From data driven decision making (DDDM) to automated data driven model based decision making (MBDM)

Ioan Doré Landau, Vlad Landau

From data driven decision making (DDDM) to automated data driven model based decision making (MBDM)

I.D. Landau *[1] et V. Landau[2]

[1]GIPSA-Lab, Université Grenoble Alpes, UMR 5216 CNRS
[2]Free Lance Consultant

11 avril 2017

Résumé

Data driven decision making (DDDM) constitutes the modern operation of systems and organizations. However a prediction model of the system can be estimated from data. The availability of data driven prediction models allows on one hand to test in simulation various decision policies and on the other hand to move toward an automated decision making procedure. These automated decision making procedures have a feedback structure and the design of the decision algorithms relies on the knowledge of the prediction model of the system. This procedure can be called *automated data driven model based decision making* in short (MBDM). The paper explores basic aspects to be considered in the design of the decision algorithms related to optimal and safe operation of the full system.

Key-Words :Data mining, Prediction model, Parameters estimation, Feedback control, Stability, Predictive control

1 Introduction

Acquired data are fundamental for taking realistic decisions. Basically one can say that decision will be made by an *agent* who uses either his professional experience or an intuitive model (or even a simple theoretical model) in order to take a decision which will be effectively implemented through an *actuator* (called also *effector*)[FT13]. The principle of DDDM (Data Driven Decision Making) is illustrated in Fig. 1. However the Data Mining methodology provides an opportunity for estimating a prediction model of the system (organization, business, environment, etc..) from the acquired data ([FT13], [LP15]). These data driven prediction models replace the intuitive or theoretical models used for taking decisions.

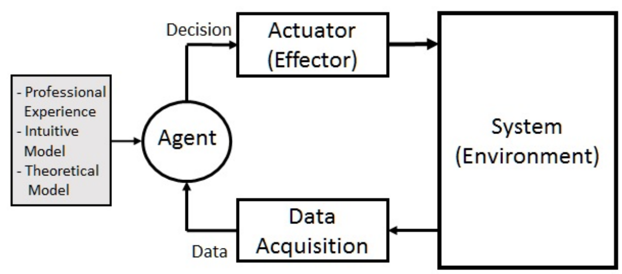

FIGURE 1 – Principle of DDDM (Data Driven Decision Making).

In fact the availability of these *prediction models* allows to move toward an automated decision procedure as it will be shown subsequently. Two basic configurations can be considered once a prediction model is available.

1. Off line simulation of the decision process and its effects using the prediction model.
2. Automated real time decision operation designed on the basis of the prediction model.

In the off line operation, the decision process is run on the prediction model. This is a simulation (to be run at high speed) which provides an image of the evolution of the real process and will assist the *agent* in selecting an adequate decision. The use of this approach is based on the assumptions that :

— the prediction model used is relevant for the operation of the system ;
— the prediction model of the real system does not evolve in time.

*ioan-dore.landau@gipsa-lab.grenoble-inp.fr

In the second approach a decision algorithm is designed based on the knowledge of the prediction model and will automatically provide decisions to be applied to the system. The automated decision procedure will have a feedback structure and is illustrated in Fig 2. In the phase 1, estimation of the prediction model of the system is done [1] and the decision algorithms will be designed based on this estimated prediction model. In the phase 2, one operates the automated model based decision making system. In addition, the automated decision procedure should include two other important elements (not represented in Fig. 2)

1. monitoring (supervision) of the performance and safe operation of the system
2. updating in real time of the prediction model (adaptation)

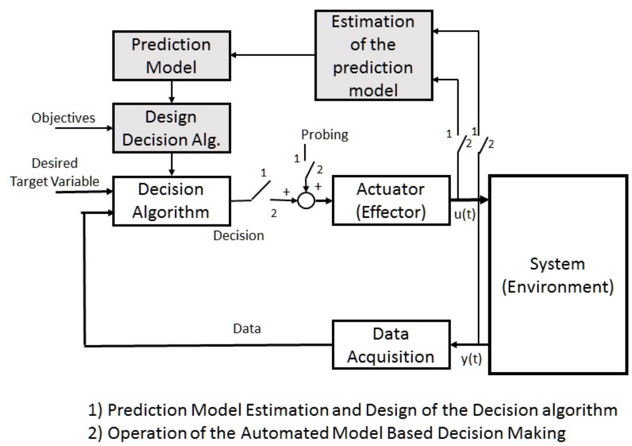

FIGURE 2 – Principle of automated MBDM (Data Driven Model Based Decision Making).

In an automated decision procedure the role of the *agent* is to supervise the operation of the system by looking to a number of performance indicators provided (these indicators can be interpreted as *agregated data*). Supervision indicators have to be provided by the system and the *agent* has to be able to interact with the system.

Since the automated decision system has a feedback structure, the dynamic properties of the environment (reflected also in the prediction model) should be taken into account for assuring a stable and optimal operation fo the full system.

The fundamental issues to be investigated when designing an automated decision making system are :

1. the dynamic features of the prediction model (i.e. of the environment)
2. the stable operation of the automated feedback model based decision making
3. the optimality of the feedback decision making process
4. the adaptation in real time of the decision algorithm in the presence of variation of the environment behavior.

In the case of non stationarity of the prediction model of the system (environment) one has to consider an *adaptive* solution i.e. real time updating of the estimated prediction model an redesign in real time of the decision algorithm. See [LLMK11]. The principle of adaptive automated MBDM is illustrated in Fig. 3.

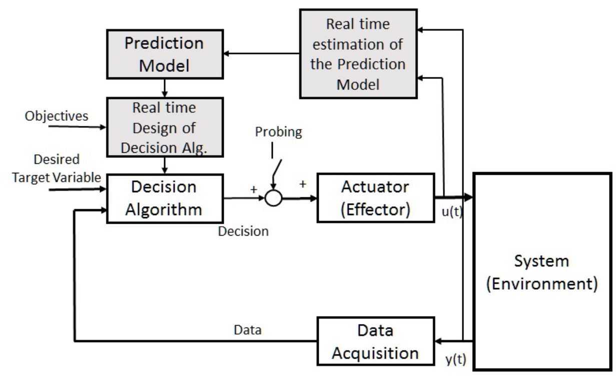

FIGURE 3 – Principle of adaptive automated MBDM (Data Driven Model Based Decision Making).

The present paper will focus on the three first issues. The real time adaptation issue will be the object of a separate contribution.

The paper is organized as follows : In Section 2 the basic dynamical features of the systems are considered. These features will be used in Section 3 for presenting the basic properties of feedback systems encountered in automated decision making. This section will focus on the stability of these systems. Section 4 will briefly present the short range and long range predictive decision strategies which use explicitly the predictive model in the decision algorithm. An example of a predictive decision strategy is illustrated in Section 5.

2 Dynamic features of the systems (prediction models)

To simplify the presentation one can say that in a system one encounter two main dynamic behaviors :

1. A probing excitation may be necessary in some cases for estimation of the prediction model (supervised learning)[LZ05]

1. Dead-time (time-lag)
2. Inertia

The *dead-time* can be defined as the time between the application of a decision (though the *actuator*) and the detection of a change in the evolution of the target variable.

The *inertia* of a system may be characterized by the time needed (after the eventual *dead-time*) in order that the *target variable* reach a steady state value (with a certain tolerance) when a step change in the control (predictive) variable is applied as the result of the decision process. Fig. 4 illustrates both dead-time and inertia.

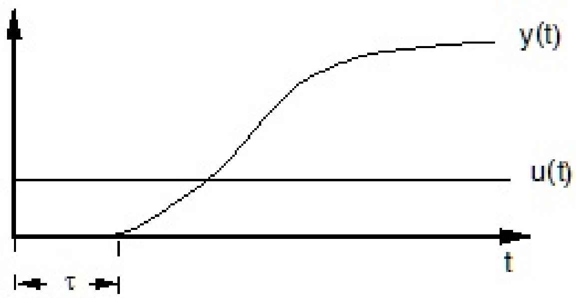

FIGURE 4 – Response of a system featuring a dead time (τ) and an inertia.

3 Feedback operation

As indicated previously, the automated decision making systems have a *feedback* structure. A key element is the *decision algorithm* (can be called also "controller") which will process the acquired information (data) and the desired values of the target variable and will generate the "decision". The *decision* will be effectively applied to the system (as indicated in Fig 2) by an actuator" or "effector" which will deliver the effective value of the predictive variable applied to the system (called also "control"). Once the structure of the automated decision system is finalized (what is the target variable ? what are the acquired data ? what are the desired values for the target variable ?), the key issue is the design of the "decision algorithm".

To design the "decision algorithm" one needs the knowledge fo the *prediction model* of the system to be controlled. This is called the "model based paradigm"([LZ05]).

For completeness, one should mention another approach for the design of the *decision algorithm* which is based on the use of the available knowledge concerning the various design decision done in the past for various configurations in order to design the "automated decision algorithm". This will be a rule based decision algorithm which try to exploit the available expertise (it is often called an "expert" system)[S.G01, AK86]. A detailed comparison of these two approaches is beyond the scope of this paper. In a total different field (automatic control of dynamical systems) the model based approach has shown its advantages with respect to the expert rule based design.

In the context of the "model based decision making" there are potentially many designs techniques for the "decision algorithm" but all require the knowledge of the prediction model of the system. One can say that directly or indirectly the decision algorithm incorporate the prediction model or some of its pertinent features.

Independently of the design method to be used in order to reach an optimal decision, there is one issue which all designed decision algorithm should assure and this is the "stability" of the system (since feedback system can become unstable !). More than this, one has to be sure that stability of the automated decision system is assured even in the presence of quantified variations of the dynamic properties of the system (and therefore of the prediction model) to be driven.

The stability issue is crucial in order to avoid the "Chernobyl effect". One also has to be sure that the "supervisor" (supervisory system) is able to restore the stability of the system even in the case of the occurrence of a beginning of an instability phenomenon.

We will examine the stability problem by considering a basic structure for the decision algorithm.

3.1 A basic decision making algorithm

The basic automated decision making algorithm should generate a value for a given predictive variable (control variable) in order to reduce the error between the desired value of the target variable and its current value. The algorithm should react to the instantaneous value of this error at a sampling instant but has also to take in account the previous values of the control variable which have been already applied to the system and whose effect are not fully reflected on the target variable at the sampling instant t because of the inertia of the system. Such a basic algorithm can have the following form :

$$u(t) = u_1(t) + u_2(t) \tag{1}$$

$$u_1(t) = K_P \epsilon(t) \tag{2}$$

$$u_2(t) = u_2(t-1) + K_I \epsilon(t) \tag{3}$$

where $\epsilon(t) = y^*(t) - y(t)$ is the error (difference) between the desired value of the target variable $y^*(t)$ and the measured value of the target value $y(t)$ at instant t. $u_1(t)$ is called the "proportional" action and $u_2(t)$ is called the "integral" action. Using the *delay operator* $q^{-1}(u(t-1) = q^{-1}u(t))$ ([LZ05], $u(t)$ can be expressed as :

$$u(t) = K_P \epsilon(t) + \frac{K_I \epsilon(t)}{1 - q^{-1}} \tag{4}$$

$$= \frac{r_0 + r_1 q^{-1}}{1 - q^{-1}} \epsilon(t) \tag{5}$$

where $r_0 = K_P + K_I$ and $r_1 = -K_P$. The corresponding decision algorithm is [2] :

$$u(t) = u(t-1) + r_0 \epsilon(t) + r_1 \epsilon(t-1) \tag{6}$$

3.2 The stability issue

To illustrate the stability issue we will discuss the design of the decision algorithm given in Eq. 6 assuming that it operates on a system whose prediction model is described by a simple inertia plus a dead-time. Such a model is :

$$y(t+1) = -a_1 y(t) + b_1 u(t-d) \tag{7}$$

a_1 reflects the inertia of the system, d reflects the dead-time expressed in terms of sampling periods (all the systems operates in discrete time and the decisions are sent to the system through the actuator at a certain sampling frequency). There is always a dead-time of one sampling period since the effect of a decision send at instant t can at the earliest be seen at instant $t+1$. Fig. 5 gives the time response of the system in the absence of the dead-time for various values of a_1 while keeping the steady state gain of the system constant (the steady state gain of the system is $b_1/(1+a_1)$). We will consider as decision algorithm, the algorithm presented in the previous subsection, Eq. 6. For stability, one has to analyze the behavior of the complete system (the system together with the decision algorithm) around a steady state value of the target variable. The full system will be described by :

$$y(t+1) = -a_1 y(t) + b_1 u(t-d) \tag{8}$$

$$u(t) = \frac{r_0 + r_1 q^{-1}}{1 - q^{-1}} [y^*(t) - y(t)] \tag{9}$$

2. This algorithm is often termed "proportional + integral"

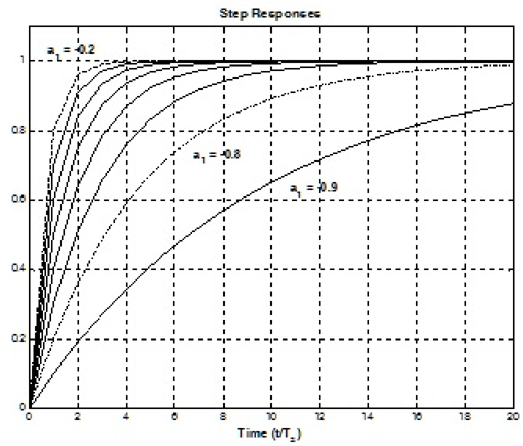

FIGURE 5 – Step responses of the discrete-time system given in Eq. 7 for different values of a_1 and $[b_1/(1+a_1)] = 1$.

Combining Eqs. 8 and 9 one gets the full system representation shown in Fig 6 . Computing the transfer

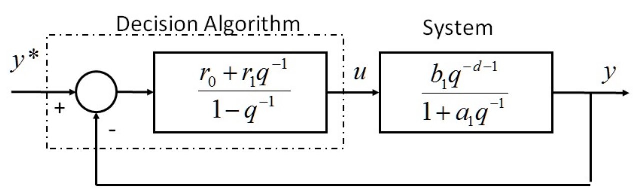

FIGURE 6 – Block diagram of the automated decision making system.

operator from y^* to y one gets :

$$H_{CL} = \frac{q^{-d-1}(r_0 + r_1 q^{-1})}{P(q^{-1}} \tag{10}$$

where $P(q^{-1})$ is called the characteristic polynomial of the closed loop. The necessary and sufficient condition for (asymptotic) stability is that the roots of this polynomial lie inside the *unit circle* (a circle of radius 1 centered in the 0 of the complex plane)([LZ05]. $P(q^{-1}$ has the expression :

$$P(q^{-1}) = 1 + (a_1 - 1)q^{-1} - a_1 q^{-2} + r_0 b_1 q^{d-1} + r_1 b_1 q^{d-2} \tag{11}$$

For the purpose of this contribution, it is enough to consider the particular case $d = 0$ in order to illustrate the stability issues in automated decision making systems. In this case $P(q^{-1}$ has the expression :

$$P(q^{-1}) = 1 + (a_1 - 1 + r_0 b_1)q^{-1} + (r_1 b_1 - a_1)q^{-2} \\ = 1 + p_1 q^{-1} + p_2 q^{-2} \quad (12)$$

This is a second order equation. In order that the system be asymptotically stable, the parameters of the decision algorithm r_0 and r_1 should be chosen such that the roots of $P(q^{-1})$ (real or complex) be inside the unit circle. The stability domain as a function of p_1 and p_2 is illustrated in Fig 7 (given a_1 and a_2 one can select r_0 and r_1 in order to get appropriate values for p_1 and p_2).

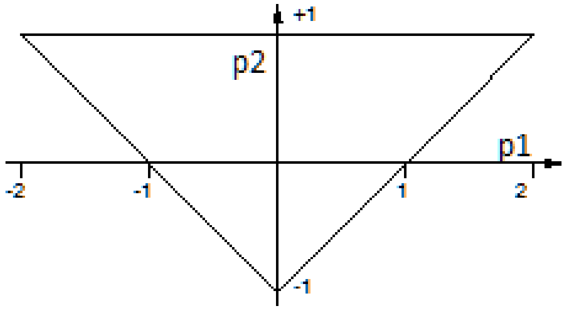

FIGURE 7 – Stability domain for a second order characteristic equation.

In the case of a stable operation the response of the system is shown in Fig 8 a for the case of real positive roots inside the unit circle (a periodic response) or as in Fig 9 for the case of complex roots located inside the unit circle (damped oscillatory response).

If the roots of the characteristic polynomial are outside the unit circle, the system is unstable. The time response can be an undamped oscillatory response as in Fig 10 (complex poles outside the unit circle) or an "explosive" type response as shown in Fig 11 (real poles outside the unit circle). These two figures are crucial for the operation of the supervision system and possible recovery of a stable operation. If a phenomenon of the form shown in Fig 10 occurs it may be possible when the target variable pass through a minimum to implement a stabilizing procedure. However if a phenomenon like in Fig ; 11 occurs there is no any chance to restore a stable operation. It is important to note that the disaster at Chernobyl occurred because of this type of instability in the loop controlling the power of the reactor ([G.03]). For this reason we will call this behavior the "Chernobyl phenomenon".

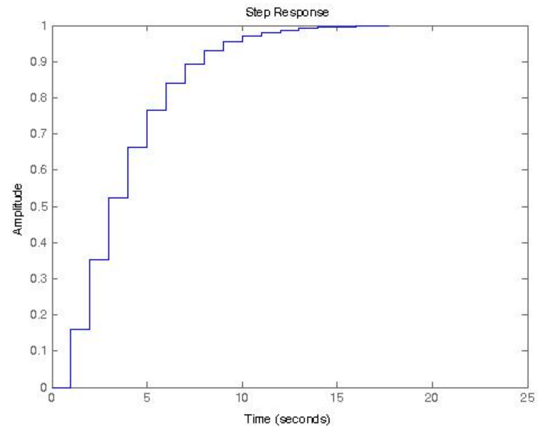

FIGURE 8 – Response of the automated model based desicion making for the case of real closed loop poles located inside the unit circle.

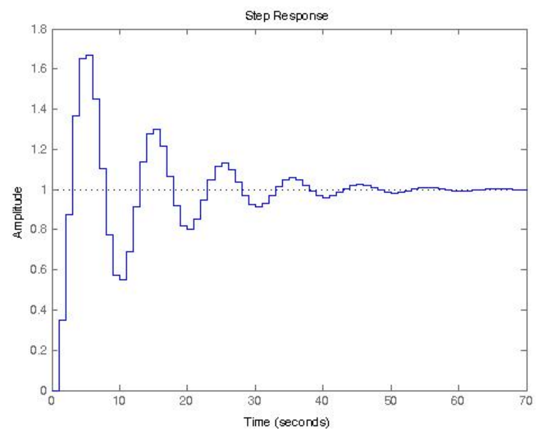

FIGURE 9 – Response of the automated model based desicion making for the case of complex closed loop poles located inside the unit circle.

4 Predictive decision strategies

Since the prediction model of the system which we would like to drive is available, one may consider to use it in order to predict the effect of our decisions over a certain horizon and develop adequate short term and long term decision algorithms. We will denote by $\hat{y}(t)$ the predicted target variable delivered by the prediction model.

Comparing the control objectives in the time domain associated with the various decision strategies one can classify these strategies in two categories :

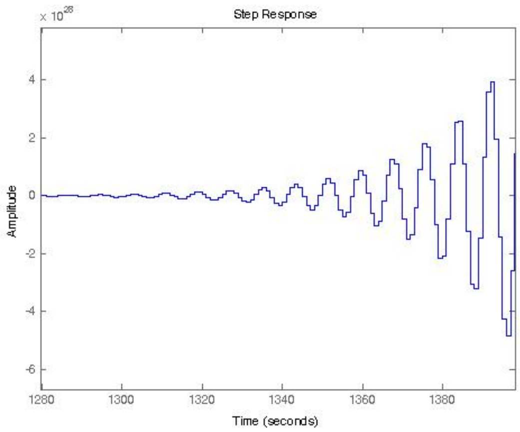

FIGURE 10 – Response of the automated model based decision making for the case of complex closed loop poles located outside the unit circle (unstable system).

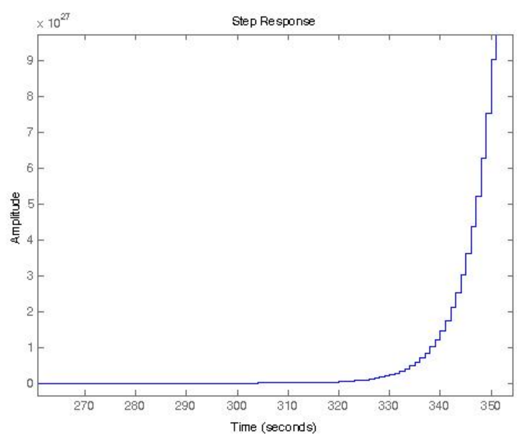

FIGURE 11 – The "Chernobyl phenomenon" (automated decision making system with real poles outside the unit circle) (unstable system).

1. *One step ahead predictive decision.* In these strategies one computes a prediction of the target variable at $t+d+1$ (d integer dead time of the sysstem) namely $\hat{y}(t+d+1)$ as a function of $u(t), u(t-1), \ldots, y(t), y(t-1), \ldots$ and one computes $u(t)$ such that a control objective in terms of $\hat{y}(t+d+1)$ be satisfied (i.e. such that $\hat{y}(t+d+1) = y*(t+d+1)$) This is illustrated in Fig. 12

2. *Long range predictive decision.* In these strategies the control objective is expressed in terms of the future values of the target variable over a certain horizon and of a sequence of future decisions

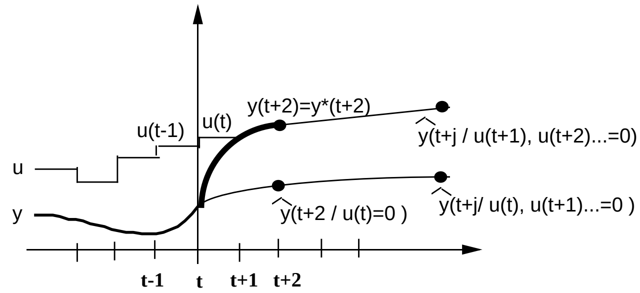

FIGURE 12 – Illustration of one step ahead predictive decision algorithm .

In order to solve the problem, one needs to compute : $\hat{y}(t+d+1), \hat{y}(t+d+2), \ldots, \hat{y}(t+d+j)$ which are expressed as :

$$\begin{aligned} \hat{y}(t+d+1) &= f_1(y(t), y(t-1), \ldots, u(t), u(t-1), \quad (13) \\ &\vdots \\ \hat{y}(t+d+j) &= f_j(y(t), y(t-1), \ldots, u(t), u(t-1), \ldots) \\ &\quad + g_j(u(t+1), \ldots, u(t+j-1)) \qquad (14) \end{aligned}$$

To satisfy the control objective the sequence of present and future values of the decisions $u(t), u(t+1), \ldots, u(t+j-1)$ is computed but only the first one (i.e., $u(t)$) is applied to the sysstem and the same procedure is restarted at $t+1$. This is called the *receding horizon procedure*.

The principle of long range predictive control is illustrated in Fig. 13 where the sequence of desired values $y^{\star}$, of predicted values $\hat{y}$ and the future decision sequences are represented (predicted values are represented for two different future decision sequences)[LLMK11, CE07].

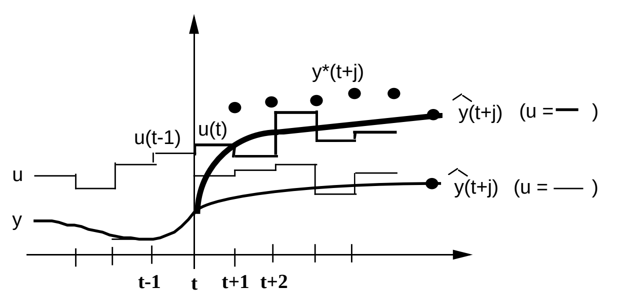

FIGURE 13 – Illustration of long range predictive decision algorithm.

All the decision strategies concern a linear design in the sense that constraints on the values of the admissible predictive variables (control) applied to the system are not considered. All the decision strategies will yield the same structure for the decision algorithm. The use of one or another strategy corresponds finally to different values of the parameters characterizing the decision algorithm whose structure as well as some parameters are defined by he prediction model used for design. Another important issue is that the control should be admissible (realizable) i.e., it should depend only on the information available up to and including time t when the control $u(t)$ is computed.

Extensions of these approaches for the case of constraints imposed on the predictive variables (control) exist, but they require considerable more computer resources and their analysis (in particular, the stability issue) is much more involved [J.M02].

5 An example of predictive decision

In using predictive decision, a key point is the computation of the d+1 step ahead prediction of the target variable based on the information available up to the instant t. One looks for an expression of the form given in Eq. 13. This will allow to compute u(t) in order that the target variable will reach the desired value at $t + d + 1$. We will illustrate this procedure for the case of a system described by the prediction model given in Eq. 8 with d=1.

$$y(t + 1) = -a_1 y(t) + b_1 u(t - 1) \tag{15}$$

We are interested in predicting the value of the target variable $y(t+2)$ at instant t on the basis of information available at instant t. In order to compute this predicted value, one expresses first the target variable y(t+2) as a function of the available information up to (and including) instant t. From Eq. 15 one gets

$$y(t + 2) = -a_1 y(t + 1) + b_1 u(t) \tag{16}$$

which can also be written

$$A(q^{-1})y(t + 2) = B^*(q^{-1})u(t) \tag{17}$$

$$A(q^{-1}) = 1 + a_1 q^{-1}; B^*(q^{-1}) = b_1 \tag{18}$$

Note that y(t+2), given by Eq. 16, depends upon $y(t + 1)$ which is unknown at instant t. Therefore this expression does not allow a two steps ahead prediction. But $y(t+1)$ can be replaced in Eq. 16 by its expression given by Eq. 15. One then gets

$$\begin{aligned} y(t + 2) &= -a_1[-a_1 y(t) + b_1 u(t - 1)] + b_1 u(t) \\ &= F(q^{-1})y(t) + E(q^{-1})B^*(q^{-1})u(t) \end{aligned} \tag{19}$$

with :

$$F(q^{-1}) = f_0 = {a_1}^2; E(q^{-1}) = 1 + e_1 q^{-1} = 1 - a_1 q^{-1} \tag{20}$$

One observes that the right hand member of Eq. 19 depends only on the information available at instant t and therefore the expression of the two steps ahead predictor will be given by :

$$\hat{y}(t + 2) = F(q^{-1})y(t) + E(q^{-1})B^*(q^{-1})u(t) \tag{21}$$

where $E(q^{-1})$ and $F(q^{-1})$ are given by Eq. 20. This technique of successive substitution of the one step ahead prediction can be generalized for any d,A and B^*. However, it is possible to directly find the polynomials $E(q^{-1})$ and $F(q^{-1})$. Using Eq. 16 in Eq. 19 for replacing the term $B^*(q^{-1})u(t)$, one gets :

$$\begin{aligned} y(t + 2) &= F(q^{-1})y(t) + E(q^{-1})A(q^{-1})y(t + 2) \\ &= [E(q^{-1})A(q^{-1}) + q^{-2}F(q^{-1})]y(t + 2) \end{aligned} \tag{22}$$

In order that the two sides of Eq. 22 be equal, one should verify the polynomial equation :

$$1 = E(q^{-1})A(q^{-1}) + q^{-2}F(q^{-1}) \tag{23}$$

In other terms, this means that the coefficients of the polynomials $E(q^{-1}$ and $F(q^{-1})$ required for the computation of the predicted value $\hat{y}(t+2)$ at instant t, are the solutions of the polynomial Eq. 23. This approach can be generalized for any A ,B^* and d (see [LZ05] for more details). Since one has a prediction of the target variable at instant $t + 2$ which depends on $u(t)$, it is now possible to compute $u(t)$ given the desired value of the target variable at $t + 2$, denoted $y^*(t + 2)$ using Eq. 19

$$y^*(t+2) = \hat{y}(t+2) = -a_1[-a_1 y(t)+b_1 u(t-1)]+b_1 u(t) \tag{24}$$

from which one obtains :

$$u(t) = \frac{y^*(t + 2) - {a_1}^2 y(t) + a_1 b_1 u(t - 1)}{b_1} \tag{25}$$

In the general case one would like to have

$$\begin{aligned} \hat{y}(t + d + 1) = y^*(t + d + 1) = F(q^{-1})y(t) \\ +E(q^{-1})B^*(q^{-1})u(t) \end{aligned} \tag{26}$$

from which one gets :

$$u(t) = \frac{1}{E(q^{-1})B^*(q^{-1})} y^*(t+d+1) - \frac{F(q^{-1})}{E(q^{-1})B*(q^{-1})} y(t) \quad (27)$$

Denoting :

$$S(q^{-1})E(q^{-1})B*(q^{-1}) = b_1 + q^{-1}S^*(q^{-1}) = b_1 + s_1 q^{-1} + s_2 q^{-2} + ... \quad (28)$$

one gets :

$$u(t) = \frac{1}{b_1}[y^*(t+d+1) - S^*(q^{-1})u(t-1) - F(q^{-1})y(t)] \quad (29)$$

As it can be observed, the decision at instant t depends upon the desired value of the target variable at a future instant and upon the current value of the target variable as well as on the previous values of $u(t)$ [3].

6 Conclusion

The present contribution has tried to present some basic features of automated data driven model based decision making systems. Two issues have been addressed : the stability issue which has to be considered since the full system has a feedabck structure (which can become unstable) and the predictive decision strategies which explicitly use the prediction model of the system.

Références

[AK86] Arzen K.E. Astrom K.J., Anton J.J. Expert control. *Automatica*, 22 :277–286, 1986.

[CE07] Bordons A.C. Camacho E.F. *Model Predictive Control.* Springer, London, 2007.

[FT13] Provost F. and Fawcett T. *Data Science for Business.* O'Reilly, 2013.

[G.03] Stein G. Respect the unstable. *IEEE Control Systems Magazine*, 23(4) :12–25, 2003.

[J.M02] Maciejovski J.M. *Predictive Control with Constraints.* Prentice Hall, London, 2002.

3. One makes the assumption that B^* is asymptotically stable.

[LLMK11] I. D. Landau, R. Lozano, M. M'Saad, and A. Karimi. *Adaptive control.* Springer, London, 2nd edition, 2011.

[LP15] Morel M. Raffaelli J.L. Lemberger P., Batty M. *Big Data et Machine Learning.* Dunod,Paris, 2015.

[LZ05] Ioan Dore Landau and G. Zito. *Digital Control Systems - Design, Identification and Implementation.* Springer, London, 2005.

[S.G01] Tzafestas S.G. Expert control systems. *Encyclopedia of Life Support Systems (EOLSS),UNESCO,http ://www.//www.eolss.net*, 17, 2001.