PHYSICAL REVIEW E **77**, 035101(R) (2008)

Role of optimization in the human dynamics of task execution

Daniel O. Cajueiro and Wilfredo L. Maldonado
Department of Economics, Catholic University of Brasilia, 70790-160, Brasilia, DF, Brazil
(Received 12 June 2007; published 21 March 2008)

In order to explain the empirical evidence that the dynamics of human activity may not be well modeled by Poisson processes, a model based on queuing processes was built in the literature [A. L. Barabasi, Nature (London) **435**, 207 (2005)]. The main assumption behind that model is that people execute their tasks based on a protocol that first executes the high priority item. In this context, the purpose of this paper is to analyze the validity of that hypothesis assuming that people are rational agents that make their decisions in order to minimize the cost of keeping nonexecuted tasks on the list. Therefore, we build and analytically solve a dynamic programming model with two priority types of tasks and show that the validity of this hypothesis depends strongly on the structure of the instantaneous costs that a person has to face if a given task is kept on the list for more than one period. Moreover, one interesting finding is that in one of the situations the protocol used to execute the tasks generates complex one-dimensional dynamics.

DOI: 10.1103/PhysRevE.77.035101 PACS number(s): 89.65.Ef, 02.50.Le, 02.50.Ga

I. INTRODUCTION

Empirical evidence has shown that the dynamics of interevent times driven by human actions may not be well described by Poisson processes [1–3]. Based on this, Barabási [4] developed a very interesting model of human activity where the distributions of the interevent times are consequences of the decision queue process. He considers that among the most relevant protocols for driving human dynamics, e.g., first-in-first-out protocol, random protocol, and a protocol based on the execution of the high priority item, the latter protocol seems to be the most important. In this protocol, while high priority tasks are executed as soon as they are added to the list, low priority tasks wait for a long time until all high priority tasks are executed, i.e., the instants of execution of low priority tasks are separated by long times of inactivity. Using this assumption, it is shown numerically in Ref. [4] and analytically in Ref. [5] that the distribution of interevent times follows a power law.

Two interesting contributions were introduced by Ref. [6]. First the authors map the variable queue length priority model considered above onto a model of biased diffusion deriving asymptotic distributions for the interevent times. Second, in order to investigate the arising of power laws in more general situations, they generalize the fixed queue length model to include tasks with a priority label and with a class label where there is always an active class and an inactive class. If the highest priority task of the inactive class exceeds that of the active class by at least a fixed switching cost, the inactive class becomes active and the active class becomes inactive.

An interesting discussion is considered in Ref. [7,8] where it is argued that other mechanisms contribute to the distributions of waiting times such as deadlines, time dependence of priorities, and the social context of the problem. In line with this debate, Ref. [9] relaxes the assumption that the priorities of tasks do not change over time and studies queueing systems where deadlines are assigned to the incoming tasks and the urgency to attend a task increases with time showing that only in the former model fat tails arise naturally as consequence of the scheduling rule.

In this paper, we investigate the assumption that people execute tasks following a protocol that first executes the high priority item. In particular, we suppose that people assign priorities to the tasks on their lists in order to minimize some cost index, i.e., a cost associated to keep a nonprocessed collection of tasks in a current period. Therefore, based on this assumption and inspired by Refs. [4–6], we have built a discounted stochastic dynamic programming model with two types of tasks (low and high priority tasks) and a cost per stage for keeping a number of low and high priority tasks without processing.

This is not the first time that a kind of optimization principle is used to understand the structure and dynamics of complex systems. In Refs. [10,11], for instance, it was shown that complex networks may arise from optimization principles. It is also important to stress that although there is a lot in the literature dealing with control of queue discipline [12–14] which this work is related to, the model presented in this paper is neither an extension nor a particular case of any of these results.

We have found that the type of protocol used to execute tasks is strongly dependent on the kind of instantaneous cost of keeping a task in the queue for an additional stage. When linear costs are used the protocol of executing preferentially the high priority costs is always the best solution. However, this does not happen when quadratic costs are considered. In this case different types of protocol are considered. Furthermore, depending on the parameters of the system, the protocol considered generates complex one-dimensional dynamics.

II. SETUP OF THE PROBLEM

We consider that there are two queues waiting for a service on a single server. Let $g(x_L,x_H)$ be the current cost of having state (x_L,x_H) which is the state of the system, where x_L (x_H) is the number of tasks in the first (second) queue. We say that the first queue is a low priority queue (or the second queue is a high priority queue) if $\frac{\partial g(x_L,x_H)}{\partial x_L}\big|_{x_L=x_H} < \frac{\partial g(x_L,x_H)}{\partial x_H}\big|_{x_L=x_H}$. We assume that this is the case. The dynamics of these queues are modeled as follows: at each period there is a probability $\lambda\rho$ of a new task arrives in the queue formed by high priority tasks and a probability $\lambda(1-\rho)$ of a new task arrives in the queue formed by low priority tasks. Within each of the queues the tasks are executed on a first in, first out basis. With probability $\mu u(x_L,x_H)$ the first task of the high priority queue is executed and with probability $\mu[1-u(x_L,x_H)]$ the first task of the low priority queue is executed. We assume here that $u(x_L,x_H)$ is the state-dependent control variable that the agent will choose in order to minimize the total cost function $J_u(x_L,x_H) = E^u_{x_L,x_H}(\sum_{t=1}^{\infty}\alpha^t g[x_L(t),x_H(t)])$, where α is the discount factor and $E^u_{x_L,x_H}[\cdots]$ is the expected value conditioned to the current state (x_L,x_H) and to the state control variable $u(\cdots)$.

Due to the principle of optimality [15,16] and the Banach fixed point theorem, if the minimum cost function $J(x_L,x_H) = \min_{u(x_L,x_H)\in[0,1]} J_u(x_L,x_H)$ exists, it must be given by the unique solution of the Bellman equation, that may be written as

$$J(x_L,x_H) = F(x_L,x_H) + \min_{u(x_L,x_H)\in[0,1]} u(x_L,x_H)G(x_L,x_H), \quad (1)$$

where

$$\begin{aligned} F(x_L,x_H) = {} & g(x_L,x_H) + \lambda\rho(1-\mu)[\alpha J(x_L,x_H+1)] \\ & + \lambda(1-\rho)(1-\mu)[\alpha J(x_L+1,x_H)] \\ & + (1-\lambda)\mu[\alpha J(x_L-1,x_H)] \\ & + \rho\lambda\mu[\alpha J(x_L-1,x_H+1)] \\ & + (1-\rho)\lambda\mu[\alpha J(x_L,x_H)] \\ & + (1-\lambda)(1-\mu)[\alpha J(x_L,x_H)] \end{aligned} \quad (2)$$

and

$$\begin{aligned} G(x_L,x_H) = {} & (1-\lambda)\mu\{\alpha[J(x_L,x_H-1) - J(x_L-1,x_H)]\} \\ & + \rho\lambda\mu\{\alpha[J(x_L,x_H) - J(x_L-1,x_H+1)]\} \\ & + (1-\rho)\lambda\mu\{\alpha[J(x_L+1,x_H-1) - J(x_L,x_H)]\}. \end{aligned} \quad (3)$$

Since the optimization problem (1) is a linear programming problem, the optimal control $u(x_L,x_H)$ in each state (x_L,x_H) will depend explicitly on the signal of $G(x_L,x_H)$. If $G(x_L,x_H)>0$, then $u(x_L,x_H)=0$. If $G(x_L,x_H)<0$, then $u(x_L,x_H)=1$. Finally, if $G(x_L,x_H)=0$, $u(x_L,x_H)$ is a mixed strategy that may have any value in the interval [0,1]. It is quite intuitive this result. Indeed, one may note that the terms in square brackets defined in $G(x_L,x_H)$, Eq. (3), comprise the variations in the cost function due to changes in the states of the queue related to the execution of one of the tasks. Since the properties of the solution of $J(x_L,x_H)$ of the Bellman equation (1) are strongly dependent on choice of the cost per stage $g(x_L,x_H)$, in the next sections, two different choices for $g(x_L,x_H)$ are investigated.

III. LINEAR COSTS

In this section, we assume that $g(x_L,x_H)=h_L x_L + h_H x_H$, for $0<h_L<h_H$, i.e., the current cost of having one additional high priority task in the queue is larger than having one additional low priority task in the queue.

Since the space of polynomials of degree 1 with the sup norm is a Banach space, one can show inductively, making recursive iterations of the dynamic programming mapping, that $J(x_L,x_H)$ is also linear. Therefore, for $x_L>0$ and $x_H>0$, one may easily solve the Bellman equation (1) and show that the cost function is given by [19]

$$J(x_L,x_H) = c + c_L x_L + c_H x_H. \quad (4)$$

Furthermore,

$$G(x_L,x_H) = \mu\frac{\alpha}{1-\alpha}(h_L - h_H) \quad (5)$$

is always negative implying that $u(x_L,x_H)=u=1$ for every state (x_L,x_H). Therefore, if linear costs are considered, the protocol to be considered is the one based on the execution of the high priority task whenever there is at least one item in this queue, i.e., $x_H>0$. This kind of protocol was very well studied in Refs. [4–6] where analytic results for the emerging of power laws may be found. In the next section, a more interesting situation arises where the optimal policy is not only limited to execute the high priority item in the queue, but the optimal policy is state dependent.

IV. QUADRATIC COSTS

Now, we assume that $g(x_L,x_H)=h_L x_L^2 + h_H x_H^2$, for $0<h_L<h_H$. Following the same reasoning already presented before for the linear cost case, one may conclude a quadratic form for the cost function.

Solving the Bellman equation, one may show that the solution of the problem depends explicitly on the signal of the function $G(x_L,x_H)$, defined in Eq. (3), in the state (x_L,x_H). In fact, three different regions will arise. We will call region A the domain of (x_L,x_H) where $G(x_L,x_H)>0$, region B the domain of (x_L,x_H) where $G(x_L,x_H)=0$ and region C the domain of (x_L,x_H) where $G(x_L,x_H)<0$.

We have found that the minimum cost function and the optimal control are given respectively by:

$$J(x_L,x_H) = \begin{cases} J^A(x_L,x_H) \text{ if } (x_L,x_H) \in A, \\ J^B(x_L,x_H) \text{ if } (x_L,x_H) \in B, \\ J^C(x_L,x_H) \text{ if } (x_L,x_H) \in C \end{cases} \quad (6)$$

$$u(x_L,x_H) = \begin{cases} 0 \text{ if } (x_L,x_H) \in A, \\ u \in [0,1] \text{ if } (x_L,x_H) \in B, \\ 1 \text{ if } (x_L,x_H) \in C, \end{cases} \quad (7)$$

where for $i=A,B,C$

$$J^i(x_L,x_H) = c^i + c^i_{L1}x_L + c^i_{H1}x_H + c_{L2}x_L^2 + c_{H2}x_H^2, \quad (8)$$

$$\begin{aligned} G^i(x_L,x_H) = {} & \mu\{c^i_{L1} - c^i_{H1} + 2(c_{L2}x_l - c_{H2}x_H) + c_{L2}[2\lambda(1-\rho) \\ & - 1] + c_{H2}(1-2\lambda\rho)\} \end{aligned} \quad (9)$$

and $G(x_L,x_H)=G^i(x_L,x_H)$, if $(x_L,x_H)\in i$ [20].

Furthermore, the parameter $\delta\in[\underline{\delta},\overline{\delta}]$, defines the set of points of $\Re^2$ such that $G[x_L,(h_L/h_H)x_L+\delta]=0$ [21]. Indeed,

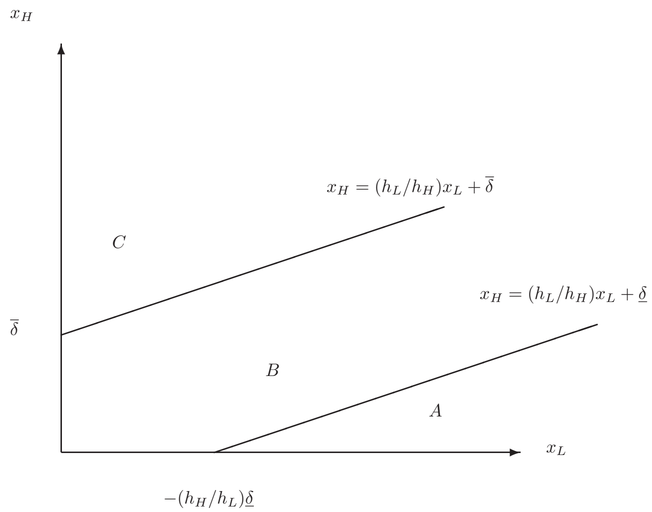

FIG. 1. The regions A, B, and C in the plane $x_L - x_H$.

$$\delta \to \underline{\delta} \Rightarrow [J^B(\delta) \to J^A, G^B(\delta) \to G^A] \tag{10}$$

and

$$\delta \to \bar{\delta} \Rightarrow [J^B(\delta) \to J^C, G^B(\delta) \to G^C]. \tag{11}$$

In order to understand the intuition behind this solution, one is invited to consider a particular case where $h_L/h_H \approx 1$, $\lambda = \mu = 1$ and $\rho = 1/2$. In this case, the region B is defined by the set of points where $|x_H - x_L| < \alpha/(1-\alpha)$. Therefore, the set of points where the decision maker can use any strategy depends strictly on the discount factor. If the discount factor is large, the decision maker may keep queues with a large difference between their sizes. On the other hand, if the discount factor is small this situation is not accepted as a solution anymore.

Differently from the linear costs case, several types of protocol are possible. Region C considers a protocol based on the execution of the high priority task. Region A considers a protocol based on the execution of the low priority task. It occurs in order to avoid that the size of the queue of the low priority tasks do not increase too much. "Too much" here is measured by the ratio h_L/h_H. In the region B there is not a unique protocol. It can be a random protocol (mixed strategy) or simply a protocol similar to that considered in region C or in region A. Figure 1 shows the geometry of these regions in the plane $x_L - x_H$.

It is not difficult to show that the expected value of the state obeys the following dynamics:

$$E_t[x(t+1)] = E_t\begin{bmatrix} x_L(t+1) \\ x_H(t+1) \end{bmatrix} = \begin{bmatrix} x_L(t) \\ x_H(t) \end{bmatrix} + \begin{bmatrix} \lambda(1-\rho) - \mu\{1 - u[x_L(t), x_H(t)]\} \\ \lambda\rho - \mu u[x_L(t), x_H(t)] \end{bmatrix} \tag{12}$$

which has infinite fixed points if and only if $\lambda = \mu$ and $u(x_L, x_H) = u = \rho$. We will analyze only the most interesting situation which is the fixed-length-queue, i.e., $\lambda = \mu$. Therefore, assuming that $\lambda = \mu$, $u^B = \rho + \epsilon$ and $\epsilon > 0$, then the expected value of the system is governed by $E_t[x(t+1)] = x(t) + \lambda\epsilon e$ if it is in region B and by $E_t[x(t+1)] = x(t) - \lambda\rho e$ if it is in region A (if the state is in region C, the expected state will certainly come to region B and will not come back to this region), where $e = (1, -1)'$. Therefore, the dynamics takes place in the line passing by $x(0)$ and following the direction e. Thus, if the expected state is in region B it goes into the direction of region A and vice versa. This dynamics is equivalent to the one-dimensional system

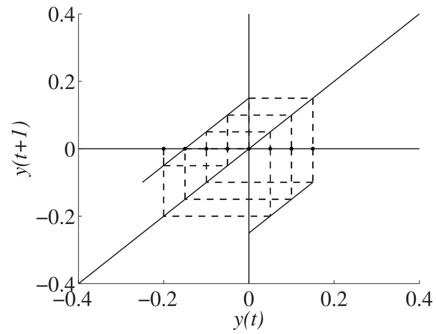

FIG. 2. The evolution of $y(t)$ for $y_0 = -0.2$, $\lambda = 0.5$, $\rho = 0.5$, and $\epsilon = 0.3$.

$$y(t+1) = \begin{cases} y(t) + t^+ & \text{if } y(t) \leq 0, \\ y(t) - t^- & \text{if } y(t) > 0 \end{cases} \tag{13}$$

defined on the interval $(-t^-, t^+]$, where $t^+ = \lambda\epsilon$ and $t^- = \lambda\rho$.

The dynamics of this system is plotted in Fig. 2 for the case of $t^- = 0.25$ and $t^+ = 0.15$. The dynamics defined in Eq. (13) is topologically conjugate with the circle rotation [17]. Therefore, if $t^+/t^- = p/q$, where p/q is a irreducible ratio representation of rational number, then this system follows a limit cycle with period $p+q$. Otherwise, the ω-limit of any point in the interval is a dense subset of it. Therefore, we can conclude that the stochastic process that defines the length of each queue is not stationary. Moreover, the dynamics of the expected value of the length of the queue exhibits a complex behavior: cycles of any order or a limit set being a dense subset in the interval. The intuition behind that complex dynamics is quite reasonable. In the region close to the frontier $x_H = (h_L/h_H)x_L + \underline{\delta}$ that separates A and B (see Fig. 1), we can observe the following: if the expected state is in A, its dynamics moves toward region B, since the priority is of L. Once the expected state is in B, the dynamics takes it back to the region A, since in this case in average the priority is of H (due to the condition $u^B = \rho + \epsilon$ and $\epsilon > 0$). Because the frequency of tasks arriving is equal to that of attending them, a cyclical or complex dynamics emerges close to the referred frontier. Figure 2 shows the case where this system is a limit cycle. A similar situation involving regions B and C arises in the case of $\epsilon > 0$ and $\rho = u^B + \epsilon$. In these situations, the protocol is ruled by the protocols considered in regions A and B in the former case and by the protocols considered in regions B and C in the later case.

For $\lambda \neq \mu$, either the expected value goes to infinite, converges to 0, to axis $x_L = 0$ or to axis $x_H = 0$, following different routes. Furthermore, different kinds of protocols are possible.

V. FINAL REMARKS

In the human dynamics of the tasks execution decisions the priority of one task is not always defined as being the most important current task. Actually, the dynamics of the work executions depends on the cumulated tasks of short run priorities, the importance of each kind of task and the intertemporal discount factor. In this paper, we provide a stochastic dynamic programming model containing all those elements and analyze the dynamics of the execution of tasks, shedding new light on the discursion considered in Ref. [7,8]. In this setting, we have found that the dynamics of the expected state of the system may be complex, exhibiting cycles of any order or with limit set being a dense subset of the interval depending on the parameter values of the model. This is a contribution to a better understanding of how human dynamics may evolve in this type of problem. Finally, it is worth noting that complex dynamics in the solution of dynamic programming problems are usually obtained for low discount factors [18]. However, in our quadratic case, complex dynamics arises for discount factors of any size.

ACKNOWLEDGMENTS

D. O. Cajueiro is indebted to CNPQ for financial support through Grant Nos. 308494/2006-7 and 400655/2006-3. W. L. Maldonado would like to thank CNPq for financial support (Grant No. 305317/2003-2) and Edital Universal (Grant No. 471899/2003-8). D. O. Cajueiro is also indebted to Professor G. Grinstein who helped him to understand some details of Ref. [6].

[1] V. Paxson and S. Floyd, IEEE/ACM Trans. Netw. **3**, 226 (1996).

[2] J. Masoliver, M. Montero, and G. H. Weiss, Phys. Rev. E **67**, 021112 (2003).

[3] E. Scalas, T. Kaizoji, M. Kirchler, J. Huber, and A. Tedeschi, Physica A **366**, 463 (2006).

[4] A. L. Barabasi, Nature (London) **435**, 207 (2005).

[5] A. Vázquez, Phys. Rev. Lett. **95**, 248701 (2005).

[6] G. Grinstein and R. Linsker, Phys. Rev. Lett. **97**, 130201 (2006).

[7] A. Kentsis, Nature (London) **441**, 04901 (2005).

[8] A. L. Barabasi and J. G. Oliveira, Nature (London) **441**, 04902 (2006).

[9] P. Blanchard and M. O. Hongler, Phys. Rev. E **75**, 026102 (2007).

[10] I. Rodriguez-Iturbe, A. Rinaldo, R. Rigon, R. L. Bras, E. Ijjasz-Vasquez, and A. Marani, Geophys. Res. Lett. **19**, 889 (1992).

[11] D. O. Cajueiro, Phys. Rev. E **72**, 047104 (2005).

[12] T. B. Crabill, D. Gross, and M. J. Magazine, Oper. Res. **25**, 219 (1977).

[13] M. Y. Kitaev and V. V. Rykov, *Controlled Queueing Systems* (CRC Press, Boca Raton, 1995).

[14] L. I. Sennott, *Stochastic Dynamic Programming and the Control of Queueing Systems* (John Wiley and Sons, New York, 1999).

[15] R. Bellman, *Dynamic Programming* (Princeton University Press, New Jersey, 1957).

[16] D. P. Bertsekas, *Dynamic Programming and Optimal Control* (Athena Scientific, Belmont, 2001), Vol. II.

[17] W. DeMello and S. VanStrien, *One Dimensional Dynamics* (Springer, Berlin, 1993).

[18] L. Montrucchio and G. Sorger, Insur. Math. Econ. **25**(2), 181 (1996).

[19] The constants are given by $c_L=\frac{h_L}{1-\alpha}$, $c_H=\frac{h_H}{1-\alpha}$ and $c=\frac{\alpha}{(1-\alpha)^2}[\lambda(1-\rho)h_L+(\lambda\rho-\mu)h_H]$.

[20] The constants are given by $c_{L2}=\frac{h_L}{1-\alpha}$, $c_{H2}=\frac{h_H}{1-\alpha}$, $c_{L1}^A=\frac{2\alpha h_L[-\mu+\lambda(1-\rho)]}{(1-\alpha)^2}$, $c_{L1}^B(\delta)=\frac{1}{(h_L+h_H)}\{\frac{h_L^2}{(1-\alpha)}[1-2\lambda(1-\rho)]+\frac{h_L h_H}{(1-\alpha)^2}[2\alpha(\lambda-\mu)+(2\lambda\rho-1+2\delta)(1-\alpha)]\}$, $c_{L1}^C=\frac{2\alpha h_L\lambda(1-\rho)}{(1-\alpha)^2}$, $c_{H1}^A=\frac{2\alpha h_H\lambda\rho}{(1-\alpha)^2}$, $c_{H1}^B(\delta)=\frac{1}{(h_L+h_H)}\{\frac{h_H^2}{(1-\alpha)}[1-2\lambda\rho-2\delta]+\frac{h_L h_H}{(1-\alpha)^2}[2(\lambda-\mu\alpha)-(2\lambda\rho+1)(1-\alpha)]\}$, $c_{H1}^C=\frac{2\alpha h_H(-\mu+\lambda\rho)}{(1-\alpha)^2}$, $c^A=\frac{\alpha}{(1-\alpha)^3}((1-\alpha)[\mu+(1-\rho)\lambda]+2\alpha(\lambda^2+\mu^2-\mu\lambda)+2\rho\lambda[\mu(1+\alpha)+\lambda\alpha(\rho-2)]-2\mu\lambda)h_L+\lambda\rho[1-\alpha+2\lambda\rho\alpha]h_H$, $c^B=\frac{\alpha}{(h_L+h_H)(1-\alpha)^3}\{[2\lambda\alpha(1-\alpha)(-\rho^2\lambda+2\lambda\rho-1-\lambda-\rho)]h_L^2+[\delta(2\lambda\rho\alpha^2+4\lambda\alpha-4\lambda\rho\alpha+1-2\lambda-2\alpha+\alpha^2-2\lambda\alpha^2+2\lambda\rho)+2\lambda(-\mu\alpha+2\lambda\rho^2\alpha^2+2\lambda\rho\alpha-2\lambda\rho^2\alpha-\alpha^2\mu-2\lambda\alpha^2\rho+\lambda\alpha^2)+2\mu(\alpha-\alpha^2+\mu\alpha^2)]h_L h_H+[\delta(-4\lambda\rho\alpha+2\delta-\alpha^2+2\delta\alpha^2+2\lambda\rho\alpha^2-1-4\alpha\delta+2\alpha+2\lambda\rho)+2\lambda\rho(\lambda\rho\alpha^2-\lambda\rho\alpha-\alpha^2+\alpha)]h_H^2\}$, and $c^C=\frac{\alpha}{(1-\alpha)^3}\{[\lambda(1-\alpha)(1-\rho)+2\lambda^2\alpha(1-\rho)^2]h_L+[(1-\alpha)(\mu+\lambda\rho)+2\mu^2\alpha+2\lambda\rho(\lambda\rho\alpha-\mu\alpha-\mu)]h_H\}$.

[21] The constants $\underline{\delta}$ and $\overline{\delta}$ are, respectively, given by $\underline{\delta}=\frac{1}{1-\alpha}\{[\frac{1-\alpha}{2}-\lambda\rho]+\frac{h_L}{h_H}[-\frac{1-\alpha}{2}+\lambda(1-\rho)-\alpha\mu]\}$ and $\overline{\delta}=\frac{1}{1-\alpha}\{[\frac{1-\alpha}{2}-\lambda\rho+\alpha\mu]+\frac{h_L}{h_H}[-\frac{1-\alpha}{2}+\lambda(1-\rho)]\}$.