Ca' Foscari
University
of Venice

Department
of Economics

Working Paper

Marco Corazza

and Francesco Bertoluzzo

Q-Learning-based financial trading systems with applications

ISSN: 1827-3580
No. 15/WP/2014

Working Papers
Department of Economics
Ca' Foscari University of Venice
No. 15/WP/2014
ISSN 1827-3580

Q-Learning-based financial trading systems with applications

Marco Corazza
Department of Economics
Ca' Foscari University of Venice

Francesco Bertoluzzo
Department of Economics
Ca' Foscari University of Venice

First Draft: October 2014

Abstract
The design of financial trading systems (FTSs) is a subject of high interest both for the academic environment and for the professional one due to the promises by machine learning methodologies. In this paper we consider the Reinforcement Learning-based policy evaluation approach known as *Q*-Learning algorithm (*Q*La). *Q*La is an algorithm which real-time optimizes its behavior in relation to the responses it gets from the environment in which it operates. In particular: first we introduce the essential aspects of *Q*La which are of interest for our purposes; second we present some original FTSs based on differently configured *Q*Las; then we apply such FTSs to an artificial time series of daily stock prices and to six real ones from the Italian stock market belonging to the FTSE MIB basket. The results we achieve are generally satisfactory.

Keywords
Financial trading system, Reinforcement Learning, *Q*-Learning algorithm, daily stock price time series, FTSE MIB basket.

JEL Codes
C61, C63, G11.

Address for correspondence:
Marco Corazza
Department of Economics
Ca' Foscari University of Venice
Cannaregio 873, Fondamenta S.Giobbe
30121 Venezia - Italy
Phone: (++39) 041 2346921
Fax: (++39) 041 2349176
E-mail: corazza@unive.it

The Working Paper Series
is availble only on line
(www.dse.unive.it/pubblicazioni)
For editorial correspondence, please contact:
wp.dse@unive.it

Department of Economics
Ca' Foscari University of Venice
Cannaregio 873, Fondamenta San Giobbe
30121 Venice Italy
Fax: ++39 041 2349210

1 Introduction

In accordance with the weak form of the Efficient Market Hypothesis (EMH), it is not possible to systematically make profitable trading in financial markets. In fact, following this theory, the economic agents acting in such markets are fully rational, that is, through the law of the demand and the supply, they are able to instantaneously and appropriately vary the prices of the financial assets on the basis of the past and the current information. In this theoretical framework, the only source of (unpredictable) variations of the prices of the financial asset between two consecutive time instants can be the arrival of unexpected new information. This last point is generally formalized as

$$P_{t+1} = \mathbb{E}\left(\widetilde{P}_{t+1}\middle|\,\Omega_t\right) + \widetilde{\varepsilon}_{t+1},$$

where t and $t+1$ indicate the two consecutive time instants, P_τ indicates the price of a given financial asset at the time instant τ, $\mathbb{E}(\cdot)$ indicates the expectation operator, $\widetilde{P}_{t+1}$ indicates the random variable "price of the given financial asset at the time instant $t+1$", Ω_t indicates the set of the information available at the time instant t, and $\widetilde{\varepsilon}_{t+1}$ indicates the random variable "prediction error of the price of the given financial asset at the time instant $t+1$", with $\mathbb{E}(\widetilde{\varepsilon}_{t+1}) = 0$ (see for instance [7]).

But, as common sense suggests, human beings (and therefore economic agents) are often non rational when making decisions, especially if under uncertainty. In fact, since the 80s of the past century experimental economists have documented several departures of the real investors' behaviours from the ones prescribed by the EMH (see for instance [8], [13], [14] and the references therein). The main implication coming from these departures from the EMH consists in the fact that financial markets are not so rarely inefficient, and consequently that they more or less frequently offer possibilities of profitable trading.

At this point an important question arises: How to take advantage of these possibilities of trading? Of course, the answer depends on the chosen reference theoretical framework. In our opinion, the currently most convincing attempt to reconcile the EMH with the empirical departures from it is given by the so-called Adaptive Market Hypothesis (AMH). Following this theory, a financial market can be viewed as an evolutionary

Electronic copy available at: http://ssrn.com/abstract=2507826

environment in which different intelligent but partly rational "species" (for instance, hedge funds, market makers, pension funds, retail investors, ...) interact among them in accordance with unknown and structurally time-varying dynamics in order to achieve the efficiency (see for more details [13] and [14]). Note that, since these species are partly (and not fully) rational, this evolutionary tending towards efficiency is not instantaneous and that it generally does not imply appropriate variations of the financial asset prices. Because of that, the AMH entails that ≪[f]*rom an evolutionary perspective, the very existence of active liquid financial markets implies that profit opportunities must be present. As they are exploited, they disappear. But new opportunities are also constantly being created as certain species die out, as others are born, and as institutions and business conditions change*≫ (from [13], page 24). So, coming back to the above question, it seems reasonable that an effective financial trading system (FTS) has to be a new specie able to real-time interact with the considered financial market in order to learn its unknown and structurally time-varying dynamics, and able to exploit this knowledge in order to real-time detecting profitable financial trading policies.

Therefore, given the reference theoretical framework we have chosen (i.e. the AMH) and given the features of the FTS we have required, in this paper we resort to a self-adaptive machine learning methodology known as Reinforcement Learning (*RL*) (see for instance [1]) in order to develop such a FTS. This methodology is also known as Neuro-Dynamic Programming (see for instance [4]) and as Dynamic Programming Stochastic Approximation (see for instance [10]). In short, this kind of learning concerns an agent (in our case the FTS) dynamically interacting with an environment (in our case a financial market). During this interaction the agent perceives the state of the environment and undertakes a related action (in our case to sell or to buy a given asset). In its turn, the environment, on the basis of this action provides a negative or a positive reward (in our case some measure of the investor's loss or gain). *RL* consists of the on-line detection of a policy (in our case a trading strategy) that permits the maximization over time of the cumulative reward (see sub-Section 3.1).

More specifically, we consider the *RL*-based policy evaluation approach known as *Q*-Learning algorithm (*Q*La) (see sub-Section 3.2). This implementation of the consid-

ered FTS can be viewed as a stochastic optimal control problem in which the *Q*La has to discover the optimal financial trading strategies. Note that the *RL* approaches, consequently also the *Q*La, do not provide optimal solutions but good near-optimal ones. So, at this point another important question arises: Why not resort to more classical stochastic dynamic programming methods guaranteeing the achievement of optimal solutions? Generally speaking, the latter typology of methods needs the precise description of the probabilistic features of the investigated financial market. But, as we state below, once chosen AMH as reference theoretical framework, the dynamics of financial markets are unknown, from which the impossibility of providing such a required precise description. Differently, ≪[t]*he so-called model-free methods of RL* [– among which the *Q*La –] *do not need the transition probability matrices*≫ (from [10], page 212).

The remainder of the paper is organized as follows. In the next Section we give a review of the literature on the *RL*-based FTSs and we describe the elements of novelties we present in our paper with respect to this literature. In Section 3 we synthetically present the essential aspects of the *RL* methodology and of the *Q*La which are of interest to our purposes. In Section 4 we present our *Q*La-based FTSs and we provide the results of their applications to an artificial time series of daily stock prices and to six real ones from the Italian stock market belonging to the FTSE MIB basket. In section 5 we give some concluding remarks.

2 A review of the literature and our elements of novelties

In this Section, first we provide a review of the literature about the *RL*-based FTSs. Our primary purpose is not to be exhaustive, but rather to highlight the main research directions. Then we describe the elements of novelties present in our paper.

Among the first contributions in this research field, we recall [15], [16], and [9]. In general, the respective Authors show that *RL*-based financial trading policies perform better than those based on supervised learning methodologies when market frictions are considered. In [15] a version of the *RL* methodology called Direct Learning is proposed and used in order to set a FTS that, taking into account transaction costs, maximizes an appropriate investor's utility function based on a differential version of the known

Sharpe ratio. Then, it is shown by controlled experiments that the proposed FTS performs better than standard FTSs. Finally, the Authors use the so developed FTS to make profitable trades with respect to assets of the U.S. financial markets. In [16], the Authors mainly compare FTSs developed by using various *RL*-based approaches with FTSs developed by using stochastic dynamic programming methodologies. In general they show by extensive experiments that the former approaches are better than the latter ones. In [9] the Author considers a FTS similar to the one developed in [15] and applies it to the financial high-frequency data, obtaining profitable performances. Also in [3] the Authors take into account a FTS similar to the one developed in [15], but they consider an investor's utility function based on the differential version of the returns weighted direction symmetry index. Then, they apply this FTS to some of the most relevant world stock market indexes achieving satisfactory results. Subsequent to these initial contributions, other approaches have been investigated. In [6] the Authors empirically compare various FTSs based on different neurocomputing methods, among which a hybrid one constituted by the *Q*La combined with a supervised Artificial Neural Network (sANN). They apply these FTSs to a simulated and to two real time series of asset prices finding that the QL-sANN-based FTS shows enough good performances, although generally not better than those of the other FTs. Also in [19] the Authors consider a hybrid method constituted by the Adaptive Network Fuzzy Inference System (ANFIS) supplemented by the *RL* paradigm. They apply the so developed FTS to five U.S.A. stocks over a period of 13 years, achieving profitable performances. In [17] the Authors proposes a *RL*-based asset allocation strategy able to utilize the temporal information coming from both a given stock and the fund over that stock. Empirical results attained by applying such asset allocation strategy to the Korean stock market show that it performs better than several classical asset allocation strategies. In [11] two stock market timing predictors are presented: an actor-only *RL* and and actor-critic *RL*. The Authors show that, when both are applied to real financial time series, the latter generally perform better than the former. In [2] an actor-critic *RL*-based FTS is proposed, but in a fuzzy version. The Authors show that, taking into account transaction costs, the profitability of this FTS when apply to important world stock market indexes is consis-

tently superior to that of other advanced trading strategies. Finally, in [12] the Authors use different *RL*-based high-frequency trading systems in order to optimally manage aspects like the data structure of the individual orders and the trade execution strategies. Their results are generally satisfactory but show anyway that when taking into account the trading costs, profitability is more elusive.

With respect to the this literature, in this paper we do the following:

- As introduced in Section 1, we consider the *RL*-based policy evaluation approach known as *Q*La. Generally, the squashing function used in the *Q*LA is the known S-shaped logistic. In this paper we substitute it with another known S-shaped squashing function, the hyperbolic tangent, in order to check the performances coming from this substitution. To the best of our knowledge, the hyperbolic tangent has been rarely (if not) used in this context;
- In several papers the classical Sharpe ratio and some its variants are used as reward functions. In this paper we utilize again the classical Sharpe ratio calculated over the last $L \in \mathbb{N}$ trading days – but simply as generally accepted benchmark – and furthermore we utilize: The average logarithmic return calculated over the last L trading days; The ratio between the sum calculated over the last L trading days of the logarithmic returns and the sum calculated over the last L trading days of the absolute value of the logarithmic returns. By doing so, we aim to verify whether the widespread choice of the Sharpe ratio as reward function is founded or not. To the best of our knowledge, the two latter reward functions have never been used in this context;
- Usually, the very very big majority of the traditional FTSs and of the advanced ones consider two signals or actions: "sell" or equivalently "stay-short-in-the-market", and "buy" or equivalently "stay-long-in-the-market". In this paper we consider also a third signal or action: "stay-out-from-the-market". By doing so, we give to our *Q*La-based FTSs the possibility to take no position, or to leave a given position, when the buy/sell signal appears weak. Note that the set of the action is finite and discrete;

– As state variables describing the financial market we consider the current and some past returns. Generally, this is not the case for several FTSs to which more or less refined state variables are provided. We have made this choice in order to check the performance capability of our FTS also starting from basic information. Notice that any state variable is continuous over $\mathbb{R}$;
– Finally, in order to test the operational capability of QLa-based FTSs, we explicitly take into account the transaction costs.

3 Basics on the *RL* methodology and on the QLa

In this Section we synthetically recall the main formal aspects of the *RL* methodology and of the QLa. In particular, we limit ourselves to those aspects which are of interest to our purposes. See for any technical deepening [1], [4] and [10].

3.1 Basics on the *RL* methodology

Let us consider a discrete-time dynamical system modeling the environment. The information concerning the system at time t are summarized in the N-dimensional vector state $s_t \in \mathscr{S} \subseteq \mathbb{R}^N$, with $N \in \mathbb{N}$. In the *RL* methodology it is assumed that the system satisfies the Markov property (this property refers to the fact that the probability of transition from the current state to the next one depends only on the current state). On the basis of the state s_t, the agent selects an action $a_t \in \mathscr{A}(s_t)$, where $\mathscr{A}(s_t)$ is the set of the possible actions the agent can take given the state s_t. At time $t+1$ the agent receives a reward, $r(s_t, a_t, s_{t+1}) \in \mathbb{R}$, as consequence of her/his actions a_t and of the new state s_{t+1}. At time t the agent wish to maximize the expected value of some global reward $R(s_t)$. A common formulation of the global reward is

$$R(s_t) = \sum_{i=0}^{+\infty} \gamma^i r(s_{t+i}, a_{t+i}, s_{t+1+i}),$$

where $\gamma \in (0,1)$ is the discount factor.

Within this framework, the *RL* methodology searches for a policy $\pi(s_t) = a_t$, that is for a mapping from states to actions, which maximizes the expected value of $R(s_t)$; such

a policy is said optimal. In order to do this, the *RL* methodology needs the estimation of the so-called value function. It is a function of the state (or of state-action pair) which attributes a value to each state (or to each state-action pairs) proportional to the global rewards achievable from the current state s_t (or from the current state-action pairs (s_t, a_t)). In qualitative terms, the value function measures how good is for the agent to be in a given state (or how good is for the agent selecting a given action being in a given state). In particular, the value of the state $s_t = s$ under a policy $\pi(\cdot)$ is given by the value function $V^{\pi}(s_t)$: $V^{\pi}(s) = \mathbb{E}[R^{\pi}(s_t)|s_t = s]$. Similarly, the value of taking the action $a_t = a$ being in the state $s_t = s$ under a policy $\pi(\cdot)$ is given by the the value function $Q^{\pi}(s_t, a_t)$: $Q^{\pi}(s,a) = \mathbb{E}[R^{\pi}(s_t)|s_t = s, a_t = a]$.

A fundamental property of the value function is that it satisfies the relationship

$$V^{\pi}(s) = \mathbb{E}[r(s_t, \pi(s_t), s_{t+1}) + \gamma V^{\pi}(s_{t+1})|s_t = s] \ \forall \ \pi(\cdot) \text{ and } \forall \ s \in \mathscr{S}. \tag{1}$$

Equation (1) is the so-called Bellman equation for $V^{\pi}(s_t)$. An equivalent equation holds for the value function $Q^{\pi}(s_t, a_t)$. It is possible to prove that the value $V^{\pi}(s)$ is the unique solution to its Bellman equation. An equivalent result holds for the value $Q^{\pi}(s,a)$. It is also possible to prove that the optimal policy identifies the values $V^*(s)$ and $Q^*(s,a)$ such that

$$V^*(s) = \max_{\pi} V^{\pi}(s) \text{ and } Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a) \ \forall \ s \in \mathscr{S} \text{ and } \forall \ a \ \in \mathscr{A}(s). \tag{2}$$

Since $V^*(s)$ is a value function under a policy, it satisfies the Bellman equation (1). Further, since it is also the optimal value function, it is possible to prove that the Bellman equation for $V^*(s)$, called Bellman optimality equation, is

$$V^*(s) = \max_{a} Q^*(s,a) = \mathbb{E}[R^*(s_t)|s_t = s, a_t = a]. \tag{3}$$

Equation (3) expresses the fact that the value of a state under an optimal policy must equal the expected global reward for the best action from the state itself.

At this point, it is possible to iteratively compute the optimal value function for a state under a policy in the following way:

- Let $V_0^{\pi}(s_t)$ for all $s_t = s \in \mathscr{S}$ be an arbitrarily initialization of the state value function;
- Let $V_k^{\pi}(s_t)$ for all $s_t = s \in \mathscr{S}$ and with $k = 0, 1, \ldots$ be approximated by using as follows the Bellman equation:

$$\widehat{V}_{k+1}^{\pi}(s_t) = \mathbb{E}\left[r(s_t, a_t, s_{t+1}) + \gamma \widehat{V}_k^{\pi}(s_{t+1})\right], \tag{4}$$

where $\widehat{V}_k^{\pi}(s_t)$ indicates the estimation of $V_k^{\pi}(s_t)$. If the expectation $\widehat{V}_{k+1}^{\pi}(s_t)$ exists, then $\lim_{k \to +\infty} \widehat{V}_k^{\pi}(s) = V^{\pi}(s)$.

This iterative computation of the estimation of the value function $V_k^{\pi}(s_t)$ permits to find better and better policies able to increase the expected value of the global reward. It is possible to improve such a searching process by combining the recursive iteration (4) with one of the so-called improvement strategies. An important family of improvement strategies acts first by stopping at each step the recursive iteration (4), then by improving the actual policy on the basis of a pre-specified criterion. In general terms, the resulting process can be written as

$$\widehat{V}_{k+1}^{\pi_{New}}(s_t) = \max_a \mathbb{E}\left[r(s_t, a, s_{t+1}) + \gamma \widehat{V}_k^{\pi}(s_{t+1})\right], \tag{5}$$

where $\widehat{V}_{k+1}^{\pi_{New}}(s_t)$ indicates the update of the estimation of the value function under the new policy determined by the improvement strategy at step $k+1$ with respect to the previous estimation of the value function under the previous policy at step k. With regard to the improvement criterion, we have chosen an approach which may produce increasing of the global reward in the long run. Following this criterion, at each time t the action is determined as

$$a_t = \begin{cases} \pi'(s_t) & \text{with probability } 1 - \varepsilon \\ a \in \mathscr{A}(s_t) & \text{with probability } \varepsilon \end{cases},$$

where $\pi'(s_t)$ indicates the candidate action which maximizes $Q^{\pi}(s,a)$, and $\varepsilon \in (0,1)$.

3.2 Basics on the QLa

The QLa is a so-called off-policy control method which is widely used for calculating the expectation (5). The term "off" indicates that two different policies are used in

the process for finding better and better policies: A first one is used to estimate the expectation of the value function; A second one is used to manage the improvement strategy.

In order to give the main aspects of the QLa, let us start by putting in evidence that it is possible to write the estimation of the state value function $\widehat{V}_{k+1}(s_t)$ in the recursive way

$$\begin{aligned}\widehat{V}_{k+1}(s_t) = \alpha \sum_{j=1}^{k+1} R_j(s_t) = \alpha \left[R_{k+1}(s_t) + \sum_{j=1}^{k} R_j(s_t) \right] = \cdots = \\ = \widehat{V}_k(s_t) + \alpha \left[R_{k+1}(s_t) - \widehat{V}_k(s_t) \right],\end{aligned} \tag{6}$$

where $\alpha \in (0,\, 1)$ is the so-called learning rate. The QLa is able to update the estimation $\widehat{V}_{k+1}(s_t)$ as soon as the quantity

$$d_k = R_{k+1}(s_t) - \widehat{V}_k(s_t) = r(s_t, s_{t+1}) + \gamma \widehat{V}_k(s_{t+1}) - \widehat{V}_k(s_t),$$

becomes available. Given this quantity, the recursive relationship (6) can be rewritten as

$$\widehat{V}_{k+1}(s_t) = \widehat{V}_k(s_t) + \alpha \left[r(s_t, s_{t+1}) + \gamma \widehat{V}_k(s_{t+1}) - \widehat{V}_k(s_t) \right].$$

Note that α represents the percentage of d_k, that is of the "error", to add to the estimation of the state value function at the k-th step for achieving the estimation of the state value function at the $k+1$-th step. Similarly, it is possible to prove that the state-action value function is given by

$$\widehat{Q}_{k+1}(s_t, a_t) = \widehat{Q}_k(s_t, a_t) + \alpha \left[r(s_t, a_t, s_{t+1}) + \gamma \max_a \widehat{Q}_k(s_{t+1}, a_{t+1}) - \widehat{Q}_k(s_t, a_t) \right].$$

At this point we recall that the state variables of interest taken into account for the development of our FTSs are continuous over $\mathbb{R}$. In this case it is possible to prove that the estimation of the state value function at step k can be approximated by a parameterized functional form with parameter vector θ_k, that we indicate by $\widehat{V}_k(s_t; \theta_k)$. Note that the parameter vector θ_k may vary step by step. In order to determine the optimal parameter vector, θ^*, which minimizes the "distance" between the unknown state

function $V^{\pi}(s_t)$ and its estimation $\widehat{V}^{\pi}(s_t;\theta_k)$, in most machine learning approaches the minimization of the mean square error is used, that is

$$\min_{\theta_k} \sum_{s} \left[V^{\pi}(s) - \widehat{V}^{\pi}(s;\theta_k) \right]^2.$$

The convergence of the parameter vector θ_k to the optimal one θ^* is proven for approximators characterized by functional forms like the affine ones. In particular, in order to build our FTSs we use the functional form

$$\widehat{V}^{\pi}(s;\theta) = \sum_{l=1}^{L} \theta_l \phi_l(s_l) = \theta' \phi(s),$$

where L indicates the number of considered state variables, and $\phi_l(\cdot)$ is the squashing function of the l-th state variable.

Finally, under mild assumptions it is possible to prove that the update rules to use for estimating the state-action value function and the parameter vector in the case of continuous state variables respectively are

$$d_k = r(s_t, a_t, s_{t+1}) + \gamma \max_{a} \widehat{Q}(s_{t+1}, a; \theta_k) - \widehat{Q}^{\pi}(s_t, a_t; \theta_k)$$

and

$$\theta_{k+1} = \theta_k + \alpha d_k \nabla_{\theta_k} \widehat{Q}^{\pi}(s_t, a_t; \theta_k).$$

4 Our QLa-based FTSs and their applications

In this Section we present our QLa-based FTSs, and we provide the results of their applications to an artificial time series of daily stock prices and to six real ones from the Italian stock market belonging to the FTSE MIB basket.

4.1 Our QLa-based FTSs

In this sub-Section we use the QLa for designing and implementing differently configured FTSs.

In a first phase we have to identify the quantities which specify the state variables, the possible actions of the FTSs, and the reward functions. In a second phase we have to specify the technical aspects of the QLa.

With regard to the state variables, recalling that we are interested in checking the performance capability of our FTSs starting from basic information, we simply use the current logarithmic return and the past $N-1$ ones of the asset to trade. So, the state of the system at the time t is described by the vector

$$s_t = (e_{t-N+1},\, e_{t-N+2},\, \ldots,\, e_t),$$

where $e_\tau = \ln\left(p_\tau \big/ p_{\tau-1}\right)$, in which p_τ indicates the price of the asset at time τ. In particular, as we wish to develop FTSs that react enough quickly to new "information", in our application we use only the last $N = 1$ stock market day, and the last $N = 5$ stock market days (a stock market week).

Concerning the possible action of the FTSs, as introduced in Section 2 we utilize the three which follow:

$$a_t = \begin{cases} -1 \text{ (sell or stay-short-in-the-market signal)} \\ \;\;0 \text{ (stay-out-from-the-market signal)} \\ \;\;1 \text{ (buy or stay-long-in-the-market signal)} \end{cases},$$

in which the stay-out-from-the-market implies the closing of whatever previously open position (if any). We recall that in most of the prominent literature only the sell and the buy signals are considered.

With reference to the reward functions $r(s_t, a_t, s_{t+1})$ we take into account:

– The known Sharpe ratio, that is

$$SR_t = \frac{\mathbb{E}_L[g_{t-1}]}{\sqrt{\mathbb{V}ar_L[g_{t-1}]}} \in \mathbb{R},$$

where SR_t indicates the Sharpe ratio at time t, $\mathbb{E}_L(\cdot)$ and $\mathbb{V}ar_L(\cdot)$ indicate respectively the sample mean operator and the sample variance one over the last L stock market days, and $g_t = a_{t-1}e_t - \delta\,|a_t - a_{t-1}|$ indicate the net-of-transaction-costs logarithmic return obtained at time t as a consequence of the action undertaken by

the FTS at time $t-1$, in which $\delta > 0$ indicates the transaction costs in terms of percentage (for simplicity's sake, in the following of the paper we use the only term "net" for the expression "net-of-transaction-costs"). Note that the such costs affect the net logarithmic return only when two consecutive actions are different between them;

– The average logarithmic return, that is

$$ALR_t = \mathbb{E}_L(g_{t-1}) \in [-100, +\infty),$$

where ALR_t indicates the average logarithmic return at time t calculated over the last L stock market days;

– The ratio between the sum of the net logarithmic returns and the sum of the absolute value of the same net logarithmic returns, that is

$$OVER_t = \frac{\sum_{i=0}^{L-1} g_{t-i}}{\sum_{i=0}^{L-1} |g_{t-i}|} \in [-100\%,\ 100\%],$$

where $OVER_t$ indicates the ratio between the sum of the net logarithmic returns and the sum of the absolute value of the same net logarithmic returns at time t both calculated over the last L stock market days. Qualitatively, $OVER_t$ can be interpreted as the percentage of the net logarithmic returns which are positive during the considered time period with respect to all the possible ones.

In particular, as we wish functions of reward that react enough quickly to the consequences of the actions of the considered FTSs, we calculated SR_t, ALR_t and $OVER_t$ by using only the last $L = 5$ stock market days, and the last $L = 21$ stock market days (a stock market month).

Finally, as we are interested in checking the applicability of the QLa to the implementation of effective FTSs, we set δ equal to the realistic percentage $0.15‰$ per transaction.

Now we pass to specify the technical aspects of the QLa:

– The linear approximator of the state-action value function we choose is

$$Q(s_t, a_t; \theta_k) \approx \theta_{k,0} + \sum_{l=1}^{L} \theta_{k,l} \tanh(s_{t,l}),$$

where $\tanh(\cdot)$ plays the role of squashing function of the state variables;

– The improvement criterion we consider is:

$$a_t = \begin{cases} \arg\max_{a_t} Q(s_t, a_t; \theta_k) & \text{with probability } 1-\varepsilon \\ u & \text{with probability } \varepsilon \end{cases}, \tag{7}$$

where $\varepsilon \in \{5.0\%, 10.0\%, 25.0\%, 50.0\%\}$, and $u \sim \mathscr{U}_d\{-1, 0, 1\}$. Note that the values for ε are both close to those suggested by the prominent literature (5.0% and 10.0%) and not-so-used (25.0% and 50.0%).

Summarizing, for developing our FTSs we consider two different values of N (1 and 5), three different functions of reward (SR_t, ALR_t and $OVER_t$), two different values of L (5 and 22), and four different values of ε (5.0%, 10.0%, 25.0% and 50%), for a total of forty-eight different configurations.

4.2 Applications and results

In this sub-Section we apply the so specified QLa-based FTSs to an artificial time series of daily prices and six real ones from the Italian stock market belonging to the FTSE MIB basket. For comparability purpose, all the considered time series have the same length, and the real ones refer to the same time period.

With regard to the artificial time series, we generate a price series as a random walk with autoregressive trend processes, like proposed in [16]. To this end we use the model

$$p_t = \exp\left\{\frac{z_t}{\max z_t - \min z_t}\right\},$$

where $z_t = z_{t-1} + \beta_{t-1} + 3a_t$, in which $\beta_t = 0.9\beta_{t-1} + b_t$, $a_t \sim \mathscr{N}(0,1)$, and $b_t \sim \mathscr{N}(0,1)$. The length of the so-generated series is $T = 7{,}438$. It shows features which are usually present in real financial price series. In particular, it is trending on short time scales and has a not constant level of volatility. As far as the real price series concerns, we utilize the closing prices of Fiat S.p.A., Pirelli & C. S.p.A., Saipem S.p.A., Telecom Italia S.p.A., UniCredit S.p.A., and UnipolSai Assicurazioni S.p.A. from January 2, 1985 to May 30, 2014 (almost 30 stock market years).

At this point we can start to present the results of the applications of the variously configured FTSs. In all the applications we set the learning rate $\alpha = 5\%$ and the discount $\gamma = 0.95\%$. These values are generally those suggested by the prominent literature.

Preliminary, we have to discuss an aspect which is related to the random initialization of the vector of the parameters θ_k. In figure 1 we graphically report the results of the application of the QLa-based FTS to the real price series of Pirelli & C. S.p.A., with $N = 1$, the Sharpe ratio as reward function, $L = 5$ and $\varepsilon = 10.0\%$. In particular: The first panel shows the price series; The second panel shows the actions taken by the FTS at each time instant; The third panel shows the reward function at each time instant; The fourth panel shows the net equity line one should obtain by investing at time $t = 0$ an initial capital C_0 equal to 100. At the end of the trading period, $t = T$, the net final capital is $C_T = 240.70\%$.

We recall that at the beginning of the trading period, $t = 0$, the vector of the parameters used in the linear approximator is randomly initialized. Because of it, by repeating more times the same application we observe a certain variability in the net final capital. With reference to the same price series considered in figure 1, some of the other net final capitals we obtained are: $C_T = 50.66$, $C_T = 101.48$ and $C_T = 128.16$. This shows that the influence of the random initialization of the vector parameters spreads during the trading period. So, in order to check and to manage the effects of this random initialization, for each of the investigated asset we repeat 250 times the application of each of the considered configurations. In tables 1 to 7 we report some statistics concerning the results. In particular, with respect to the 250 iterations of each application: The column labeled $\overline{g}$ shows the net average yearly logarithmic return obtained during the trading period; The column labeled % shows the percentage of times in which the net average capital $\overline{C_t}$, with $t = N+1, \ldots, T$, is greater than the initial capital C_0 (we recall that the first N logarithmic returns can be used only as state variables); The column labeled $\overline{\#}$ shows the average number of transactions per stock market year.

The main stylized facts detectable from tables 1 to 7 are the following ones:

– In general terms, the performances of our FTSs are enough satisfactory. In fact, jointly considering all the investigated assets, there emerges that: The percentages

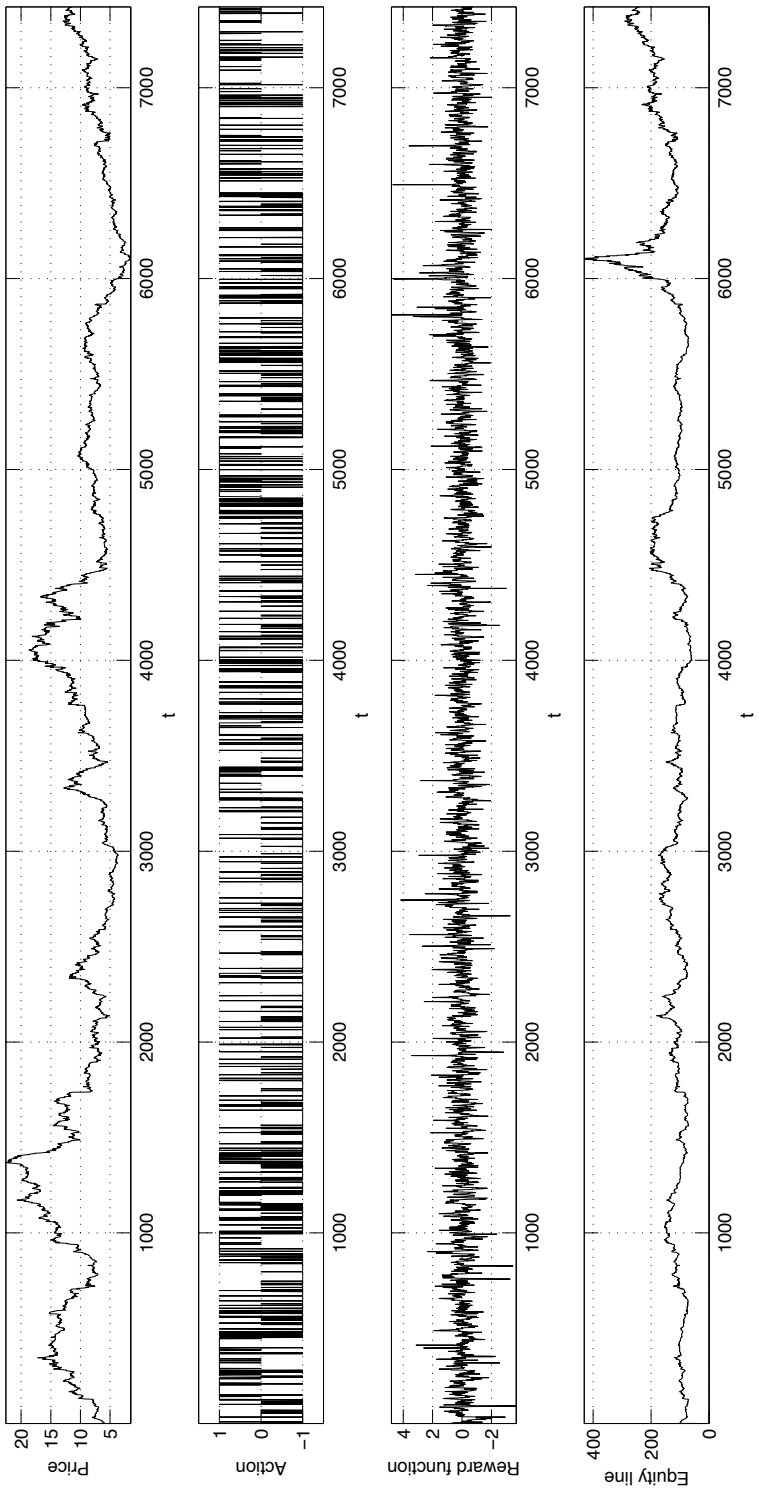

Fig. 1. Results of the QLa-based FTS applied to the Pirelli & C. S.p.A. price series with $N = 1$, the Sharpe ratio as reward function, $L = 5$ and $\varepsilon = 10.0\%$. Net final capital: $C_T = 240.70\%$.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	4.33%	99.87%	35.16	4.38%	98.86%	35.47
SR_t	5	10.0%	4.80%	99.76%	59.88	4.74%	99.56%	60.00
SR_t	5	25.0%	4.38%	99.73%	119.71	4.41%	99.29%	119.77
SR_t	5	50.0%	2.27%	95.43%	185.40	2.15%	92.45%	185.80
SR_t	21	5.0%	1.57%	95.74%	51.92	1.62%	96.65%	52.05
SR_t	21	10.0%	1.47%	97.30%	75.70	1.50%	95.88%	75.84
SR_t	21	25.5%	1.17%	93.13%	127.95	1.14%	96.17%	127.82
SR_t	21	50.0%	−0.05%	3.12%	185.68	0.14%	16.31%	185.38
ALR_t	5	5.0%	2.29%	87.19%	38.35	2.11%	92.22%	33.78
ALR_t	5	10.0%	2.36%	96.20%	54.49	1.85%	91.63%	51.72
ALR_t	5	25.0%	2.30%	90.10%	109.64	1.34%	76.11%	106.37
ALR_t	5	50.0%	0.77%	44.99%	176.16	−0.21%	2.13%	173.54
ALR_t	21	5.0%	0.73%	86.39%	43.16	0.33%	70.48%	35.80
ALR_t	21	10.0%	0.68%	64.44%	58.91	0.11%	55.31%	53.29
ALR_t	21	25.0%	0.25%	30.12%	111.20	−0.47%	0.32%	106.72
ALR_t	21	50.0%	−0.97%	0.00%	176.04	−1.68%	0.00%	172.54
$OVER_t$	5	5.0%	4.11%	97.81%	38.27	4.10%	98.26%	38.39
$OVER_t$	5	10.0%	4.56%	99.50%	63.41	4.57%	98.82%	63.01
$OVER_t$	5	25.0%	4.36%	99.41%	123.15	4.40%	99.56%	122.66
$OVER_t$	5	50.0%	2.43%	94.18%	187.29	2.39%	93.42%	187.54
$OVER_t$	21	5.0%	1.55%	97.07%	52.95	1.32%	95.49%	53.55
$OVER_t$	21	10.0%	1.45%	96.45%	77.73	1.29%	94.89%	78.43
$OVER_t$	21	25.0%	1.22%	96.28%	129.78	1.21%	95.56%	129.90
$OVER_t$	21	50.0%	0.06%	12.94%	186.87	0.01%	4.72%	186.48

Table 1. Artificial daily stock price series: Some statistics.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	−2.51%	67.85%	35.01	−2.99%	66.37%	35.02
SR_t	5	10.0%	−2.83%	71.88%	60.87	−2.40%	70.18%	60.77
SR_t	5	25.0%	1.19%	99.69%	120.62	−0.37%	87.69%	120.69
SR_t	5	50.0%	1.07%	98.32%	185.40	−0.74%	60.35%	184.60
SR_t	21	5.0%	−1.63%	69.71%	47.77	−1.64%	72.81%	46.55
SR_t	21	10.0%	−0.04%	89.15%	71.49	−0.66%	80.21%	70.75
SR_t	21	25.5%	1.19%	99.93%	123.55	0.39%	99.93%	122.99
SR_t	21	50.0%	1.82%	99.91%	183.33	−0.56%	88.89%	182.77
ALR_t	5	5.0%	−0.01%	91.24%	31.14	0.29%	93.22%	28.46
ALR_t	5	10.0%	1.46%	99.97%	51.26	0.98%	99.96%	49.06
ALR_t	5	25.0%	0.84%	96.40%	106.49	1.88%	84.07%	104.33
ALR_t	5	50.0%	−1.25%	41.54%	173.83	−1.77%	26.75%	172.09
ALR_t	21	5.0%	−0.31%	83.79%	34.06	−0.25%	85.60%	29.85
ALR_t	21	10.0%	0.35%	96.85%	53.56	0.41%	94.20%	49.60
ALR_t	21	25.0%	0.15%	68.05%	108.06	0.41%	99.74%	104.52
ALR_t	21	50.0%	−1.70%	26.29%	174.41	−2.49%	35.31%	172.20
$OVER_t$	5	5.0%	−2.15%	69.44%	35.70	−2.63%	69.45%	36.36
$OVER_t$	5	10.0%	−0.68%	83.56%	61.93	−2.22%	76.61%	62.52
$OVER_t$	5	25.0%	1.90%	99.96%	122.05	1.66%	92.77%	122.68
$OVER_t$	5	50.0%	−0.35%	88.27%	186.63	0.77%	90.40%	186.57
$OVER_t$	21	5.0%	−1.71%	71.45%	47.78	−0.94%	77.92%	46.34
$OVER_t$	21	10.0%	0.21%	95.80%	71.82	−0.10%	88.58%	71.75
$OVER_t$	21	25.0%	3.33%	99.96%	125.07	0.95%	99.66%	125.14
$OVER_t$	21	50.0%	0.36%	98.56%	184.31	0.07%	87.54%	184.15

Table 2. Fiat S.p.A. daily stock price series: Some statistics.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	1.91%	97.95%	36.72	3.08%	99.99%	36.40
SR_t	5	10.0%	3.43%	99.96%	61.92	3.68%	99.99%	61.79
SR_t	5	25.0%	3.81%	99.85%	121.37	3.85%	99.85%	121.38
SR_t	5	50.0%	0.99%	97.66%	184.88	1.14%	99.15%	184.81
SR_t	21	5.0%	−0.97%	42.37%	48.22	−1.54%	39.78%	48.60
SR_t	21	10.0%	0.77%	74.87%	71.37	0.62%	66.50%	70.92
SR_t	21	25.5%	0.73%	96.28%	123.90	1.30%	99.92%	123.13
SR_t	21	50.0%	−0.54%	44.89%	183.16	−0.81%	61.19%	182.89
ALR_t	5	5.0%	−0.34%	53.72%	30.56	−0.73%	59.83%	28.76
ALR_t	5	10.0%	0.35%	97.83%	50.90	−0.08%	96.68%	49.23
ALR_t	5	25.0%	−0.45%	81.37%	107.19	−0.24%	48.30%	104.57
ALR_t	5	50.0%	−1.22%	43.63%	173.72	−2.17%	3.30%	172.52
ALR_t	21	5.0%	−1.53%	32.19%	33.18	−1.33%	28.05%	28.17
ALR_t	21	10.0%	−0.25%	41.78%	53.16	−1.26%	29.95%	49.35
ALR_t	21	25.0%	−0.96%	48.81%	108.48	−0.27%	77.19%	104.07
ALR_t	21	50.0%	−1.63%	6.16%	174.62	−2.64%	0.00%	171.86
$OVER_t$	5	5.0%	2.22%	99.78%	38.26	1.44%	93.23%	38.54
$OVER_t$	5	10.0%	4.30%	99.77%	62.61	4.94%	99.99%	62.19
$OVER_t$	5	25.0%	4.63%	99.03%	122.77	4.34%	99.99%	122.52
$OVER_t$	5	50.0%	1.72%	99.96%	186.96	1.91%	99.85%	186.74
$OVER_t$	21	5.0%	−1.84%	36.37%	49.05	−1.87%	39.92%	48.67
$OVER_t$	21	10.0%	0.55%	67.38%	72.34	0.32%	62.13%	72.19
$OVER_t$	21	25.0%	2.30%	99.78%	125.31	1.45%	99.70%	125.14
$OVER_t$	21	50.0%	−0.08%	33.91%	184.41	−0.61%	47.82%	183.77

Table 3. Pirelli & C. S.p.A. daily stock price series: Some statistics.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	−6.37%	29.79%	36.34	−6.56%	26.99%	36.20
SR_t	5	10.0%	−4.97%	44.71%	60.58	−4.84%	47.20%	60.42
SR_t	5	25.0%	−2.07%	52.64%	120.22	−2.06%	64.10%	119.87
SR_t	5	50.0%	−2.57%	46.78%	185.25	−2.58%	52.14%	184.64
SR_t	21	5.0%	−1.50%	53.82%	45.31	−1.53%	52.66%	44.47
SR_t	21	10.0%	0.01%	78.97%	69.32	1.44%	82.53%	68.68
SR_t	21	25.5%	0.46%	89.81%	122.47	1.42%	99.99%	122.63
SR_t	21	50.0%	−1.26%	23.40%	183.03	−1.37%	26.42%	182.91
ALR_t	5	5.0%	−0.47%	60.85%	30.65	−0.58%	52.25%	27.62
ALR_t	5	10.0%	0.40%	56.54%	51.24	−0.17%	59.15%	49.11
ALR_t	5	25.0%	−0.44%	30.23%	106.27	0.27%	66.47%	104.56
ALR_t	5	50.0%	−1.40%	15.66%	174.00	−2.15%	0.15%	172.26
ALR_t	21	5.0%	0.17%	63.10%	32.61	−1.11%	31.60%	28.81
ALR_t	21	10.0%	−0.11%	52.51%	53.46	−0.23%	57.56%	49.56
ALR_t	21	25.0%	−0.08%	16.09%	108.31	−1.12%	1.85%	104.40
ALR_t	21	50.0%	−0.23%	9.62%	174.56	−2.66%	0.40%	171.93
$OVER_t$	5	5.0%	−5.17%	38.41%	36.54	−5.09%	41.58%	36.88
$OVER_t$	5	10.0%	−3.63%	56.28%	60.85	−3.18%	57.27%	61.24
$OVER_t$	5	25.0%	−0.44%	88.31%	121.50	−0.53%	79.16%	121.20
$OVER_t$	5	50.0%	−1.17%	47.98%	186.73	−1.57%	58.63%	187.02
$OVER_t$	21	5.0%	−0.61%	62.77%	44.87	−1.14%	57.78%	44.51
$OVER_t$	21	10.0%	0.01%	69.86%	69.70	−0.39%	69.14%	69.33
$OVER_t$	21	25.0%	0.78%	89.56%	124.57	0.95%	99.99%	123.86
$OVER_t$	21	50.0%	−0.79%	55.52%	183.94	0.04%	66.11%	183.53

Table 4. Saipem S.p.A. daily stock price series: Some statistics.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	6.38%	97.94%	35.20	7.13%	96.82%	34.89
SR_t	5	10.0%	6.98%	97.01%	60.22	7.12%	98.05%	60.18
SR_t	5	25.0%	5.31%	96.92%	120.16	5.43%	97.59%	120.15
SR_t	5	50.0%	1.75%	96.94%	184.72	1.24%	96.79%	184.75
SR_t	21	5.0%	2.56%	94.09%	46.26	3.81%	98.40%	46.16
SR_t	21	10.0%	4.08%	97.78%	69.66	4.88%	95.08%	69.90
SR_t	21	25.5%	3.65%	97.08%	124.02	3.58%	98.14%	123.23
SR_t	21	50.0%	0.01%	97.28%	182.85	0.41%	97.52%	182.77
ALR_t	5	5.0%	3.47%	97.49%	30.80	3.73%	87.72%	28.38
ALR_t	5	10.0%	4.06%	90.85%	50.66	2.13%	87.84%	48.73
ALR_t	5	25.0%	2.71%	89.52%	107.00	2.44%	89.35%	104.17
ALR_t	5	50.0%	−0.37%	89.90%	173.84	−1.91%	31.52%	172.48
ALR_t	21	5.0%	1.36%	82.61%	33.67	0.92%	79.81%	28.68
ALR_t	21	10.0%	1.31%	88.98%	53.57	1.57%	90.52%	49.39
ALR_t	21	25.0%	0.94%	82.81%	108.29	0.58%	88.53%	104.08
ALR_t	21	50.0%	1.73%	95.39%	174.30	−1.17%	3.46%	171.97
$OVER_t$	5	5.0%	6.40%	98.33%	36.10	6.04%	98.46%	36.26
$OVER_t$	5	10.0%	7.42%	98.25%	61.72	7.90%	98.49%	60.84
$OVER_t$	5	25.0%	5.64%	97.12%	122.70	7.80%	97.91%	121.87
$OVER_t$	5	50.0%	0.95%	96.78%	186.42	2.74%	98.05%	186.31
$OVER_t$	21	5.0%	3.79%	97.29%	46.43	4.27%	97.36%	45.93
$OVER_t$	21	10.0%	4.37%	97.01%	71.37	5.83%	98.34%	71.01
$OVER_t$	21	25.0%	3.54%	97.45%	125.48	4.48%	97.78%	124.71
$OVER_t$	21	50.0%	0.40%	96.51%	184.26	−0.11%	93.73%	183.82

Table 5. Telecom Italia S.p.A. daily stock price series: Some statistics.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	−2.36%	25.24%	37.01	−2.59%	25.25%	36.62
SR_t	5	10.0%	1.37%	41.42%	60.53	−0.35%	32.68%	60.67
SR_t	5	25.0%	1.23%	67.77%	120.25	1.25%	58.72%	120.34
SR_t	5	50.0%	−0.50%	35.96%	185.03	0.29%	51.79%	184.71
SR_t	21	5.0%	−1.27%	30.69%	49.39	−1.08%	31.28%	48.33
SR_t	21	10.0%	0.08%	37.63%	72.39	−0.72%	31.56%	71.70
SR_t	21	25.5%	1.03%	72.81%	123.23	0.53%	58.37%	123.19
SR_t	21	50.0%	0.18%	46.58%	182.85	−0.03%	54.31%	182.94
ALR_t	5	5.0%	0.29%	66.98%	30.78	0.78%	51.47%	27.76
ALR_t	5	10.0%	−0.92%	21.17%	50.87	0.00%	48.88%	48.48
ALR_t	5	25.0%	−0.92%	22.47%	106.60	0.48%	25.84%	103.84
ALR_t	5	50.0%	−1.51%	6.29%	173.63	−1.70%	10.96%	172.17
ALR_t	21	5.0%	0.53%	56.36%	34.13	0.05%	40.66%	29.08
ALR_t	21	10.0%	0.22%	71.72%	53.86	0.57%	52.36%	49.03
ALR_t	21	25.0%	−0.81%	39.18%	108.38	−0.25%	36.78%	104.06
ALR_t	21	50.0%	−0.44%	14.96%	174.19	−3.05%	0.77%	171.89
$OVER_t$	5	5.0%	−3.31%	26.83%	38.40	−3.09%	23.94%	38.55
$OVER_t$	5	10.0%	−0.49%	34.29%	62.77	−0.84%	30.80%	62.75
$OVER_t$	5	25.0%	1.59%	60.39%	122.32	1.41%	54.22%	122.46
$OVER_t$	5	50.0%	−0.74%	46.01%	186.68	0.71%	44.70%	186.47
$OVER_t$	21	5.0%	−1.34%	33.68%	49.58	−0.84%	37.49%	48.14
$OVER_t$	21	10.0%	0.26%	48.69%	73.11	−0.57%	39.42%	72.84
$OVER_t$	21	25.0%	1.34%	82.15%	125.42	1.04%	86.75%	125.27
$OVER_t$	21	50.0%	−1.09%	52.03%	184.41	0.01%	31.87%	184.68

Table 6. UniCredit S.p.A. S.p.A. daily stock price series: Some statistics.

			$N=1$			$N=5$		
Reward function	L	ε	$\overline{g}$	%	$\overline{\#}$	$\overline{g}$	%	$\overline{\#}$
SR_t	5	5.0%	1.21%	69.13%	34.81	0.52%	67.61%	34.84
SR_t	5	10.0%	1.25%	63.26%	59.79	0.93%	51.50%	59.52
SR_t	5	25.0%	1.50%	68.49%	120.58	2.55%	72.00%	120.18
SR_t	5	50.0%	2.14%	60.83%	184.51	2.40%	43.56%	184.85
SR_t	21	5.0%	−0.73%	39.05%	47.51	−1.76%	34.54%	46.87
SR_t	21	10.0%	−0.83%	35.73%	71.07	−0.34%	37.67%	70.81
SR_t	21	25.5%	2.23%	77.73%	123.63	3.50%	57.62%	122.49
SR_t	21	50.0%	−0.63%	44.34%	182.87	−1.00%	39.16%	182.64
ALR_t	5	5.0%	−0.68%	76.50%	30.39	−2.43%	48.34%	27.84
ALR_t	5	10.0%	−0.93%	46.43%	50.84	−2.77%	53.98%	48.58
ALR_t	5	25.0%	−2.38%	35.75%	107.38	−2.69%	35.16%	103.85
ALR_t	5	50.0%	−2.20%	41.40%	173.82	−2.60%	19.53%	172.47
ALR_t	21	5.0%	−1.23%	59.63%	34.05	−2.31%	43.52%	28.63
ALR_t	21	10.0%	−1.34%	55.52%	53.41	−0.65%	52.97%	49.34
ALR_t	21	25.0%	−4.30%	30.96%	108.48	−3.33%	32.41%	104.45
ALR_t	21	50.0%	−0.81%	23.24%	174.07	−3.47%	30.44%	171.82
$OVER_t$	5	5.0%	0.22%	56.12%	35.41	−0.01%	61.03%	35.50
$OVER_t$	5	10.0%	2.97%	57.56%	60.85	−0.54%	43.00%	61.36
$OVER_t$	5	25.0%	1.63%	70.26%	122.05	2.42%	80.13%	121.93
$OVER_t$	5	50.0%	2.38%	98.07%	186.84	0.20%	50.04%	186.83
$OVER_t$	21	5.0%	−0.46%	40.69%	48.15	−0.96%	42.37%	46.86
$OVER_t$	21	10.0%	0.61%	45.93%	72.32	−0.97%	35.86%	72.27
$OVER_t$	21	25.0%	2.11%	64.93%	125.58	1.86%	67.30%	124.65
$OVER_t$	21	50.0%	0.65%	71.45%	184.03	3.04%	79.25%	183.83

Table 7. UnipolSai Assicurazioni S.p.A. daily stock price series: Some statistics.

of checked configurations for which $\overline{g} > 0$, that is for which the average yearly logarithmic return obtained during the trading period is positive, is equal to 55.95%; The percentages of checked configurations for which $\overline{\#} > 50\%$, that is for which the net average capital $\overline{C_t}$, with $t = N+1, \ldots, T$, is greater than the initial capital C_0 at least for the fifty percent of the trading days, is equal to 68.45%. Moreover, considering only the real assets, the values of these percentages result, respectively, slightly changed to 50.35% and to 66.57%. Note that the latter percentage is more informative about the quality of the performances of our FTSs than the former one. Indeed, the former percentage is based only on the two capitals C_0 and $\overline{C_T}$, whereas the latter one takes into account all the capitals $\overline{C_t}$, with $t = N+1, \ldots, T$. The observations reported in the following points are all based on such percentages;

- Concerning the number of the state variables to consider, N, it appears that our FTSs are generally better performing with $N = 1$. Following the AMH, it indicates that the profit opportunities which are present in the "piece" of financial market constituted by the investigated assets are quickly exploited by the agents acting in that same "piece" of financial market;
- With regard to the reward function to take into account, the performances coming from the use of SR_t and $OVER_t$ result more or less equivalent, whereas the performances coming from the use of ALR_t are generally worse than both the previous ones. Once again in the light of the AMH, it shows that simple reward function are not sufficient "to capture" the complexity of real financial markets. Furthermore, with respect to the number of the (last) trading days over which the various reward functions are calculated, L, in the case of the artificial price series there is a strong evidence in favor of $N = 5$, whereas in the case of the real price series this evidence becomes very light;
- Finally, concerning the value of ε to set in the improvement criterion (7), it appears that generally the best performances of out FTSs are achieved in correspondence of $\varepsilon = 10.0\%$ and $\varepsilon = 25.0\%$.

5 Some concluding remarks

In this paper, first we have designed, developed and applied some original FTSs based on differently configured QLas, then we have presented the generally satisfactory results coming out from the applications of these FTSs to to an artificial time series of daily stock prices and to six real ones from the Italian stock market belonging to the FTSE MIB basket. Of course, many questions remain to be explored. Among the main ones:

- The choice of the logarithmic returns as state variables has been deliberately simple. Once checked the capability of our FTSs to perform well also starting from basic information, currently we are beginning to work to specify more refined state variables (in the first experimentations they have provided interesting results);
- SR_t and $OVER_t$ have performed more than decently as reward functions. Nevertheless, both suffer serious financial limits which make them incapable to appro-

priately measure the performances of advanced FTSs (like ours) when applied to the complexity of real financial markets. Therefore, currently we are beginning to consider new and not standard reward functions;

- Generally speaking, modern financial markets are at least not stationary, and are also often characterized by structurally time-varying dynamics. Both these features are important sources of information that a FTS should be capable to effectively exploit. To this end, it is possible to prove that using a constant learning rate does not assure the achievement of the optimal trading strategy (see for instance [1]). To this end, we are beginning to work to develop some approach for a dynamic management of the learning rate;
- The results we have provided in the previous Section are "on average", in the sense that each of them is based on 250 applications of the same given configurations. Of course, such an approach requires 250 initial capitals. But a real operating trading system has available only 1 initial capital. So, an important question to investigate in a future research is: How to detect since the first time instants the optimal (or good sub-optimal) trading strategy among all the iterated ones?
- Finally, in order to deepen the assessment about the capabilities of our FTSs, we have to apply them to more and more price series coming from different financial markets.

Acknowledgements

The Authors wish to thank the Department of Economics of the Ca' Foscari University of Venice for the support received within the research project *Gestione neuro-dinamica di capitali di rischio* [*Neuro-dynamic management of risky capitals*].

References

1. Barto A.G., Sutton R.S. (1998), *Reinforcement Learning: An Introduction*. *Adaptive Computation and Machine Learning*. The MIT Press.

2. Bekiros S.D. (2010), Heterogeneous trading strategies with adaptive fuzzy Actor-Critic reinforcement learning: A behavioral approach, *Journal of Economic Dynamics & Control*, 34 (6), 1153-1170.
3. Bertoluzzo F., Corazza M. (2007) Making financial trading by recurrent reinforcement learning. In: Apolloni B, Howlett R.J., Jain L. (Eds.) *Knowledge-Based Intelligent Information and Engineering Systems*, 619-626 [Lecture Notes in Artificial Intelligence, vol. 4693]. Springer.
4. Bertsekas D.P., Tsitsiklis J.N. (1996), *Neuro-Dynamic Programming*. Athena Scientific.
5. Brent R.P. (1973), *Algorithms for Minimization without Derivatives*. Prentice-Hall.
6. Casqueiro P.X., Rodrigues A.J.L. (2006), Neuro-dynamic trading methods, *European Journal of Operation Research*, 175 (3), 1400-1412.
7. Cuthbertson K., Nitzsche D. (2004), *Quantitative Financial Economics*. Wiley.
8. Farmer D., Lo A.W. (2002), Market force, ecology and evolution, *Industrial and Corporate Change*, 11 (5), 895-953.
9. Gold C. (2003), FX trading via recurrent Reinforcement Learning, *Proceedings of the IEEE International Conference on Computational Intelligence in Financial Engineering*, 363-370.
10. Gosavi, A. (2003), *Simulation-based Optimization. Parametric Optimization Techniques and Reinforcement Learning*. Kluwer Academic Publishers.
11. Li H., Dagli C.H., Enke D. (2007) Short-term stock market timing prediction under reinforcement learning schemes, *Proceedings of the 2007 IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning*, 233-240.
12. Kearns M., Nevmyvaka Y. (2013) Machine learning for market microstructure and high frequency trading. In: Easley D., López de Prado M., O'Hara M. (Eds.) *High-Frequency Trading – New Realities for Traders, Markets and Regulators*, 91-124. Risk books.
13. Lo A.W. (2004), The Adaptive Markets Hypothesis. Market efficiency from an evolutionary perspective, *The Journal of Portfolio Management*, 30 (5), 15-29.
14. Lo A.W. (2005), Reconciling efficient markets with behavioral finance: The Adaptive Markets Hypothesis, *The Journal of Investment Consulting*, 7 (2), 21-44.
15. Moody J., Wu L., Liao Y., Saffel M. (1998), Performance functions and Reinforcement Learning for trading systems and portfolios, *Journal of Forecasting*, 17 (56), 441-470.
16. Moody J., Saffel M. (2001), Learning to trade via Direct Reinforcement, *IEEE Transactions on Neural Network*, 12, 875-889.

17. O J., Lee J., Lee J.W., Zhang B.-T. (2006) Adaptive stock trading with dynamic asset allocation using reinforcemnt learning, *Information Sciences*, 176 (15), 2121-2147.
18. Smart W.D., Kaelbling L.P. (2000), Practical Reinforcement Learning in continuous spaces, *Proceedings of the 17th International Conference on Machine Learning*, 903-910.
19. Tan Z., Quek C., Cheng P.Y.K (2011), Stock trading with cycles: A financial application of ANFIS and Reinforcement Learning, *Expert Systems with Applications*, 38 (5), 4741-4755.