AUTOMATIC STUDY IN STOCHASTIC CONTROL

J.P.CHANCELIER, C.GOMEZ, J.P.QUADRAT, A.SULEM
INRIA Domaine de Voluceau ,BP105,78153 Le Chesnay (France)

Abstract:

The purpose of this paper is to give an example of automatic generation of a complete study in stochastic control done by an expert system designed at INRIA by the authors. This study includes the:
- *automatic choice of an algorithm,*
- *automatic checking of the mathematical well posedness of the problem,*
- *automatic generation of a numerical routine,*
- *automatic test of this routine on a numerical example,*
- *automatic generation of graphics,*
- *automatic writing of a report describing all the obtained results .*

1) INTRODUCTION

We give here the present state of an expert system about optimization and identification of dynamic systems, started four years ago. In this system we want to code the state of art in the domain, using the more advanced techniques in computer science: -inference engine, -symbolic manipulation, -numerical techniques. This system must be able to derive by itself classical theoretical results, to choose a method, to apply algorithms, to write programs, to make a numerical study, to design some graphics and to write a complete report from a specification in a specialized command language.

The present code has about 15000 lines of a mixture of LISP, MACSYMA, and PROLOG and needs a lot of improvements but has actually all the possibilities described above.

After a brief description of its mathematical capabilities in optimal control we give an example of a session to give an idea of the level of mathematical background needed by the user. Then we present the report generated by the expert system. This is an example where the dynamic programming approach has been chosen. It is the most developped part of the system.

The computer science aspect of the system has been described in Chancellier-Gomez-Quadrat-Sulem [43], the mathematical aspects briefly reported here are described more precisely in Theosys [21] .

A similar work for nonlinear filtering is developped by the Blankenship team at the Maryland University.

2) STOCHASTIC CONTROL

We define the stochastic control of diffusion processes following Kushner[25] Fleming-Rishel[15], Bensoussan[4], Bensoussan-Lions[6], by :

i) its **dynamics** :

(1) $dX_t=b(X_t,U_t)dt+hdW_t$

with :
- t the time,
- X_t the state of the system,
- U_t the control,
- W_t a brownian motion modelling of the stochastic perturbations,
- h the local standard deviation of the noise,
- b the drift of the system;

ii) the criterium to be minimized is one of the three following types:

(2) $Inf_U\ E\{_0\int^T c(X_t,U_t)\ dt\}$

with:
- T the horizon which can be a stopping time,
- c an instantaneous cost ;

(3) $Inf_U\ E\{_0\int^\infty e^{-lt}\ c(X_t,U_t)\ dt\}$

with l the discount rate,

(4) $Inf_U\ Lim_{T\to\infty}\ \{\ 1/T\ _0\int^T\ c(X_t,U_t)\ dt\ \}$.

3) ALGORITHMS

The system uses four kinds of methods to solve more or less completely the problem :
- **dynamic programming** approach,
- **decoupling** method,
- **stochastic gradient** technique,
- **perturbation** methods.

Each approach has its most efficient domain of application.

3.1) DYNAMIC PROGRAMMING

This approach leads to solve one of the following kinds of partial differential equation (PDE).

- The **finite horizon** case (2) :

(5) $D_tV + Inf_u\ \{\ b(x,u)\ D_xV + c(x,u)\ \} + tr(a\ D_{xx}V) = 0$

where V denotes the dynamic programming function defined by:

(6) $V(t,x)=\text{Inf } E\{ \int_t^T c(X_s,U_s)ds \mid X_t=x \}$;

- The **infinite horizon case with a discounted cost** (10) :

(7) $-lV + \text{Inf}_u \{ b(x,u).D_xV + c(x,u) \} + tr(a.D_{xx}V)=0$,

with :

(8) $V(x)= \text{Inf } E \{ \int_0^\infty e^{-ls} c(X_s,U_s)ds \mid X_0=x \}$;

- The **average cost by unit of time case** (11) :

(9) $-m + \text{Inf}_u \{ b(x,u).D_xW + c(x,u) \} + tr(a.D_{xx}W) = 0$,

with m defined by :

(10) $m = \text{Inf } \lim_{T\to\infty} 1/T \int_0^T c(X_s,U_s)\, ds$.

We have used the notation:

(11) $a=1/2h.h^*$.

This approach is the best one when it can be used , that is when the dimension of the system is less than 3 or 4. Indeed we have to solve a discretized version of the PDE numerically Kushner[25], Quadrat[34],[35], Goursat-Quadrat[22].

It is the only approach which gives a complete answer under general hypotheses needed in practise. More over we can use the aggregation method , for example Turgeon[42],Torrion[41], to reduce the dimension of the system.

This is the approach chosen by the system in the session example in the report generated. We can see the explanation obout the discretization of the dynamic programming equation and the proof of the existence and uniqueness of a solution of the Bellman Equation. In the future the system will use the better result of P.L.Lions described in these proceedings.

3.2) DECOUPLING METHOD

If the system has a special structure : **-uncoupled dynamics**, -computable criterium without numerical evaluation of multiple integrals on a large dimension space, then it is possible to optimize in the class of local feedbacks.

If the system has not this special structure, by changing the feedback it is sometimes possible to come down to this structure . It is the classical problem of decoupling in the automatic control litterature (Claude[8], Geromel-Lévine-Willis[16], in the non linear case).

Then the problem can be reduced to a PDE control problem. Indeed one controls the marginal laws of the density of probabilty of the state. By hypotheses on the structure, these laws are independent and satisfy Fokker-Planck equations .

More precisely given :

- an **uncoupled dynamic** :

(12) $dX_{it} = b_i(X_{it}, U_{it})dt + h_i dW_{it}$

(where i denotes the number of the subsystem)

- a coupling criterium **computable by convolutions or gaussian approximation** from the law of the subsystem:

(13) $E \int_0^T f(z_t - \Sigma_i c_i(X_{it}, U_{it}))dt$,

- z_t a function of time ,

we **minimize (13) in the class of local feedbacks.**

The problem comes down to **the control of the marginal laws** p_i of each subsystem given by :

(14) $- D_t p_i - D_x\{b(x_i, u_i)p_i\} + D_{xx}(a_{ii}p_i) = 0$

(15) $p(0,x)$ known ,

with the criterium :

(16) $\text{Min} \int_0^T f\{ z_t - \Sigma_i c_i(x_i, u_i(x_i)) \} \Pi_i p_i(t, x_i) dx_i$

where $p_i(t, x_i)$ denotes the density for X_{it} to be at x_i at time t.

This problem can be solved numerically. A theoretic study of the algorithm for a discrete version of the problem can be found in Quadrat-Viot[37]. An application to hydropower management is given in Delebecque-Quadrat[11]. A beautiful study of the loss of optimality is given in Torrion[41].

With this kind of approach we can solve Quadrat_Viot[37] some steady state control problem of Jackson type queuing network BCMP[2].

The possibilities of the system for automatic generation of numerical routines based on this method have been described in Gomez-Quadrat-Sulem[20].

3.3) STOCHASTIC GRADIENT

Sometimes we have an idea of a good class of feedback in which we would like to optimize.Then it is possible to use stochastic gradient techniques.

Denoting by $U(x,\mathbf{a})$ this parametized class, where **a** is a parameter, **a** becomes an open loop control to be optimized in the following control problem:

(17) $dX_t=b(X_t,U(X_t,A_t))dt+hdW_t$,

(18) $\mathrm{Inf}_A\ E\{{}_0\int^T c(X_t,U(X_t,A_t))dt\}$.

Then we have to compute the sequence indexed by n :

(19) $A_{n+1}=A_n-r_nD_AJ(w_n,A_n)$

(20) $J(w_n,A_n)= {}_0\int^T c(X_{tn},U(X_{tn},A_{tn}))dt$

(21) $\Sigma_n r_n=\infty,\ r_n>0,\ r_n\to 0$

where w_n denotes independent simulation of trajectories of the noise.

A_n converges to the optimum if we have made some convexity assumptions Polyak[31],[32], Kushner-Clark[26], Dodu-Goursat-Hertz-Quadrat-Viot[12]. The computational cost increases linearly with the dimension of the system so it is particularly useful for large system. For small size systems dynamic programming gives a more complete answer.

3.4) REGULAR PERTURBATION

When the size of the noise is small, we can compute affine feedback good up to order 4 Cruz[10], Fleming[14], Bensoussan[5], for enough regular systems.

We are in finite horizon problem case (13) with :

-the dynamics :

(22) $dX_t=b(X_t,U_t)dt+hdW_t$, $h \ll 1$,

-the affine following feedback :

(23) $\mathbf{u(t,x)=u_o(t)+K(t)(x-x_o(t))}$

where (u_o,x_o) is the solution of the following deterministic control problem (h=0):

(24) $dx_o(t)=b(x_o(t),u_o(t))dt$,

(25) Inf $_0\int^T c(x_o(t),u_o(t))dt$,

and K(t) is the solution of the tangent linear quadratic problem :

(26) $K = -H_{uu} (H_{ux} + b_u P)$,

(27) $P' + A^*P + PA - PSP + Q = 0$,

(28) $A = b_x - b_u H_{uu}^{-1} H_{ux}$,

(29) $S = b_u H_{uu}^{-1} b_u^*$,

(30) $Q = H_{xx} - H_{ux}^* H_{uu}^{-1} H_{ux}$,

(31) H=b.p+c where p denotes the dual variables of the states, gives, when (26)......(31) is well posed, a feedback which has a **loss of optimality of order h^4.**

This method is the best one for stabilization problems.

The system knows all these methods and is able to generate a study based on all of these point of view. The more advanced part of the system is the dynamic programming part. It is for example the only method for which the system is able to make the theoretical work.

It is able to chose the method then it checks the well-posedness of the HJB equation if the dynamic programming approach has been chosen, it generates numerical subroutines and test them on a numerical example. Finally it generates the report.

The system is written in LISP,MACSYMA,PROLOG. The generated numerical program is in FORTRAN. Prolog is used to organize the program, Macsyma is used for the algebraic manipulations and for building the numerical routines, the 3D plotting, the scientific editing. Lisp is used for a good integration of Prolog and Macsyma and for general purpose programming. Let us recall that the Prolog used, called Oblogis, is written in Lisp like Macsyma. The system is currently under development on the Symbolics Lisp Machine and a former version exists on Multics.

4) A SESSION EXAMPLE OF THE PRESENT SYSTEM

We give an example of the interaction of the system with the user and the generated report .

```
(robot)

NOUS ALLONS ESSAYER DE RESOUDRE VOTRE PROBLEME DE CONTROLE STOCHASTIQUE
ENONCEZ LE PROBLEME
EN CAS DE DIFFICULTES TAPEZ help
PRECISEZ TOUT D'ABORD LE NUMERO OU LE NOM DE VOTRE PROBLEME EN TAPANT PAR EXEMPLE pr
obleme 0
PRECISEZ EGALEMENT LA LANGUE DANS LAQUELLE LE RAPPORT DEVRA ETRE REDIGE EN TAPANT ve
rsion francaise OU version anglaise
===>  VERSION ANGLAISE
Loading LM1:>sulem>new>clauses-text.lisp into package MACSYMA
Loading LM1:>sulem>new>texte-anglais.lisp into package MACSYMA
===>  HELP
VOUS AVEZ BESOIN D ' AIDE POUR
- ENTRER LA DYNAMIQUE DU PROBLEME (1)
- DONNER L ' HORIZON ET LES CONDITIONS AUX LIMITES (2)
- QUESTIONNER OU MODIFIER LA BASE DE CONNAISSANCE DU ROBOT (3)
- GENERER LES PROGRAMMES FORTRAN , OBTENIR LES RESULTATS THEORIQUES ET NUMERIQUES ,
GENERER LE RAPPORT ET LES GRAPHES (4)
TAPER LE NUMERO CORRESPONDANT
1
LE ROBOT COMPREND LES PHRASES DU TYPE SUIVANT :
ON PEUT UTILISER N ' IMPORTE LAQUELLE DES EXPRESSIONS SEPAREES PAR LE SYMBOLE /
   PROBLEME DE COMMANDE/EVALUATION_COUT_MOYEN/CALCUL_DENSITE_PROBABILITE
   X EST UN ETAT/COMMANDE/PARAMETRE
   1 EST LE COUT-INSTANTANE
   LA MOYENNE-DYNAMIQUE/VARIANCE-DYNAMIQUE EN X1 EST 1
DANS LE CAS D ' UN CALCUL DE DENSITE DE PROBABILITE , PRECISEZ :
   INVARIANTE/MARGINALES EST LA LOI
   LA SIGNIFICATION/UNITE/DOMAINE-DE-VARIATION DE X EST ...
===>  PROBLEME 0
PROBLEME-GLOBAL 0
===>  X1 IS A STATE VARIABLE
JE NE COMPRENDS PAS
===>  X1 IS A STATE
JE NE COMPRENDS PAS
===>  BIEN-POSE

 WHAT IS THE TYPE (type)

- COMMAND (commande) ,

- EVALUATION OF AN AVERAGE COST (evaluation_cout_moyen),

- PROBABILITY DENSITY COMPUTATION (calcul_densite_probabilite) OF THE PB 0 ?
COMMANDE;

 WHAT IS THE horizon

-FINITE (fini),

-INFINITE (infini),

-ERGODIC (ergodique) FOR THE PB 0 ?
INFINI;

 WHAT IS THE TIME NAME (temps) FOR THE PB 0 ?
T;
```

```
WHAT IS THE LIST OF THE STATE NAMES (etat)  FOR THE PB 0 ?
[X1,X2];

WHAT IS THE LIST OF THE PARAMETER NAMES (parametre) FOR THE PB 0 ?
[];

WHAT IS THE LIST OF THE PRICE NAMES (prix) DU PB 0 ?
[P1,P2];

WHAT IS THE LIST OF THE PRICE DERIVATIVE NAMES (derivee-prix) FOR THE PB 0 ?
[Q1,Q2];

WHAT IS THE  LIST OF THE COMMAND NAMES (commande) OF THE PB 0 ?
[U1,U2];

 WHAT IS THE NAME OF THE OPTIMAL COST (cout-optimal) FOR THE PB 0 ?
V;

WHAT IS THE AVERAGE OF THE DYNAMIC (moyenne-dynamique), FOR THE PB 0 ,OF X1 ?
3*X1+U1;

WHAT IS THE VARIANCE OF THE DYNAMIC (variance-dynamique), FOR THE PB 0 ,OF X1 ?
1+X1^2;

WHAT IS THE AVERAGE OF THE DYNAMIC (moyenne-dynamique), FOR THE PB 0 ,OF X2 ?
3*X2+U2;

WHAT IS THE VARIANCE OF THE DYNAMIC (variance-dynamique), FOR THE PB 0 ,OF X2 ?
1+X2^2;

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF X1 ?
[0,1];

WHAT IS THE BOUNDARY CONDITION (condition-frontiere) FOR THE PB 0 AT X1 = 0 ?
[REFLECHI,0];

WHAT IS THE BOUNDARY CONDITION (condition-frontiere) FOR THE PB 0 AT X1 = 1 ?
[REFLECHI,0];

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF X2 ?
[0,1];

WHAT IS THE BOUNDARY CONDITION (condition-frontiere) FOR THE PB 0 AT X2 = 0 ?
[ARRET,0];

WHAT IS THE BOUNDARY CONDITION (condition-frontiere) FOR THE PB 0 AT X2 = 1 ?
[ARRET,0];

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF U1 ?
[-1,1];

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF U2 ?
[-1,1];

 WHAT IS THE VALUE OF THE INSTANTANEOUS COST (cout-instantane) FOR THE PB 0 ?
U1^2+U2^2+(X1-1/2)^2+X2^2;

WHAT IS THE actualisation RATE OF THE PB 0 ?
3;
```

===> **METHODE**

METHODE : PROGRAMMATION-DYNAMIQUE

===> **RESOUDRE ?PPD**

DO YOU WANT THE SUBPROGRAM OUTPUT (sortie) YES (oui),NO (non) FOR THE PB 0 ?

OUI;

WHAT IS THE WANTED precision FOR THE PB 0 ?

0.1;

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF P1 ?

[-1,1];

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF P2 ?

[-1,1];

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF Q1 ?

[-1,1];

WHAT IS THE DOMAIN OF VARIATION (domaine-de-variation) , FOR THE PB 0 ,OF Q2 ?

[-1,1];

===> **HELP**

VOUS AVEZ BESOIN D ' AIDE POUR
- ENTRER LA DYNAMIQUE DU PROBLEME (1)
- DONNER L ' HORIZON ET LES CONDITIONS AUX LIMITES (2)
- QUESTIONNER OU MODIFIER LA BASE DE CONNAISSANCE DU ROBOT (3)
- GENERER LES PROGRAMMES FORTRAN , OBTENIR LES RESULTATS THEORIQUES ET NUMERIQUES , GENERER LE RAPPORT ET LES GRAPHES (4)

TAPER LE NUMERO CORRESPONDANT

4

ON GENERE LE SOUS-PROGRAMME/PROGRAMME PRINCIPAL EN TAPANT PAR EX :
 SOUS-PROGRAMME/PROGRAMME-PRINCIPAL ?PPD
POUR GENERER LE SOUS-PROGRAMME ,LE PROGRAMME PRINCIPAL ET EXECUTER ON TAPE:
 RESOUDRE ?PPD
POUR AVOIR LE GRAPHE DE V EN FONCTION DE (X1 X2) POUR LA METHODE PROGRAMMATION DYNAMIQUE , ON TAPE :
 GRAPHE V [X1,X2] ?PPD
POUR AVOIR LES FIGURES DE V ET U AVEC LES COURBES DE NIVEAU :
 FIGURES V U ?PPD
POUR AVOIR LES RESULTATS THEORIQUES (EXISTENCE D'UNE SOLUTION) :
 DEMONSTRATION
POUR GENERER LE RAPPORT :
 RAPPORT

===> **DEMONSTRATION**

In the hamiltonian H(V) :

$$H(V) = \underset{U}{MIN} \left(U_2 \frac{dV}{dX_2} + U_1 \frac{dV}{dX_1} + U_2^2 + U_1^2\right)$$

- the coefficients $[U_1 , U_2]$ are bounded by 1
- the coefficient $U_2^2 + U_1^2$ is bounded by 2.

The hamiltonian H(V) is lipschitzian with coefficient 1.4142135.

The variational formulation of the problem in the space H^1 is :

$$C(V, W) = - \int (H(V)\ W) + A(V, W) = \int ((X_2^2 + (X_1 - 0.5)^2)\ W)$$

where A(V, W) is the linear part :

$$A(V, W) = \int \left((X_2^2 + 1) \frac{dV}{dX_2} \frac{dW}{dX_2}\right) + \int \left((X_1^2 + 1) \frac{dV}{dX_1} \frac{dW}{dX_1}\right) - \int \left(X_2 \frac{dV}{dX_2} W\right) - \int \left(X_1 \frac{dV}{dX_1} W\right) + 3 \int (V\ W)$$

The linear part is coercive in the space H^1 :

$$A(V, V) >= \frac{(9235521.0\ TETA^2 + 2.9180478e7\ TETA + 1.0)\ NORM^2(V, L^2)}{1.191288e7\ TETA + 3453520.0} - 2.5510203e{-4}\ (3039.0\ TETA - 3039.0)\ NORM^2(V, H^1)$$

with 0.0 < TETA < 1.0.

The problem is bounded in the space H^1
because the coefficients of the hamiltonian are bounded.

The problem is hemicontinuous in the space H^1
because the hamiltonian is lipschitzian.

The problem is monotonous in the space H^1 with
0.11699876 < TETA < 0.693925

The problem is inf-compact in the space H^1 with
0.355051 < TETA < 0.717224

The problem has a unique solution in the space H^1.

```
===> GRAPHE V [X1,X2] ?PPD
===> GRAPHE U1 [X1,X2] ?PPD
===> FIGURES V U ?PPD
===> RAPPORT
Written: LM1:>quadrat>fortran>rapport.fortran.61
===> HELP
VOUS AVEZ BESOIN D ' AIDE POUR
- ENTRER LA DYNAMIQUE DU PROBLEME (1)
- DONNER L ' HORIZON ET LES CONDITIONS AUX LIMITES (2)
- QUESTIONNER OU MODIFIER LA BASE DE CONNAISSANCE DU ROBOT (3)
- GENERER LES PROGRAMMES FORTRAN , OBTENIR LES RESULTATS THEORIQUES ET NUMERIQUES ,
GENERER LE RAPPORT ET LES GRAPHES (4)
TAPER LE NUMERO CORRESPONDANT
3
ON QUESTIONNE LE ROBOT PAR DES PHRASE DU TYPE :
```

```
   QUELLE EST LA COMMANDE
   QUELLES SONT LES CONDITION-FRONTIERE
POUR SUPPRIMER UNE CLAUSE ON TAPE :
   SUPPRIMER LE COUT-FINAL
POUR LISTER LES DONNEES DU PROBLEME ON TAPE :
   DONNEES
POUR VERIFIER QUE LE PROBLEME EST BIEN POSE ON TAPE :
   BIEN-POSE
POUR SORTIR DU SYSTEME ON TAPE
   STOP
===> DONNEES
ACTUALISATION 0 3
COMMANDE 0 U1
COMMANDE 0 U2
CONDITION-FRONTIERE 0 X1 0 REFLECHI 0
CONDITION-FRONTIERE 0 X1 1 REFLECHI 0
CONDITION-FRONTIERE 0 X2 0 ARRET 0
CONDITION-FRONTIERE 0 X2 1 ARRET 0
                            2          1 2     2     2
COUT-INSTANTANE 0 X2  + (X1 - -)  + U2  + U1
                                       2
COUT-OPTIMAL 0 V
DERIVEE-PRIX 0 Q1
DERIVEE-PRIX 0 Q2
DOMAINE-DE-VARIATION 0 X1 0 1
DOMAINE-DE-VARIATION 0 X2 0 1
DOMAINE-DE-VARIATION 0 U1 - 1 1
DOMAINE-DE-VARIATION 0 U2 - 1 1
DOMAINE-DE-VARIATION 0 P1 - 1 1
DOMAINE-DE-VARIATION 0 P2 - 1 1
DOMAINE-DE-VARIATION 0 Q1 - 1 1
DOMAINE-DE-VARIATION 0 Q2 - 1 1
ETAT 0 X1
ETAT 0 X2
HORIZON 0 INFINI
MOYENNE-DYNAMIQUE 0 X1 3 X1 + U1
MOYENNE-DYNAMIQUE 0 X2 3 X2 + U2
PRECISION 0 0.1
PRIX 0 P1
PRIX 0 P2
PROBLEME 0
SORTIE 0 OUI
TEMPS 0 T
TYPE 0 COMMANDE
                              2
VARIANCE-DYNAMIQUE 0 X1 X1  + 1
                              2
VARIANCE-DYNAMIQUE 0 X2 X2  + 1
METHODE-POSSIBLE 0 PROGRAMMATION-DYNAMIQUE
DISCRETISATION 0 ESPACE [0, 0]
OPTIMISATION 0 GRADIENT-PROJECTION
NB-PT-DISCRETISATION-ESPACE 0 [11, 11]
NB-PT-DISCRETISATION-TEMPS 0 11
NB-MAXI-ITERATION-OPTIMISATION 0 24
NB-MAXI-ITERATION-IMPLICITE 0 1232
PAS-GRADIENT 0 0.25
PAS-IMPLICITE 0 0.00124533
COEF-CONTRACTION 0 0.0037359898
PRECISION-RES-IMPLICITE 0 0.0015068494
PLOT 0 VPERSP
```

```
PLOT 0 VCONTOUR
PLOT 0 U1PERSP
PLOT 0 U1CONTOUR

===>  STOP
AU REVOIR
```

A DISCOUNTED 2-DIMENSIONAL STOCHASTIC CONTROL PROBLEM

T. PANDORE

INRIA Domaine de Voluceau BP 105 Rocquencourt 78153 Le Chesnay France

ABSTRACT

We consider a static system the 2-dimensional state of which is modelled by a controlled diffusion process defined on [0,1]x[0,1]. The purpose is to minimize the expected discounted cost that includes a integral cost and stopping and reflecting costs. The optimal cost satisfies a Bellman equation derived from the optimal principle of Dynamic Programming. The Dynamic Programming equation is discretized and then solved numerically.

1 - Notations

State variables: [X1, X2]
Control variables : [U1, U2]
Time: T
Optimal cost: V
State dimension : N
ith state variable : XI
Derivative Operator with respect to the time variable: D_0
Derivative Operator with respect to XI: D_I

2 - Evolution equation of the system:

We consider the control diffusion process defined by the dynamic equation :

$$(0-2)\quad dX2_T = (3\,X2 + U2)\,dT + SQRT(2\,X2^2 + 2)\,.\,dW2_T$$

$$(1-1)\quad dX1_T = (3\,X1 + U1)\,dT + SQRT(2\,X1^2 + 2)\,.\,dW1_T - dZ1_1 + dZ1_0$$

where

X1 belongs to [0, 1]
X2 belongs to [0, 1]
U1 belongs to [- 1, 1]
U2 belongs to [- 1, 1]

W_T is a Wiener process, i.e. a continuous gaussian process with independent increments.

Z_J is an increasing process, strictly increasing when X_T is on the boundary $X = J$.

$X1_T$ is reflected on the boundary $X1 = 0$.

$X1_T$ is reflected on the boundary $X1 = 1$.

This process is well defined [7]. It is the limit when the time step h goes to 0 of a markovian discrete process X^h_n which satisfies:

(i) $$E(X^h_{n+1} - X^h_n \mid F_n) = h \begin{bmatrix} 3\,X1 + U1 \\ 3\,X2 + U2 \end{bmatrix} + O(h)$$

(ii) $$E\left((X^h_{n+1} - X^h_n)^{\otimes 2} \mid F_n\right) = h \begin{bmatrix} X1^2 + 1 & 0 \\ 0 & X2^2 + 1 \end{bmatrix} + O(h)$$

(iii) A uniform integrability condition of the increment $X^h_{n+1} - X^h_n$

where F_n denotes the sigma-algebra generated by $X_0, X_1, \ldots, X_n$.

3 - Value function

The stopping time TF is defined by :

$$TF = MIN(T2_1 , T2_0)$$

with

$$T2_0 = \inf \{T \,/\, X2_T \leq 0\}$$
$$T2_1 = \inf \{T \,/\, X2_T \geq 1\}$$

The problem is to minimise the expectation of the function :

(2) $$J(S) = \int_0^{TF} \%E^{-3\,T} \left(X2^2 + (X1 - \frac{1}{2})^2 + U2^2 + U1^2\right)_T dT$$

in the feedbacks class, i.e. the applications :

$S : [X1, X2] \longrightarrow [U1, U2]$.

4 - Optimality conditions :

The Bellman function V is defined by:

(3) $V(Y1, Y2) = \min_S \left(E \left[J(S) \mid [X1_0 = Y1, X2_0 = Y2] \right] \right)$

V satisfies the Dynamic Programming equation [2] [1]:

(4) $$\min_{U1,\, U2} \left((X2^2 + 1) \frac{d^2 V}{dX2^2} + (3\, X2 + U2) \frac{dV}{dX2} + (X1^2 + 1) \frac{d^2 V}{dX1^2} + (3\, X1 + U1) \frac{dV}{dX1} + X2^2 + \left(X1 - \frac{1}{2}\right)^2 + U2^2 + U1^2 \right) - 3\, V(X1, X2) = 0$$

$$V(X1, 0) = 0$$
$$V(X1, 1) = 0$$
$$\frac{d}{dX1} V(0, X2) = 0$$
$$\frac{d}{dX1} V(1, X2) = 0$$

5 - Theoretical Analysis

In the hamiltonian H(V) :

$$H(V) = \min_U \left(U_2 \frac{dV}{dX_2} + U_1 \frac{dV}{dX_1} + U_2^2 + U_1^2 \right)$$

- the coefficients $[U_1, U_2]$ are bounded by 1
- the coefficient $U_2^2 + U_1^2$ is bounded by 2.

The hamiltonian H(V) is lipschitzian with coefficient 1.4142135.

The variational formulation of the problem in the space H_1 is :

$$C(V, W) = -\int (H(V)\, W) + A(V, W) = \int ((X_2^2 + (X_1 - 0.5)^2)\, W)$$

where A(V, W) is the linear part :

$$A(V, W) = \int \left((X_2^2 + 1) \frac{dV}{dX_2} \frac{dW}{dX_2} \right) + \int \left((X_1^2 + 1) \frac{dV}{dX_1} \frac{dW}{dX_1} \right) - \int \left(X_2 \frac{dV}{dX_2} W \right) - \int \left(X_1 \frac{dV}{dX_1} W \right) + 3 \int (V\, W)$$

The linear part is coercive in the space H^1 :

$$A(V, V) >= \frac{(9235521.0\, TETA^2 + 2.9180478e7\, TETA + 1.0)\, NORM(V, L^2)^2}{1.191288e7\, TETA + 3453520.0} - 2.5510203e{-4}\, (3039.0\, TETA - 3039.0)\, NORM(V, H^1)^2$$

with $0.0 < TETA < 1.0$.

The problem is bounded in the space H^1
because the coefficients of the hamiltonian are bounded.

The problem is hemicontinuous in the space H^1
because the hamiltonian is lipschitzian.

The problem is monotonous in the space H^1 with
$0.11699876 < TETA < 0.693925$

The problem is inf-compact in the space H^1 with
$0.355051 < TETA < 0.717224$

The problem has a unique solution in the space H^1 .

6 - Dynamic programming method

Our purpose is to solve the Bellman equation (4) after
discretization [5] [6] [3] [4].
This is possible because the state dimension is small.

6-1 Discretization:

We denote:

- HI : discretization step for the i-th space variable XI

We define the following operators:

S_I : V(X1,..,XI,..,XN) --> V(X1,..,XI+HI,..,XN) I=1,..,N

$$\delta_I = \frac{S_I - 1}{HI}$$

$$\partial_I = \frac{\delta_I}{2\, S_I} + \frac{\delta_I}{2}$$

$$\gamma_I = \frac{\delta_I^2}{S_I}$$

We thus approximate:

$$\frac{d^2 V}{dXI^2} \text{ by } \gamma_I(V)$$

$$\frac{dV}{dX1} \text{ by } \partial_1(V)$$

$$\frac{dV}{dX2} \text{ by } \partial_2(V)$$

The discretized Bellman equation is:

$$(5)\quad \min_{U1,\, U2} \left((3\, X2 + U2)\, \partial_2(V) + (X2^2 + 1)\, \gamma_2(V) + (3\, X1 + U1)\, \partial_1(V) + (X1^2 + 1)\, \gamma_1(V) + X2^2 + \left(X1 - \frac{1}{2}\right)^2 + U2^2 + U1^2 \right) - 3\, V(X1, X2) = 0$$

- reordering with respect to S_I we get:

$$(6)\quad \min_{U1,\, U2} \left(- \left(\left(\frac{8\, X2^2}{H2^2} + \frac{8\, X1^2}{H1^2} + \frac{8}{H2^2} + \frac{8}{H1^2} \right) V \right.\right.$$

$$+ \left(S_2^{<-1>} . V\right) \left(- \frac{4\ X2^2}{H2^2} + \frac{6\ X2}{H2} + \frac{2\ U2}{H2} - \frac{4}{H2^2}\right)$$

$$+ \left(S_2 . V\right) \left(- \frac{4\ X2^2}{H2^2} - \frac{6\ X2}{H2} - \frac{2\ U2}{H2} - \frac{4}{H2^2}\right) - 4\ X2^2$$

$$+ \left(S_1^{<-1>} . V\right) \left(- \frac{4\ X1^2}{H1^2} + \frac{6\ X1}{H1} + \frac{2\ U1}{H1} - \frac{4}{H1^2}\right)$$

$$+ \left(S_1 . V\right) \left(- \frac{4\ X1^2}{H1^2} - \frac{6\ X1}{H1} - \frac{2\ U1}{H1} - \frac{4}{H1^2}\right) - 4\ X1^2 + 4\ X1 - 4\ U2^2 - 4\ U1^2 - 1)/4)$$

$$- 3\ V(X1, X2) = 0$$

6-2 Probabilistic Interpretation of the discretized equation :

The discretization of the Bellman equation

$$(7) \quad \min_{U1,\ U2} \left(A\ V + C(U1, U2)\right) - \lambda\ V = 0$$

can be interpreted as a control problem of Markov chain with discount factor k and instantaneous cost kC.The associated cost function is

$$(8) \quad k \sum_{N=0}^{\infty} C(X_N, U_N)\ (\lambda\ k + 1)^{-N-1}$$

and the Markov matrix M:

$$M = k\ A + I$$

where I is the Identity matrix,
and k : inverse of the maximum of the diagonal of A

is given by:

INITIAL_PT	FINAL_PT	TRANSITION_PROBABILITY
[X1, X2]	[X1, X2]	0
[X1, X2]	[X1 + H1, X2]	$\dfrac{\dfrac{2\,X1^2}{H1^2} + \dfrac{3\,X1}{H1} + \dfrac{U1}{H1} + \dfrac{2}{H1^2}}{2\left(\dfrac{2\,X2^2}{H2^2} + \dfrac{2\,X1^2}{H1^2} + \dfrac{2}{H2^2} + \dfrac{2}{H1^2}\right)}$
[X1, X2]	[X1, X2 + H2]	$\dfrac{\dfrac{2\,X2^2}{H2^2} + \dfrac{3\,X2}{H2} + \dfrac{U2}{H2} + \dfrac{2}{H2^2}}{2\left(\dfrac{2\,X2^2}{H2^2} + \dfrac{2\,X1^2}{H1^2} + \dfrac{2}{H2^2} + \dfrac{2}{H1^2}\right)}$
[X1, X2]	[X1 - H1, X2]	$\dfrac{\dfrac{2\,X1^2}{H1^2} - \dfrac{3\,X1}{H1} - \dfrac{U1}{H1} + \dfrac{2}{H1^2}}{2\left(\dfrac{2\,X2^2}{H2^2} + \dfrac{2\,X1^2}{H1^2} + \dfrac{2}{H2^2} + \dfrac{2}{H1^2}\right)}$
[X1, X2]	[X1, X2 - H2]	$\dfrac{\dfrac{2\,X2^2}{H2^2} - \dfrac{3\,X2}{H2} - \dfrac{U2}{H2} + \dfrac{2}{H2^2}}{2\left(\dfrac{2\,X2^2}{H2^2} + \dfrac{2\,X1^2}{H1^2} + \dfrac{2}{H2^2} + \dfrac{2}{H1^2}\right)}$

Indeed if the following conditions are satisfied:

$$(9\text{-}1)\quad H1 \le \sup_{X1,\, X2,\, U1,\, U2} \left(\frac{2\,(X1^2 + 1)}{ABS(3\,X1 + U1)}\right)$$

$$(9\text{-}2)\quad H2 \le \sup_{X1,\, X2,\, U1,\, U2} \left(\frac{2\,(X2^2 + 1)}{ABS(3\,X2 + U2)}\right)$$

the matrix coefficients are positive and the sum of the coefficients on a same line is equal to 1.The matrix M is thus a transition matrix of a Markov chain.Moreover the optimal cost obeys:

$$(10)\quad (\lambda\, k + 1)\, V = \min_{U1,\, U2} (M(U1,\, U2)\, V + k\, C(U1,\, U2))$$

Thus we can use the contraction iteration:

$$(11)\quad V_{N+1} = \frac{\min_{U1,\, U2} (M(U1,\, U2)\, .\, V_N + k\, C(U1,\, U2))}{\lambda\, k + 1}$$

6-3 Optimisation method:

We denote by H the Hamiltonian defined by:

$$(12)\quad (X2^2 + 1)\frac{d^2 V}{dX2^2} + (3\,X2 + U2)\frac{dV}{dX2} + (X1^2 + 1)\frac{d^2 V}{dX1^2} + (3\,X1 + U1)\frac{dV}{dX1} + X2^2 + \left(X1 - \frac{1}{2}\right)^2 + U2^2 + U1^2$$

H is minimised by a projected gradient method:

$$(13)\quad U_{N+1} = PROJ_{[-1,\, 1]}\left(U_N - \left(\frac{d}{dU_N}(H(U_N))\right) R\right)$$

that is:

$$
(14)\quad \left[\begin{aligned} U1_{N+1} &= PROJ_{[-1,\,1]}\left(U1_N - R\left(\frac{dV}{dX1} + 2\,U1_N\right)\right) \\ U2_{N+1} &= PROJ_{[-1,\,1]}\left(U2_N - R\left(\frac{dV}{dX2} + 2\,U2_N\right)\right) \end{aligned}\right]
$$

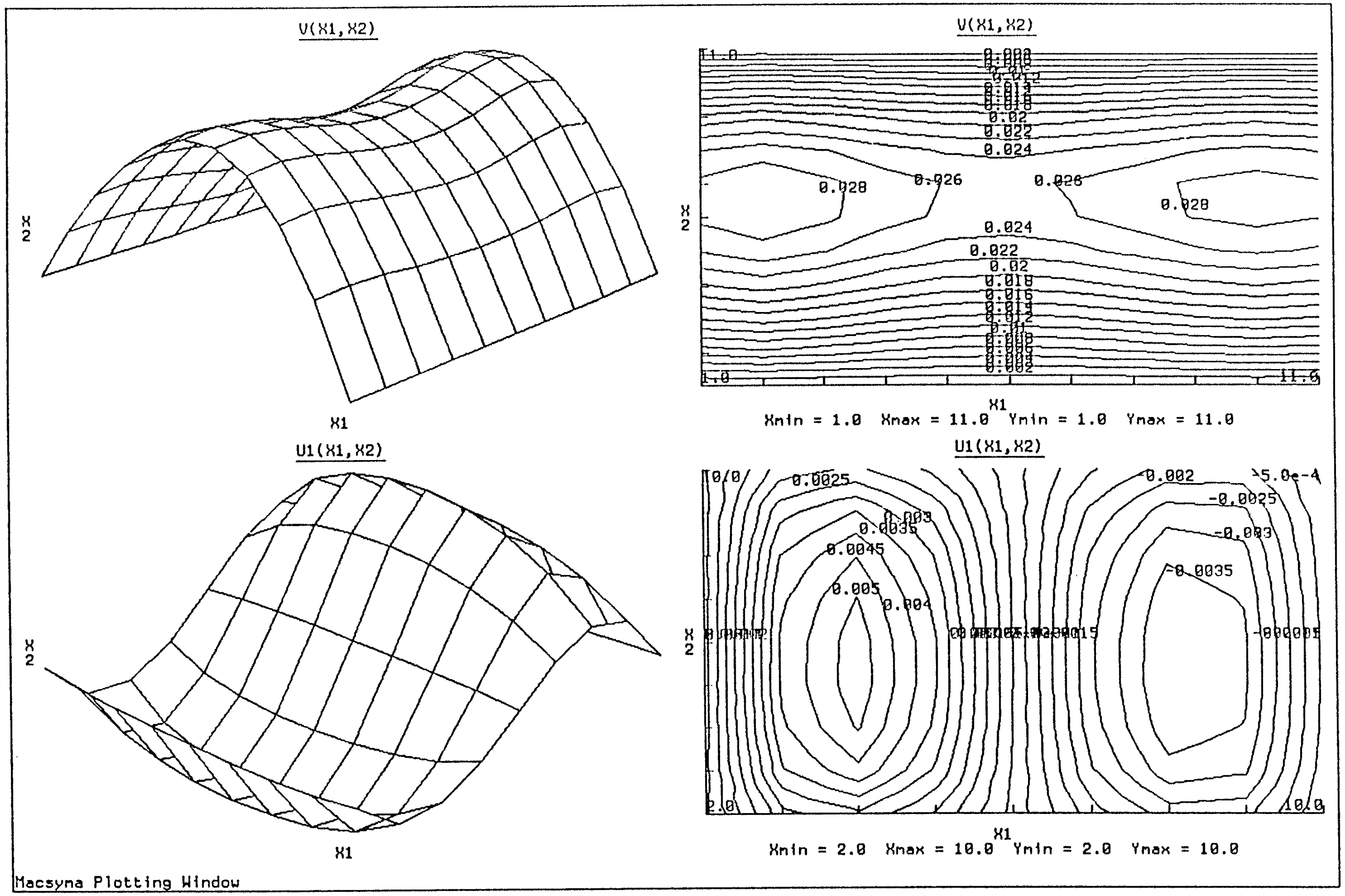
This algorithm converges when the step R satisfies:

$$
(15)\quad 0 < R < \frac{2\,m}{M^2}
$$

with:

$$
(16)\quad m\,|V|^2 \le D^2_U H(V)\,.\,V \le M\,|V|^2
$$

6-4 Numerical results


V(X1,X2)
X1
X2
V(X1,X2)
0.028
0.026
0.024
0.022
0.02
X1
Xmin = 1.0 Xmax = 11.0 Ymin = 1.0 Ymax = 11.0
U1(X1,X2)
X1
X2
U1(X1,X2)
0.0025
0.0035
0.0045
0.005
0.004
-0.002
-0.0025
-0.003
-0.0035
-5.0e-4
X1
Xmin = 2.0 Xmax = 10.0 Ymin = 2.0 Ymax = 10.0
Macsyma Plotting Window

Annex 1 Main Program

```
      PROGRAM M_PPD
      REAL V(11,11),U(2,11,11)
C
        do 1002 K1=1,11
C
          do 1001 K2=1,11
            V(K1,K2)=0.0
C
              do 1000 J=1,2
              U(J,K1,K2)=0.0
 1000         CONTINUE
C                fin de do
C
 1001     CONTINUE
C            fin de do
C
 1002   CONTINUE
C          fin de do
C
      CALL PPD(11,11,V,0.0015068494,1232,0.00124533,U,0.010000001,24,0.2
     1   5)
      STOP
      END
```

Annex 2 Subroutine solving the dynamic programming equation

```
      SUBROUTINE PPD(N1,N2,V,EPSIMP,IMPMAX,RO,U,EPS,NMAX,ROG)
      DIMENSION V(N1,N2),U(2,N1,N2)
C         Resolution de l equation de Kolmogorov dans le cas ou:
C               Les parametres sont
C               L etats-temps est:  X1 X2
C               La dynamique du systeme est decrite par l operateur
C                    2     2                         2     2
C         plus( Q2 X2  + X2  + 3 P2 X2 + Q1 X1  + X1  + 3 P1 X1 - X1 + Q2
C                                                2           2
C                    + Q1 + 0.25 , Minu( U2  + P2 U2 + U1  + P1 U1 ) )
C               ou v(..) et w designe  le cout
C               ou pi designe sa derivee premiere par rapport a xi
C               ou qi designe sa derivee seconde par rapport a xi
C               Le probleme est statique
C               Les conditions aux limites sont:
C                    X2 = 0    V = 0
C                    X2 = 1    V = 0
C                    X1 = 0    -p1 = 0
C                    X1 = 1    p1 = 0
C         Les nombres de points de discretisation sont: N1 N2
C                    X2 = 1 correspond a I2 = N2
C                    X2 = 0 correspond a I2 = 1
C                    X1 = 1 correspond a I1 = N1 - 1
C                    X1 = 0 correspond a I1 = 2
C         Le taux d actualisation vaut: 3
C         impmax designe le nbre maxi d iterations du systeme implicite
C         epsimp designe l erreur de convergence du systeme implicite
C         ro designe le pas de la resolution du systeme implicite
C                                   par une methode iterative
C         P2 est discretise par difference divise  symetrique
C         P1 est discretise par difference divise  symetrique
C         Minimisation par la methode de  gradient avec projection
C                                                    de l'Hamiltonien:
C
C                2          2
C               U2  + P2 U2 + U1  + P1 U1
C         contraintes sur le controle:
C               - 1 =< U2 =< 1
C               - 1 =< U1 =< 1
C         nmax designe le nombre maxi d iteration de la methode de
C                                              gradient avec projection
C         eps designe l erreur de convergence de la methode de
C                                              gradient avec projection
C         rog designe le pas, qui est constant, dans la methode de gradi#
C        ent
      H2 = 0.999999/(N2-1)
      H1 = 0.999999/(N1-3)
      U2 = U(2,1,1)
      U1 = U(1,1,1)
      HIH2 = H2**2
      HIH1 = H1**2
      H22 = 2*H2
      H21 = 2*H1
      NM2 = N2-1
      NM1 = N1-1
      do 1019 I2=1,N2,1
      do 1019 I1=1,N1,1
      V(I1,I2) = 0.0
```

```
1019 CONTINUE
     IMITER = 1
1013 CONTINUE
     ERIMP = 0
     do 1011 I1=1,N1,1
     X1 = H1*(I1-2)
     V(I1,N2) = 0
     V(I1,1) = 0
1011 CONTINUE
     do 1009 I2=2,NM2,1
     X2 = H2*(I2-1)
     V(N1,I2) = V(N1-2,I2)
     V(1,I2) = V(3,I2)
1010 CONTINUE
     do 1009 I1=2,NM1,1
     X1 = H1*(I1-2)
     Q2 = (V(I1,I2+1)-2*V(I1,I2)+V(I1,I2-1))/HIH2
     Q1 = (V(I1+1,I2)-2*V(I1,I2)+V(I1-1,I2))/HIH1
     P2 = (V(I1,I2+1)-V(I1,I2-1))/H22
     P1 = (V(I1+1,I2)-V(I1-1,I2))/H21
     W = V(I1,I2)
     NITER = 0
     W0 = -1.0e20
1001 CONTINUE
     NITER = NITER+1
     if (NITER-NMAX) 1002,1002,1003
1003 CONTINUE
     WRITE(8,1801)I1,I2
1801  FORMAT(' descente n a pas converge', 2 i3)
     GOTO 1004
1002 CONTINUE
     UN1 = (1-2*ROG)*U1-P1*ROG
     UN2 = (1-2*ROG)*U2-P2*ROG
     U1 = UN1
     U2 = UN2
     U1 = AMAX1(U1,-1)
     U1 = AMIN1(U1,1)
     U2 = AMAX1(U2,-1)
     U2 = AMIN1(U2,1)
     WW = U2**2+P2*U2+U1**2+P1*U1
     ER = ABS(WW-W0)
     if (ER-EPS) 1004,1004,1005
1005 CONTINUE
     W0 = WW
     GOTO 1001
1004 CONTINUE
     U(1,I1,I2) = U1
     U(2,I1,I2) = U2
     W0 = WW
     W1 = Q2*X2**2+X2**2+3*P2*X2+Q1*X1**2+X1**2+3*P1*X1-X1+Q2+Q1+0.25
     W0 = W1+W0
     W0 = W0-3*V(I1,I2)
     VNEW = RO*W0+V(I1,I2)
     V(I1,I2) = VNEW
     ERIMP = ABS(W0)+ERIMP
1009 CONTINUE
     IMITER = IMITER+1
     if (IMITER-IMPMAX) 1016,1015,1015
1016 CONTINUE
     if (EPSIMP-ERIMP) 1013,1012,1012
```

```
10.15 CONTINUE
      WRITE(8,1807)
1807   FORMAT(' schema implicite n a pas converge')
1012 CONTINUE
      do 1017 I1=1,N1,1
      do 1017 I2=1,N2,1
      WRITE(8,1800)I1,I2,V(I1,I2)
1800   FORMAT(' FTNV[', (i3,','), i3,']:', e14.7,'$')
      WRITE(8,1901)I1,I2,U(1,I1,I2)
1901   FORMAT(' FTNU1[', (i3,','), i3,']:', e14.7,'$')
      WRITE(8,1902)I1,I2,U(2,I1,I2)
1902   FORMAT(' FTNU2[', (i3,','), i3,']:', e14.7,'$')
1017 CONTINUE
      RETURN
      END
```

REFERENCES

[1] BENSOUSSAN,A.-LIONS,J.L.(1978).Applications des inequations variationnelles en controle stochastique.Dunod.

[2] FLEMING,W.H.-RISHEL,R.(1975).Deterministic and Stochastic Optimal Control. Springer Verlag,New York.

[3] GOURSAT,M.-QUADRAT,J.P.(1975).Analyse numeriques d'inequations quasi variationnelles elliptiques associees a de problemes de controle impulsionnel.IRIA Report.

[4] KUSHNER,H.J.(1977).Probabilty methods in stochastic control and for elliptic equations. Academic Press

[5] QUADRAT,J.P.(mars 1980).Existence de solution et algorithme de resolutions numeriques de problemes stochastiques degeneres ou non.SIAM Journal of Control.

[6] QUADRAT,J.P.(1975).Analyse numerique de l'equation de Bellman stochastique.IRIA Report.

[7] STROOCK,F.-VARADHAN,S.R.S.(1979).Multidimensional Diffusion Process. Springer Verlag.

REFERENCES

[1]**J.P.AUBIN,** Approximation of elliptic Boundary-value Problems. Wiley Inter-science, New-York, 1972.

[2]**F.BASKET,M.CHANDY,R.MUNTZ,J.PALACIOS** Open Closed Mixed Network Of Queus With Different Class of Customers, J.A.C.M n°22, p. 248-260, 1975.

[3]**R.BELLMAN,** Dynamic Programming,Princeton University Press,1957.

[4]**A.BENSOUSSAN,** Stochastic Control by Functional Analysis Method, North-Holland 1982.

[5]**A.BENSOUSSAN,** Méthodes de Perturbartion en Contrôle Optimal,1985.

[6]**A.BENSOUSSAN,J.L.LIONS,** Applications des inéquations variationnelles en contrôle stochastique, Dunod,1978.

[7]**P.F.CIARLET,** The finite element method for elliptic problems, North Holland, 1978.

[8]**D.CLAUDE,** Decoupling of Nonlinear Systems, Systems Control Letters n°1, p242-248, 1982.

[9]**A.etC.COLMERAUER,** Prolog en 10 figures, La Recherche 1985.

[10]**J.B.CRUZ,** Feedback Systems, McGraw-Hill, 1972.

[11]**F.DELEBECQUE,J.P.QUADRAT,**

-Sur l'Estimation des Caractéristiques Locales d'un Processus de diffusion avec Sauts, Rapport INRIA 1978.

-Contribution of Stochastic Control Singular Perturbation Averaging and Team Theories to an Example of Large-Scale System: Management of Hydropower Production, IEEE AC23 n°2, p209-221 1978.

[12]**J.C.DODU,M.GOURSAT,A.HERTZ,J.P.QUADRAT,M.VIOT** Méthodes de gradient stochastique pour l'optimisation des investissements dans un réseau électrique, EDF Bulletin Serie C, n°2, 1981

[13]**I.EKLAND,R.TEMAM,** Analyse convexe et problèmes variationnels, Dunod, 1974.

[14]**W.H.FLEMING,** Control for small noise intensities, SIAM J.Control, vol.9, n°3, 1971.

[15]**W.H.FLEMING-R.RISHEL,** Optimal Deterministic and Stochastic Control, Springer-Verlag, 1975.

[16]**F.GEROMEL,J.LEVINE,P.WILLIS,** A fast Algorithm for Systems Decoupling using Formal Calculus, L.N.C.I.S n°63, Springer-Verlag 1984.

[17]**P.GLOESS,** Logis User's manual, Université de Compiegne, janvier 1984.

[18]**P.GLOESS,** Understanding Expert Systems,Université de Compiegne, janvier 1984.

[19]**C.GOMEZ,J.P.QUADRAT,A.SULEM,** Towards an Expert System in Stochastic Control : the Hamilton-Jacobi equation Part, L.N.C.I.S n°63, Springer-Verlag 1984.

[20]**C.GOMEZ,J.P.QUADRAT,A.SULEM,** Towards an Expert System in Stochastic Control : the Local-feedback Part, Congrés Rome sur le contrôle Stochastique, L.N.C.I.S Springer-Verlag 1985.

[21]**THEOSYS** Numerical methods in stochastic control., RAIRO Automatique 1983.

[22]**M.GOURSAT,J.P.QUADRAT,**

- Analyse numériques d'inéquations quasi variationnelles elliptiques associées à des problèmes de contrôle impulsionnel, IRIA Rapport,1975.
- Analyse numériques d'inéquations variationnelles elliptiques associées à des problèmes de temps d'arrêt optimaux, IRIA Rapport,1975.

[23]**J.JACOD,** Calcul Stochastique et Problèmes de Martingale, Springer-Verlag, 1979.

[24]**E.KIEFER,J.WOLFOWITZ,** Stochastic estimation of the maximum of a regression function, Ann. Math. Statistic, 23,n°3,1952.

[25]**H.J.KUSHNER,** Probabilty methods in stochastic control and for elliptic equations, Academic Press, 1977.

[26]**H.J.KUSHNER,D.S.CLARK,** Stochastic approximation methods for constrained and unconstrained systems,Springer Verlag,1978.

[27]**P.L.LIONS,B.MERCIER,** Approximation numérique des équations de Jacobi-Bellman, RAIRO,14 ,pp.369-393, 1980.

[28]**P.LIPCER-A.SHIRIAEV** Statistique des Processus Stochastiques, Presse Universitaire

Moscou, 1974.
[29]**LISP** Machine Lisp Manual, MIT Press,1982.
[30]**MACSYMA** Manual,MIT Press,1983.
[31]**B.T.POLYAK,** Convergence and convergence rate of iterative stochastic algorithms.Automatica i Telemekhanika, Vol.12,1976,pp.83-94.
[32]**B.T.POLYAK,** Subgradient methods a survey of soviet research in nonsmooth optimization, C.Lemarechal and K.Mifflin eds.Pergamon Press, 1978.
[33]**B.T.POLYAK,Y.Z.TSYPKIN,** Pseudogradient adaptation and training algorithms.Automatica i Telemekhanika,Vol.3,1973.
[34]**J.P.QUADRAT,** Existence de solution et algorithme de résolutions numériques de problèmes stochastiques dégénérés ou non, SIAM Journal of Control, mars 1980.
[35]**J.P.QUADRAT,** Analyse numérique de l'équation de Bellman stochastique, IRIA Report,1975.
[36]**J.P.QUADRAT,** On optimal stochastic control problem of large systems, Advances in filtering and optimal stochastic control, Lecture Notes in Control and Computer Science n° 42, Springer-Verlag,1982.
[37]**J.P.QUADRAT,M.VIOT,** Product form and optimal local feedback for multi-index Markov chains, Allerton Conference,1980.
[38]**C.QUEINNEC,** LISP Langage d'un autre type, Eyrolles, 1983.
[39]**H.ROBBINS,S.MONRO,** A stochastic approximation method. Ann. Math. Statist., 22, pp400-407, 1951.
[40]**F.STROOCK,S.R.S.VARADHAN,** Multidimensional Diffusion Processes, Springer-Verlag,1979.
[41]**TORRION,** Differentes méthodes d'optimisation appliquées à la gestion annuelle du système offre-demande français. Note EDF, EEG 1985.
[42]**TURGEON,** Optimal operation of multi-reservoir power system with stochastic inflows.Water resource resarch. April 1980
[43]**CHANCELIER-GOMEZ-QUADRAT-SULEM** Un systeme expert pour l'optimisation de système dynamique, Congres d'Analyse Numerique INRIA Versailles Decembre 85, North Holland.