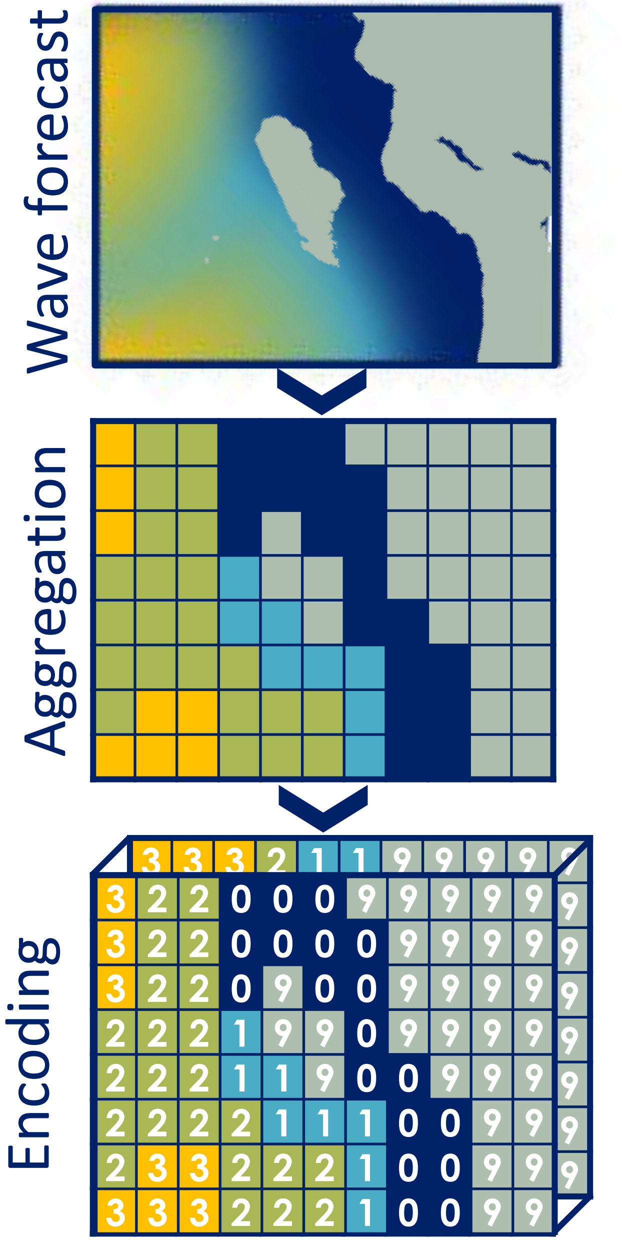
# MODEL FITTING – ACTOR-CRITIC



# WAVE CONDITIONS FORECAST

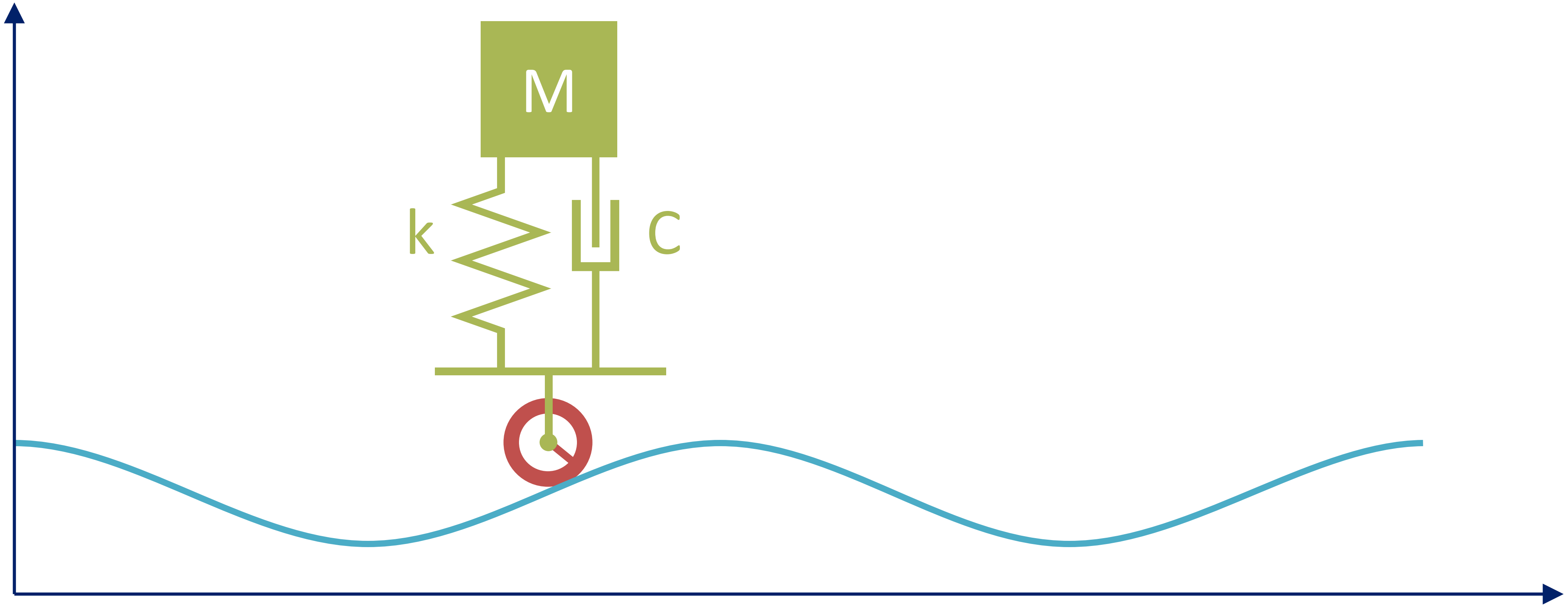





Observation = environment description in a given state

# MOTION RESPONSE MODELIZATION

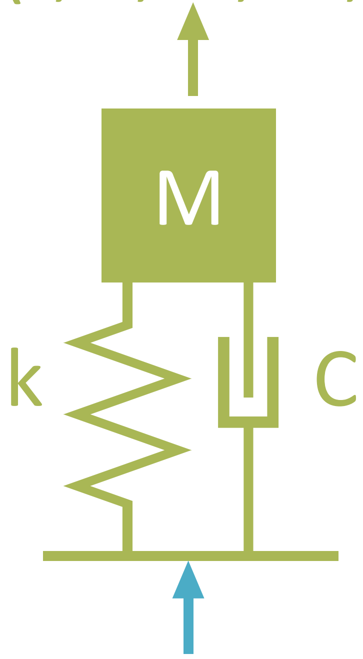
# MOTION RESPONSE MODELIZATION



# MOTION RESPONSE MODELIZATION

Ship characteristics and loading

$$f(t, a, \omega, m, c, k) = A.\sin(\omega.t+\phi)$$




$$h(t) = a.\sin(\omega.t)$$

Wave conditions forecast

$$x(t) = \frac{a\sqrt{k^2 + (c\omega)^2}}{\sqrt{(k - m\omega)^2 + (c\omega)^2}} \sin(\omega t - \alpha - \phi_1)$$

$$\phi_1 = \tan^{-1}\left[\frac{c\omega}{k - m\omega^2}\right]$$