

Reinforcement Learning for Options on Target-Volatility Funds



Speaker: **Stefano POLO** from Intesa Sanpaolo
stefano.polo@intesasanpaolo.com

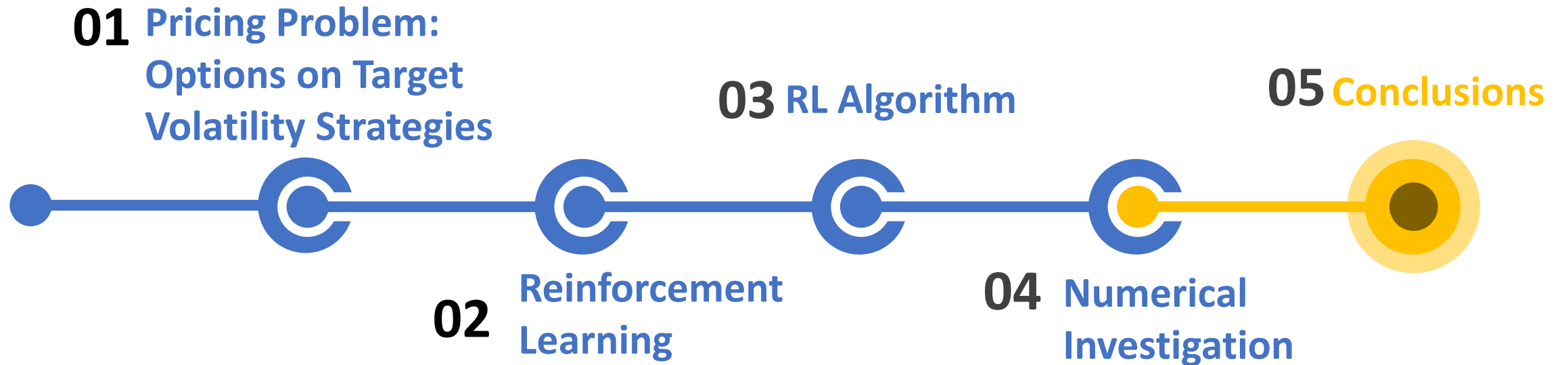
Joint work with: **Roberto DALUIO** from Intesa Sanpaolo
Emanuele NASTASI from Marketz S.p.A.
Andrea PALLAVICINI from Intesa Sanpaolo

Special thanks to: **Marco BIANCHETTI** and **Diego Pierluigi GIOVANNINI**

Polimi Fintech
June 10, 2021



Presentation Contents




Target Volatility Strategy

A **target volatility strategy (TVS)** is a portfolio of risky assets (typically equities) and a risk-free asset dynamically re-balanced with the aim of maintaining the overall portfolio volatility level closed to some target value $\bar{\sigma}$

$$\frac{dI_t}{I_t} = \underbrace{\omega_t \alpha_t \cdot \frac{dS_t}{S_t}}_{\substack{\text{Risky component} \\ n\text{-dimensional vector}}} + (1 - \underbrace{\omega_t \alpha_t \cdot \mathbb{I}}_{\substack{\text{Risk-free} \\ \text{component}}}) \frac{dB_t}{B_t}$$

- ❖ \mathbb{I} n -dimensional vector with unitary entries
- ❖ ω_t **automatic volatility targeting algorithm**
- ❖ α_t **time-dependent allocation strategy** for the risky-asset portfolio

The Pricing Problem

- We consider the point of view of a **bank selling a protection** to a portfolio manager on the capital invested in a TVS.
- The manager buys a **plain vanilla call option** on the TVS to ensure capital protection.
- The portfolio **manager** has the **freedom of changing the relative weights** α_t of the risky assets during the life of the TVS
- The risky assets have different hedging costs  **the bank shall adjust the price** of the protection to include them in the worst case scenario

$$V_0 := \sup_{\alpha} \mathbb{E}[D_0(T)(I_T(\alpha) - K)^+]$$

Pricing problem becomes a stochastic control problem: finding the optimal strategy α_t^* which maximizes the option price

TVS Dynamics

- We consider a fund trading a basket of n risky assets with price process S_t funded with a cash account B_t accruing at r_t .

$$\frac{dI_t}{I_t} = \omega_t \alpha_t \cdot \frac{dS_t}{S_t} + (1 - \omega_t \alpha_t \cdot \mathbb{I}) \frac{dB_t}{B_t}$$

- We assume a generic semi-martingales dynamics under the risk-neutral measure for the securities

$$\frac{dS_t}{S_t} = (r_t - \mu_t)dt + \nu_t \cdot dW_t$$

❖ μ_t : hedging costs of the assets

❖ ν_t : diffusion matrix $\left\{ \begin{array}{l} \text{Time-dependent Black and Scholes (BS): } \nu_t := \nu(t) \\ \text{Local Volatility model (LV): } \nu_t := \nu(t, S_t) \end{array} \right.$

TVS Dynamics

$$\frac{dI_t}{I_t} = \omega_t \alpha_t \cdot \frac{dS_t}{S_t} + (1 - \omega_t \alpha_t \cdot \mathbb{I}) \frac{dB_t}{B_t}$$

$$\frac{dS_t}{S_t} = (r_t - \mu_t)dt + v_t \cdot dW_t$$

ω_t automatic volatility targeting algorithm: $Var_t[dI_t] = \bar{\sigma} I_t^2 dt \longrightarrow \omega_t = \frac{\bar{\sigma}}{\|\alpha_t \cdot v_t\|}$

$$\frac{dI_t}{I_t} = \left(r_t - \bar{\sigma} \frac{\alpha_t \cdot \mu_t}{\|\alpha_t \cdot v_t\|} \right) dt + \frac{\bar{\sigma} \alpha_t}{\|\alpha_t \cdot v_t\|} \cdot v_t \cdot dW_t$$

Optimal Strategy Closed Solution

- In the case of an European payoff we are able to prove that exists an optimal strategy

$$V_0 := \sup_{\alpha} \mathbb{E}[D_0(T)(I_T(\alpha) - K)^+]$$

- We get the Markovian projection (MP) of the dynamics followed by the TVS by applying the Gyöngy Lemma

$$\frac{dI_t^{MP}}{I_t^{MP}} := [r_t - \ell_{\alpha}(t, I_t^{MP})]dt + \bar{\sigma}dW_t^{MP}$$
$$\ell_{\alpha}(t, K) := \bar{\sigma}\mathbb{E}\left[\frac{\alpha_t \cdot \mu_t}{\|\alpha_t \cdot \nu_t\|} | I_t = K\right]$$

- Since European payoffs depend only on the marginal distribution at maturity, we can compute the option price by means of the MP

$$V_0 := \sup_{\alpha} \mathbb{E}[D_0(T)(I_T^{MP}(\alpha) - K)^+]$$

Optimal Strategy Closed Solution

- In the Black and Scholes model with deterministic rates, the TVS dynamics is Markovian. As consequence the local drift can be written as

$$\ell_{\alpha}(t, K) := \bar{\sigma} \frac{\alpha(t, K) \cdot \mu(t)}{\|\alpha(t, K) \cdot \nu(t)\|}$$

- The optimization problem can be solved by looking only at the MP with no simulation needed. The optimal strategy for a non-decreasing European payoff consists in minimizing the local drift function, independently on the current state I_t :

$$\alpha^*(t) := \underset{\alpha}{\operatorname{argmin}} \frac{\alpha \cdot \mu(t)}{\|\alpha \cdot \nu(t)\|}$$

- In absence of constraint, even a closed form formula is available:

$$\alpha_{free}^*(t) = - \frac{\Sigma^{-1}(t) \cdot \mu(t)}{\|(\Sigma^{-1}(t) \cdot \mu(t)) \cdot \nu(t)\|} \quad \text{with } \Sigma(t) = \nu(t)\nu^T(t)$$

The Black and Scholes Model

$$\alpha^*(t) := \operatorname{argmin}_{\alpha} \frac{\alpha \cdot \mu(t)}{\|\alpha \cdot \nu(t)\|}$$

- **Absence of stochastic elements** for the estimation of the optimal allocation strategy ➡ *a priori* knowledge of α^*
- Problem is solved just by looking at the market data $\mu(t)$ and $\nu(t)$ for each pillar and assuming a piecewise constant dependence on time.
- Once the optimal strategy $\alpha^*(t)$ is known, then we can price the payoff by means of a Black and Scholes formula

$$V_0^{BS} = BS(F^{TVS}(0, T; \alpha^*), K, T, \bar{\sigma}, D(0, T; \zeta))$$

$$F^{TVS}(t, T; \alpha^*) := I_t \exp \left[\int_t^T (r(u) - \ell_{\alpha^*}(u)) du \right]$$

Hamilton-Jacobi-Bellman Problem for TVO

$$\begin{cases} dX_t = M(X_t)dt + \Sigma(X_t) \cdot dW_t \\ \frac{dI_t}{I_t} = \left(r_t - \bar{\sigma} \frac{\alpha_t \cdot \mu_t}{\|\alpha_t \cdot v_t\|} \right) dt + \frac{\bar{\sigma} \alpha_t}{\|\alpha_t \cdot v_t\|} \cdot v_t \cdot dW_t \end{cases}$$

The Hamilton-Jacobi-Bellman for the optimal derivative price $V := V(I_t, X_t, t)$, naming $I := I_t$ and $X := X_t$

$$\begin{aligned} & \frac{\partial V}{\partial t} + \frac{\partial V}{\partial I} I r_t + \cancel{(\nabla_X V) \cdot M(X)} + \frac{1}{2} \frac{\partial^2 V}{\partial I^2} \bar{\sigma} I^2 + \frac{1}{2} \cancel{\text{Tr}(\Sigma(X)^T (H_X V) \Sigma(X))} \\ & + \bar{\sigma} I \max_{\alpha} \left\{ -\frac{\partial V}{\partial I} \frac{\alpha \cdot \mu_t}{\|\alpha \cdot v_t\|} + \cancel{(\nabla_{X,I} V) \cdot \Sigma(X)^T \cdot \left(\frac{\alpha \cdot \mu_t}{\|\alpha \cdot v_t\|} \right)} \right\} = 0 \end{aligned}$$

$\nabla_X V$: Gradient of V w.r.t. X

$H_X V$: Hessian matrix of V w.r.t. X

$$\nabla_{X,I} V := \left(\frac{\partial^2 V}{\partial X^1 \partial I}, \dots, \frac{\partial^2 V}{\partial X^n \partial I} \right)$$

Time-dependent Black and Scholes: $V := V(I_t, t)$



$$\alpha^*(t) := \underset{\alpha}{\operatorname{argmin}} \frac{\alpha \cdot \mu(t)}{\|\alpha \cdot v(t)\|}$$

Hamilton-Jacobi-Bellman Equation

$$\begin{cases} dX_t = M(X_t)dt + \Sigma(X_t) \cdot dW_t \\ \frac{dI_t}{I_t} = \left(r_t - \bar{\sigma} \frac{\alpha_t \cdot \mu_t}{\|\alpha_t \cdot v_t\|} \right) dt + \frac{\bar{\sigma} \alpha_t}{\|\alpha_t \cdot v_t\|} \cdot v_t \cdot dW_t \end{cases}$$

The Hamilton-Jacobi-Bellman for the optimal derivative price $V := V(I_t, X_t, t)$, naming $I := I_t$ and $X := X_t$

$$\begin{aligned} & \frac{\partial V}{\partial t} + \frac{\partial V}{\partial I} I r_t + (\nabla_X V) \cdot M(X) + \frac{1}{2} \frac{\partial^2 V}{\partial I^2} \bar{\sigma} I^2 + \frac{1}{2} \text{Tr}(\Sigma(X)^T (H_X V) \Sigma(X)) \\ & + \bar{\sigma} I \max_{\alpha} \left\{ -\frac{\partial V}{\partial I} \frac{\alpha \cdot \mu_t}{\|\alpha \cdot v_t\|} + \left(\nabla_{X,I} V \right) \cdot \Sigma(X)^T \cdot \left(\frac{\alpha \cdot \mu_t}{\|\alpha \cdot v_t\|} \right) \right\} = 0 \end{aligned}$$

$\nabla_X V$: Gradient of V w.r.t. X

$H_X V$: Hessian matrix of V w.r.t. X

$$\nabla_{X,I} V := \left(\frac{\partial^2 V}{\partial X^1 \partial I}, \dots, \frac{\partial^2 V}{\partial X^n \partial I} \right)$$

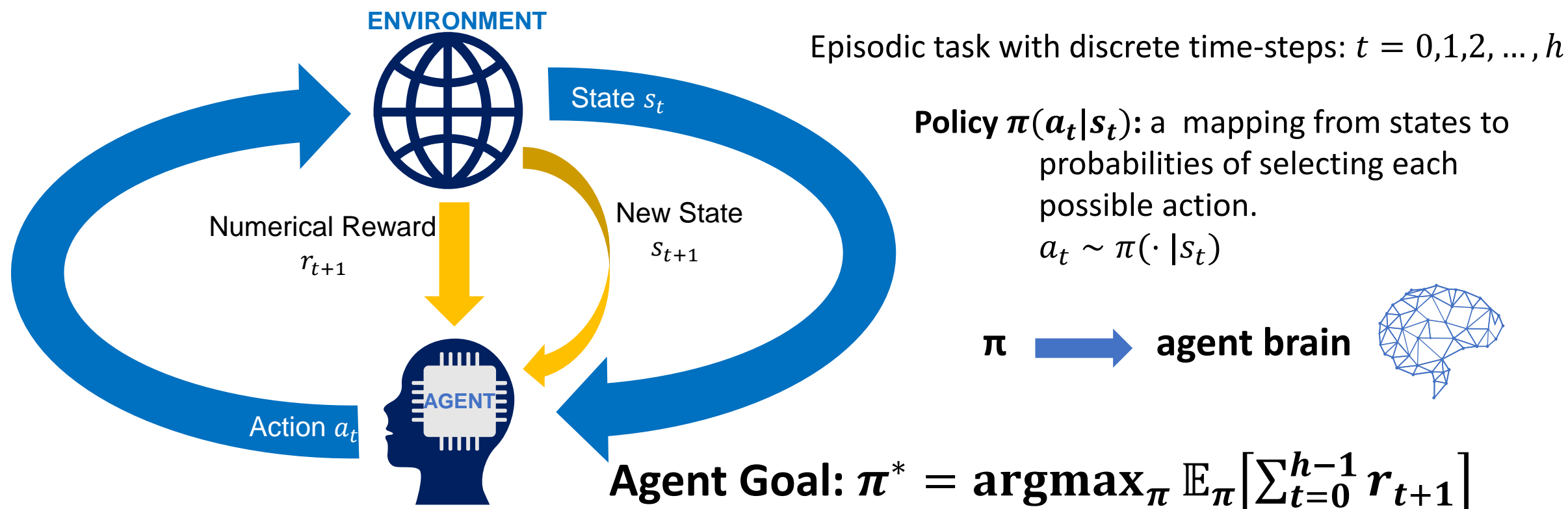
Local Volatility: $v_t := v(t, S_t)$
second order term. No closed
formula available!




numerical approach:
Reinforcement Learning

Reinforcement Learning

Reinforcement Learning describes how an agent behaves in a branch of Machine Learning which describes how an agent interacts with an environment so to maximize some notion of cumulative reward



Reinforcement Learning

- No *a priori* knowledge of the model  **sample the system** to gather statistical knowledge.
- **The agent** is not told which actions to take but instead it must discover **by trial and error** which are the behaviours yielding to the greatest reward by trying them several times.
- Challenges that are not present in other kinds of learning:
 - ❖ temporal credit assignment;
 - ❖ exploration-exploitation trade-off.
- Wide taxonomy of the modern Reinforcement Learning landscape based on the question of what to learn.

We will focus on Policy Gradient Algorithms

RL Policy Gradient Algorithm

Direct estimation of π^* (Direct policy search)



Parameterization of the policy π_θ by a set of parameters $\theta \in \Theta \subseteq \mathbb{R}^{D_\theta}$



Optimization of θ as to maximize the objective function $L(\theta)$

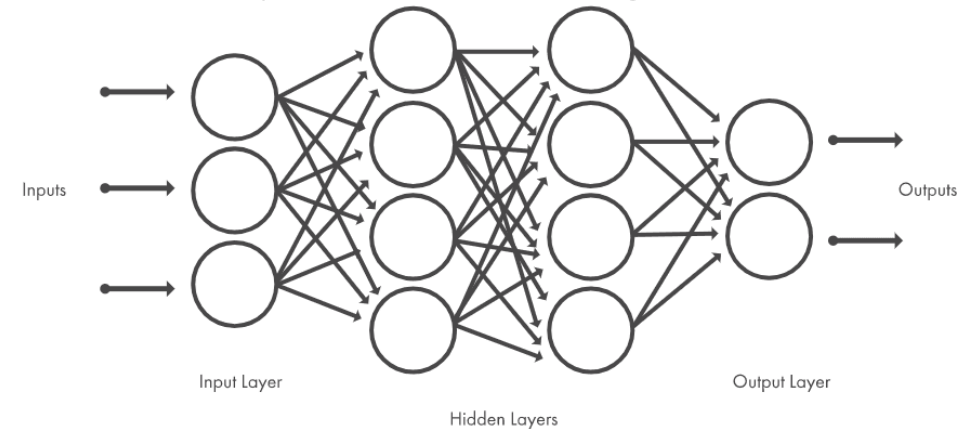
$$\theta^* = \underset{\theta \in \Theta}{\operatorname{argmax}} L(\theta)$$

$$\theta_{k+1} = \theta_k + \xi_k \nabla_{\theta} L(\theta)|_{\theta_k} \text{ with } \xi_k \text{ learning rate}$$

The definition of the objective function $L(\theta)$ depends on the learning strategy adopted



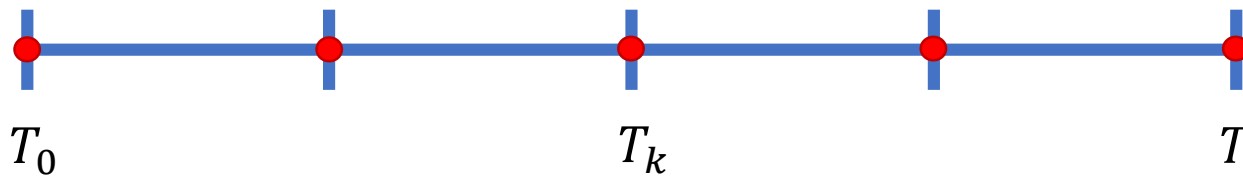
Artificial Neural Network (ANN):
 θ corresponds to the weights of the ANN



Proximal Policy Optimization (PPO)
algorithm (Schulman et al., 2017)

RL Algorithm: State Space

- **Episode:** the episode takes place on a time grid $\mathfrak{T} := \{T_0, \dots, T_k, \dots, T_m\}$ with $T_0 = 0$ and $T_m = T$

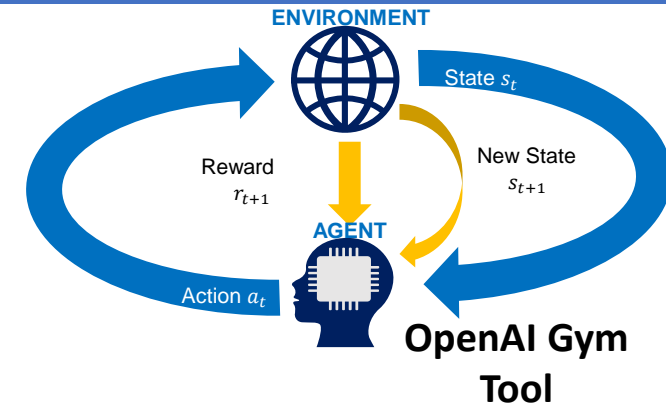


- **State:** $s_k = [T_k, I_{T_k}, S_{T_k}] \quad \forall T_k \in \mathfrak{T}$, so that the state space is $(\mathbb{R}^+)^{n+1} \times [0, T]$.

❖ The state is the input of the policy ANN ➡ need **input data normalization**

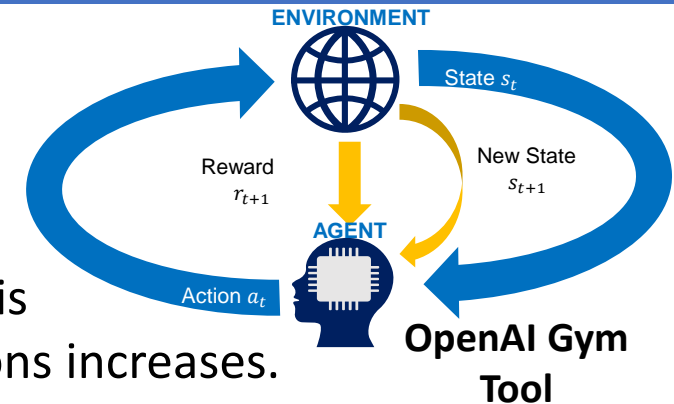
$$X_t = \frac{\log\left(\frac{S_{T_k}}{F(0, T_k)}\right) + \frac{1}{2} \int_0^{T_k} (v^2(t) \cdot \mathbb{I}) dt}{\sqrt{\int_0^{T_k} (v^2(t) \cdot \mathbb{I}) dt}}$$

- **Normalized state space:** $[-2.5, 2.5]^n \times [0, 1] \times [0, T]$



RL Algorithm: Action Space

- **Agent Policy:** the asset allocation weights α_t
- The policy is a n -dimensional Diagonal Gaussian Distribution where **the mean vector $\mu_\theta(s_t)$ is the output of the ANN** and the **log-standard deviation $\log \sigma$** is an **independent parameter** which is reduced as the number of the PPO iterations increases.



- We study the case of long only positions in the strategy:
$$\sum_{i=1}^n \alpha_t^i = 1 \quad \text{with} \quad \alpha_t^i > 0$$

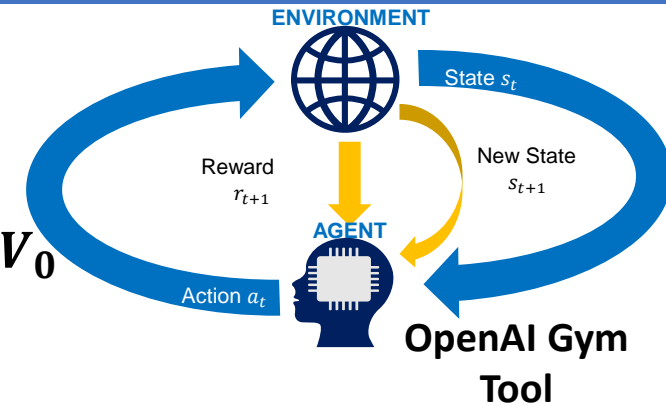
❖ **Action normalization:** bound the actions domain in $[0,1]^n$ and then normalize the action such as it is satisfied the long position constraint:

$$\alpha_t := \frac{a_t}{\sum_{i=1}^n a_t^i}$$

RL Algorithm: Reward

We have implemented two different reward functions to train the agent

$$1. \text{ Reward: } r_{k+1} := \begin{cases} D_0(T)(I_T - K)^+ & \text{if } T_{k+1} = T \\ 0 & \text{otherwise} \end{cases} \longrightarrow \mathbb{E}_{\pi_{\theta^*}} [\sum_{k=0}^{m-1} r_{t+1}] \sim V_0$$



$$2. \text{ Reward: } r_k := \gamma^k [V_{BS}(T_{k+1}) - V_{BS}(T_k)]$$

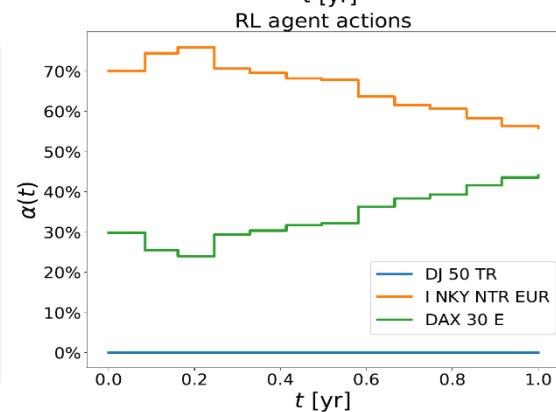
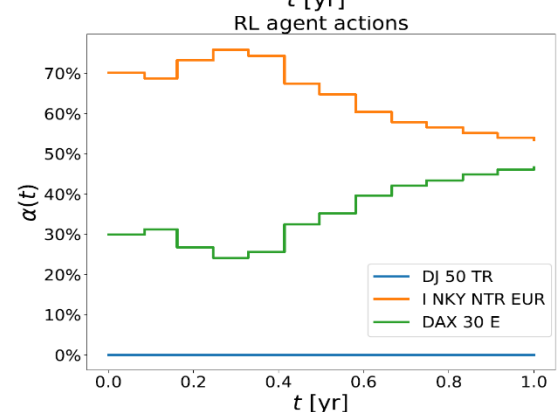
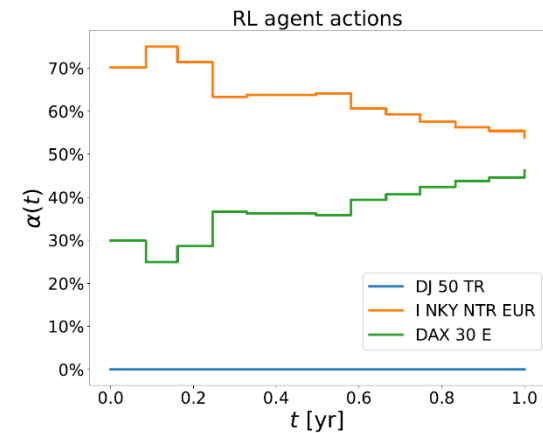
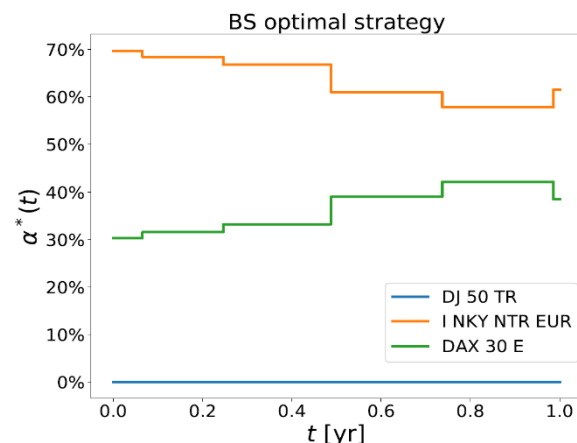
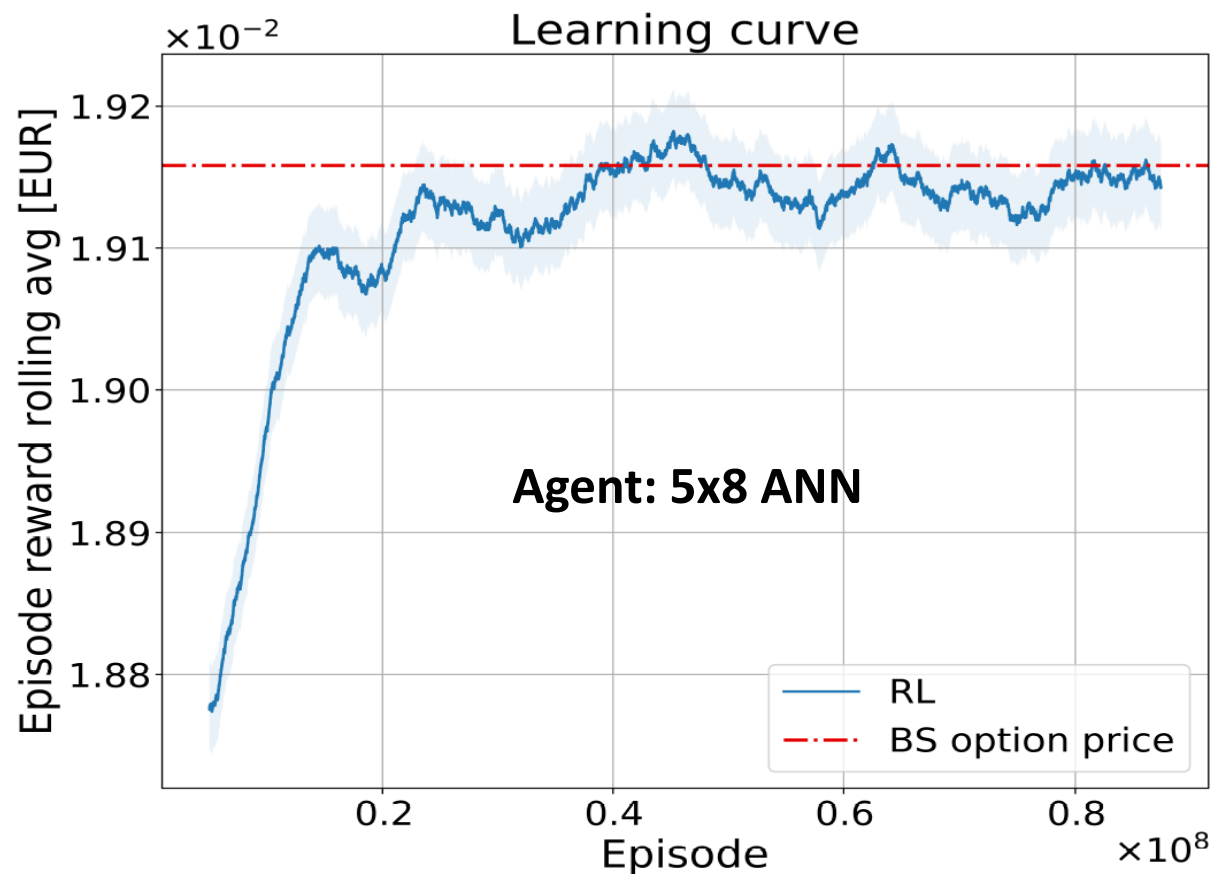
where $V_{BS}(T_k) := BS(F^{TVS}(T_k; T; \alpha_{BS}^*), K, T - T_k, \bar{\sigma}, D(T_k, T, \zeta))$ and $V(T_0) = 0$

- This kind of function introduces intermediate rewards to help the agent to take an optimal choice
- γ is a reward discount factor; if we choose $\gamma = 1$ then we have

$$\sum_{k=0}^{m-1} r_{t+1} = \sum_{k=0}^{m-1} [V_{BS}(T_{k+1}) - V_{BS}(T_k)] = V_{BS}(T) = (I_T - K)^+$$

Numerical Result: BS Dynamics

Option Contract: $I_0 = K = 1$ [EUR], $T = 1$ [yr], $\bar{\sigma} = 5\%$

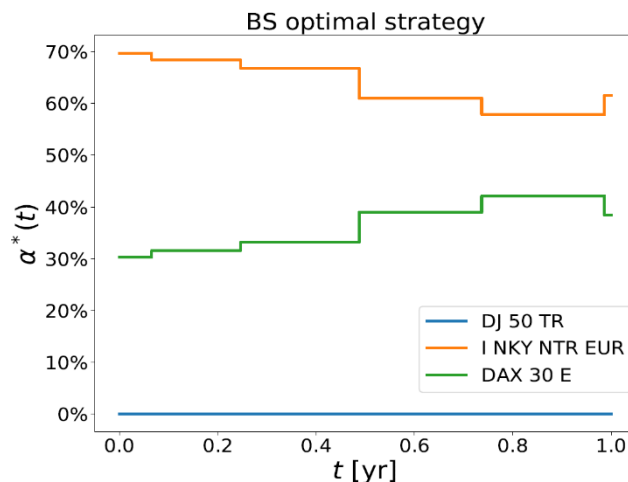
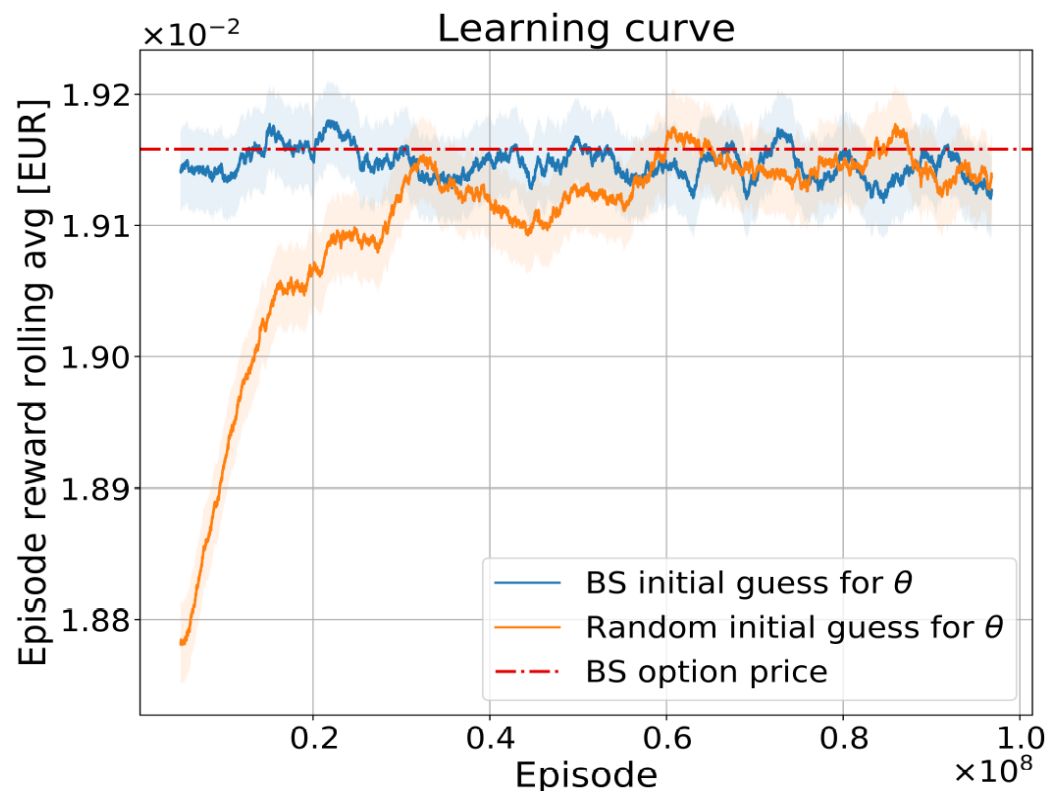


BS price: $\Pi_0^* = 1.915 \times 10^{-2}$ [EUR]

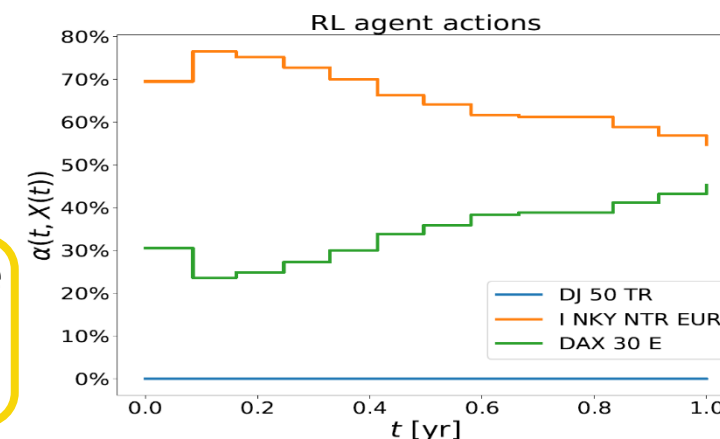
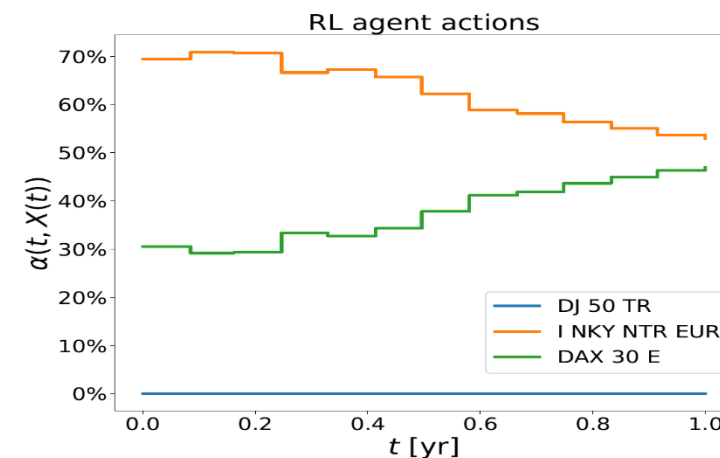
MC price: $\Pi_0^{\text{RL}} = (1.914 \pm 0.003) \times 10^{-2}$ [EUR]

Numerical Results: LV Dynamics

- The volatility term is function not only of time t but also of the state S_t → **no a priori optimal allocation strategy**
- The price process dynamics model changes but not the market data conditions: two different way to train the agent
 - **Random initialization** of the ANN weights θ
 - **BS initial guess**: same θ of the ANN trained in the BS environment



The agent seems to recover the same optimal strategy as in the BS environment



Numerical Results: the Baseline

- We can try to understand if the agent has reached a suboptimal policy by building a naïf strategy to measure the RL performance. We call this strategy: Baseline
- Baseline: intuitive strategy that applies path-wise the BS solution of maximizing the local drift of the TVS

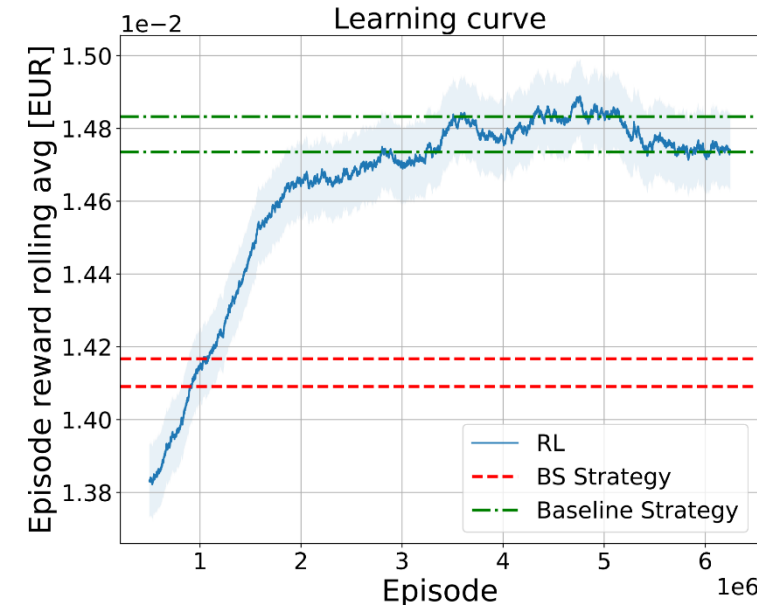
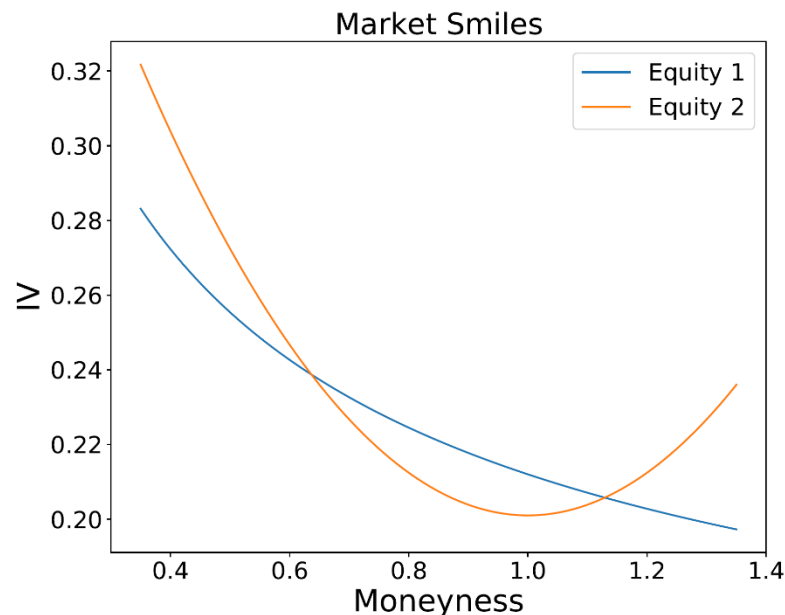
$$\alpha^*(t, S_t) := \operatorname{argmin}_{\alpha} \frac{\alpha \cdot \mu(t)}{\|\alpha \cdot \nu(t, S_t)\|}$$

Strategy	Option price [EUR]
RL from random θ	$(1.912 \pm 0.003) \times 10^{-2}$
RI from BS θ	$(1.915 \pm 0.003) \times 10^{-2}$
Baseline θ	$(1.915 \pm 0.004) \times 10^{-2}$

Market data display a Black and Scholes market

Numerical Results: the Baseline

- We build a toy market to investigate the control problem in LV dynamics. We choose two assets with equal $\mu(t)$ but different implied volatilities.
- In this way the BS optimal allocation strategy will be completely different from the LV optimal one since it is only sensitive to the atm implied volatility.
- We train the RL agent in this market parameterizing the action through the baseline



Conclusions



Option on targetvol: control problem

We showed how the presence of different funding costs coming from hedging the risky assets underlying the TVS needs to solve a control problem to find the conservative option price.



Black and Scholes Dynamics

We show how under BS dynamics we are able to recover a closed form solution for the control problem by applying the Gyöngy Lemma and the Hamilton-Jacobi-Bellman equation



Reinforcement Learning

- **Black and Scholes:** the availability of an a priori solution allowed us to test if the RL agent was able to recover the optimal policy. Moreover it provided us the possibility to perform fine tuning of the RL hyper-parameters
- **Local Volatility:** no a priori solution. The agent is able to find as optimal strategy the baseline one: the application of a path-wise BS solution



Future Development

Compare the RL results with standard numerical techniques based on backward recursions such as least square Monte Carlo