Author manuscript, published in "IS'10 : 5th IEEE International Conference on Intelligent Systems, London : United Kingdom (2010)"

hal-00565838, version 1 - 14 Feb 2011

Multicriteria Reinforcement Learning Based on a Russian Doll Method for Network Routing

Alain Petrowski[*], Farouk Aissanou[†], Ilham Benyahia[‡], Sébastien Houcke[§]

[*]Dept. RS2M, TELECOM Sud Paris, 91011 Evry Cedex, France, Alain.Petrowski@telecom-sudparis.eu
[†]Dept. RS2M, TELECOM Sud Paris, 91011 Evry Cedex, France, Farouk.Aissanou@telecom-sudparis.eu
[†]Université du Québec en Outaouais, Gatineau, (Québec) Canada J8X 3X7
[‡]Université du Québec en Outaouais, Gatineau, (Québec) Canada J8X 3X7, Ilham.Benyahia@uqo.ca
[§]Dept. SC, TELECOM-Bretagne, 29238 Brest Cedex 3, France, Sebastien.Houcke@telecom-bretagne.eu

***Abstract*—The routing in communication networks is typically a multicriteria decision making (MCDM) problem. However, setting the parameters of most used MCDM methods to fit the preferences of a decision maker is often a difficult task. A Russian doll method able to choose the best multicriteria solution according to a context defined beforehand is proposed. This context is given by a set of nested boxes in the criteria space, the shapes of which can be established from objective facts such as technical standards, technical specifications, etc. This kind of method is well suited for self-adaptive systems because it is designed to be able to give pertinent results without interaction with a decision maker, whatever the Pareto front.**

The Russian doll multicriteria decision method is used with a reinforcement learning to optimize the routing in a mobile ad-hoc network. The results on a case study show that the routing can be finely controlled because of the possibility to include as much parameters as desired to adjust the search of the best solution on Pareto fronts a priori unknown. These results are clearly better than those obtained with the optimization of a weighted sum or the minimization of a Chebyshev distance to a reference point.

I. Introduction

A. Background on Routing Protocols for Mobile Ad-Hoc Networks

Mobile ad hoc networks (MANETs) are self adaptive telecommunication networks formed by several wireless nodes that attempt to build an efficient topology in order to communicate. The aim of the routing task is to find a path between a source node and a destination node. As the transmission range of the nodes is limited, the path often uses successive forwarding nodes called hops. The routing should optimize Quality of Service (QoS) parameters like hop count, delay or bitrates. In addition, the mobility of nodes causes frequent changes of the network topology, thus the routing mechanism must be able to modify paths in order to adapt to changes. Many works have been published on the field of routing in MANET [1].

The most often, the protocols described in [1] deal with a single performance parameter namely hop count. Nevertheless, many applications like video and voice require different performance parameters; therefore a new range of protocols is needed.

Recently, some protocols based on reinforcement learning methods [2] have emerged in order to deal with QoS optimization in MANET. These protocols are either based on Q-learning [3] like LQ-Routing [4], or on “Ant Colony Optimization”, which is a special case of reinforcement learning [5].

B. Routing as a Multi-criteria decision problem

The performance based routing problem is typically a multi-criteria decision making problem (MCDM). It should be solved in such a way that several QoS criteria are taken into account simultaneously. However, these criteria are often contradictory. Moreover, the importance given to the different criteria depends on the requirements of the applications using the network. For example, a telephony application could accept quite high error rates, but the maximum transmission delay should be of the order of 0.1 second. On the contrary, a file transfer application will be efficient if the error rate is low, while the delay could be quite high, of the order of several seconds.

C. Related works

Especially for wireless networks, the task of selecting among many candidates the multicriteria solution fitting at best the preferences of a decision maker is discussed in many research works. Common approaches use parametric preference functions where the parameters, designated as “weights”, have to be adjusted to express the relative importance given to each criterion. Often these approaches transform the MCDM problem into a single-objective optimization problem. However, it is difficult to determinate good values for the weights according to the preferences of the decision maker, especially when criteria are incommensurable. Thus, in [10], the authors minimize a weighted sum in order to optimize the delay and the distance from a source node to a destination node, to deal with the issue of greedy-forwarding mechanism in ad hoc networks. The values of the two weights are fixed using multiple simulation scenarios. In [9] the authors formulate the routing task in a mobile ad hoc network as a MCDM problem that they solve by the weighted sum method. However, no idea is suggested for the setting of the weight values. The authors study the sensitivity of the optimization process results according to these weights. They claim that a good estimation

hal-00565838, version 1 - 14 Feb 2011

of the weight values is very important to obtain a good efficiency and depend only on the application classes. In [11], the authors consider the network selection problem as a MCDM problem. The aim is to optimize two contradictory objectives namely cost (price) and transmission time when a set of components (data, video, voice, etc.) is transmitted through different network connections. The problem is solved by minimizing the weighted Chebyshev distance between the solution and a reference point. Again, the authors state that the weight values have to be chosen by the user himself. However they argue that the proposed method is not very sensitive to them thanks to their choice of a specific reference point.

In [8], the authors solve the problem of multicast routing with a multi objective genetic algorithm (MOGA) [6], which tries to approximate the set of non-dominated solutions. The aim is to construct a multicast tree that optimizes several objectives like delay and bandwidth. The choice of the best solution among the set of non-dominated solutions is performed by a decision maker according to the requirement of the application. [7] use a reference solution given by the decision maker to guide the search process of the best multicriteria solutions during the selection step of a MOGA. But, an adequate choice of the reference point is strongly dependent on the optima locations, which are a priori unknown.

D. *Multicriteria routing for autonomous networks*

Our work is in the line of autonomic networks [12], which have a self-adaptive capacity. Such networks should be able to learn from their environment, to configure themselves and they should take intelligent decisions without external help. In this context, it is unacceptable that human decision makers help the network to choose best multicriteria routing decisions.

However, the standard MCDM approaches based on parametric preference functions need an a priori knowledge of the location of optima to give pertinent results. But, this knowledge is a priori not available. When results are disappointing, interactions with a decision maker are required for choosing new parameters. For this reason, we think that standard MCDM approaches are quite adequate for human machine interactions, but not for autonomous systems. In the frame of self-adaptive machines, a more suitable way could be to consider multicriteria decision problems in terms of homeostasis. With this point of view, the importance of criteria changes according to their value in order to regulate them.

Section II presents a Russian doll method to implement multicriteria decision making. Section III shows how the Russian doll method is used by a reinforcement learning method able to optimize routing in a mobile ad-hoc network. Section IV describes a case study involving an ad-hoc network of boats. Results of experiments are then given and discussed.

II. A Multicriteria Decision Making Method for autonomous systems

A. *Pareto dominance*

The criteria are represented by vectors defined in a criteria space $\mathcal{C} \subset \mathbf{R}^m$, for which each coordinate should be minimized. A vector of criteria $\mathbf{F}(\mathbf{x})$ is associated with each vector $\mathbf{x}$ in search space $\Omega \subset \mathbf{R}^n$.

The Pareto dominance is defined as follows: Let $\mathbf{a} = (a_1, ..., a_m)$ and $\mathbf{b} = (b_1, ..., b_m)$ be criteria vectors, $\mathbf{a}$ dominates $\mathbf{b}$ if and only if

$$\forall i \in [1, m], a_i \leq b_i \text{ and } \exists j \in [1, m] \text{ such that } a_j < b_j$$

Let $\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_m(\mathbf{x}))$ be the objective functions of the multi-criteria optimization problem. The resolution of this problem consists in finding the Pareto optimal set of vectors $\mathbf{X}^*$ such that each criteria vector $\mathbf{z} = \mathbf{F}(\mathbf{x})$ with $\mathbf{x} \in \mathbf{X}^*$ is non dominated. The Pareto front $\mathcal{F}$ is defined as the set $\mathcal{F} = \{\mathbf{F}(\mathbf{x}), \forall \mathbf{x} \in \mathbf{X}^*\}$.

Multi-criteria decision consists in choosing a Pareto optimal point that is recognized as the best choice by a decision maker.

B. *Decision making by autonomous systems*

We propose in this paper a MCDM approach suitable for autonomous systems where the parameters of the multicriteria decision are directly deduced from the requirements we want to reach. Typically, this kind of information could come from technical standards, technical specifications of system components or experiments.

For example, for a high quality Voice over IP (VoIP) application, the delay d should be lower than 0.1 s and the bit error rate BER should be lower than 10^{-6}. The problem is finding routes maximizing the QoS. Table I gives as an example a series of thresholds associated to different levels of quality.

TABLE I
Different levels of quality for a VoIP application

	Delay d	BER
High quality	[0, 0.1)	$[0, 10^{-6})$
Degraded quality	[0.1, 0.5)	$[10^{-6}, 10^{-4})$
Unacceptable	≥ 0.5	$\geq 10^{-4}$

Figure 1 can be used to illustrate an example of Pareto front in the criteria space generated by the BER (in abscissa) and the delay. Please note that this figure is only given to support explanations and does not come from experiments. Box 0 is defined as the whole criteria domain. According to Table I, box 2 contains high quality VoIP communications criteria vectors while vectors of box 1 that are not in box 2 are associated to degraded communications. Criteria vectors of box 0 but not in box 1 are in the "unacceptable" domain.

Figure 1 shows that box 2 does not contain any solution of Pareto front $\mathcal{F}$. So there are only "degraded quality" solutions at best for this Pareto front. The best solution chosen is the closest to box 2 (high quality domain box) according to a distance to define.

This example has given an idea about the working of the proposed MCDM algorithm. It is described in details in the following.

hal-00565838, version 1 - 14 Feb 2011

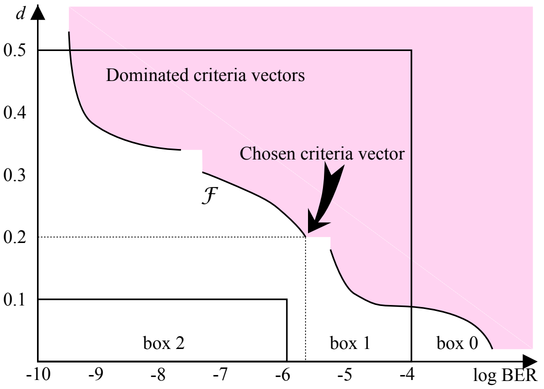

Fig. 1. Choice of a non dominated criteria vector with the help of nested boxes

C. Russian doll decision making method

Without loss of generality, we consider the case of a minimization problem for all the criteria. A box is defined by the Cartesian product of m intervals $[l_i, u_i]$ with $i \in [1, m]$. A set of $k+1$ boxes $B_j = \prod[l_{i0}, u_{ij}]$ indexed from 0 to k must be defined in the criteria space in such a way that any box, except box 0, is nested in another box like Russian dolls. Box 0 is the whole criteria domain. The parameters of a box are given by bounding values or threshold values for each criterion. The lower bounds l_{i0} of boxes are the lower bounds of the whole criteria domain.The required number of boxes and their shapes are specification-dependent. Many nested boxes allow the system designer to accurately control the choice of a solution on the Pareto front.

In the following, we need to define a distance between vectors and boxes. We propose to use the weighted Chebyshev distance $d^\infty_{\mathbf{w}}$ to ensure that any point of the Pareto front could be reached [13] by defining appropriate shapes for boxes. Let $\mathbf{a}$ and $\mathbf{b}$ be two vectors of $\mathbf{R}^m$.

$$d^\infty_{\mathbf{w}}(\mathbf{a}, \mathbf{b}) = \max_{i\in[1,m]} w_i |a_i - b_i| \text{ with } w_i \geq 0$$

Now, we define the normalized Chebyshev distance between a criteria vector $\mathbf{x} \in \mathbf{B}_{j-1}$ and box $\mathbf{B}_j$ as follows:

$$d^\infty(\mathbf{x}) = \max_{i\in[1,m]} \delta_{ij}, \text{ with } \delta_{ij} = \begin{cases} 0 & \text{if } x_i \in [l_{ij}, u_{ij}] \\ 0 & \text{if } u_{ij} = u_{ij-1} \\ \dfrac{x_i - u_{ij}}{u_{ij-1} - u_{ij}} & \text{otherwise} \end{cases}$$

In this way, every point on the upper borders u_{ij-1} of a box $\mathbf{B}_{j-1}$ is at the same distance from the nested box B_j except if $u_{ij} = u_{ij-1}$.

Let $\mathbf{u}$ and $\mathbf{v}$ be two criteria vectors to compare. $\mathbf{B}(\mathbf{u})$ and $\mathbf{B}(\mathbf{v})$ are defined as the smallest boxes containing vectors $\mathbf{u}$ and $\mathbf{v}$ respectively. Let $d^\infty(\mathbf{u})$ be the Chebyshev distance between $\mathbf{u}$ and the largest box contained in $\mathbf{B}(\mathbf{u})$. If $\mathbf{u}$ is contained by the smallest box $\mathbf{B}_k$, $d(\mathbf{u})$ is not defined.

- Let $i(\mathbf{B})$ be the index of box $\mathbf{B}$. If $i(\mathbf{B}(\mathbf{u})) > i(\mathbf{B}(\mathbf{v}))$, then $\mathbf{u}$ is said "better than" $\mathbf{v}$.
- If $\mathbf{B}(\mathbf{u}) = \mathbf{B}(\mathbf{v})$ and $i(\mathbf{B}(\mathbf{u})) \neq k$ and $d^\infty(\mathbf{u}) < d^\infty(\mathbf{v})$, then $\mathbf{u}$ is said "better than" $\mathbf{v}$.
- If $\mathbf{B}(\mathbf{u}) = \mathbf{B}(\mathbf{v})$ and $i(\mathbf{B}(\mathbf{u})) \neq k$ and $d^\infty(\mathbf{u}) = d^\infty(\mathbf{v})$ and $\mathbf{u}$ dominates $\mathbf{v}$, then $\mathbf{u}$ is said "better than" $\mathbf{v}$.
- In the other cases, $\mathbf{u}$ and $\mathbf{v}$ are said "indifferent".

By applying these simple rules, it is easy to show that the relation "is better than or indifferent to" is reflexive and transitive. However, this relation is not antisymmetric because of the "indifferent" rule and it is not symmetric because of the "better than" rules. Moreover, this relation is total: all couples of criteria vectors can be compared.

The Russian doll decision making method is implemented by comparing couples of criteria vectors with the comparison rules described above. The best criteria vector is the one that is the closest of the largest box that does not contain any criteria vector.

III. Multi-criteria reinforcement learning for routing in ad-hoc networks

A. Choosing the best routes according to the returns of the environment

For each class of application (VoIP, FTP, video streaming, etc.), there is a specific routing policy fitting at best the requirements for each of them. The criteria evaluation for a packet is made when it reaches its destination.

Let a, s, d and n respectively stand for "application class", "source node", "destination node" and "neighbor node". A router r has to choose a neighbor $n*$ according to a and d such that the criteria vector evaluated on the destination node d satisfies at best the QoS requirements according to a. The criteria vectors are seen as realizations of a random vector $\mathbf{C}$. It is assumed that each router r, keeps in memory expected criteria vector estimations $\mathbf{Q}(a, d, r, n) = \hat{\mathrm{E}}(\mathbf{C}|a, d, r, n)$. Note that (a, d) are given by the packet transmitted by router r. Let n^* be the best neighbor according to the QoS requirements for application class a. n^* is chosen by router r by comparing the expected criteria vectors $\mathbf{Q}(a, d, r, n)$ for every neighbor n. These multicriteria comparisons are implemented by the Russian doll method.

B. Discovering routes by reinforcement learning

Reinforcement learning (RL) [2] takes place in the computation of $\mathbf{Q}(a, d, s, n)$ only at the source node of a packet indexed by t according to the following expression:

$$\begin{aligned} \mathbf{Q}_t(a, d, s, n) &= \mathbf{Q}_{t-1}(a, d, s, n) & \\ &\quad + \alpha(\mathbf{c}_t - \mathbf{Q}_{t-1}(a, d, s, n)), & \text{if } t > 0 \\ \mathbf{Q}_0(a, d, s, n) &= 0 & \text{otherwise} \end{aligned}$$

where

- criteria vector $\mathbf{c}_t$ is evaluated for packet number t at its destination. It is seen as the return of the environment of router s ;
- α is a constant step-size parameter: $\alpha \in [0, 1)$.

Our RL approach has been designed to work well with the Interleave Division Multiple Access (IDMA) method [14] that allows several nodes to transmit data simultaneously. However,

hal-00565838, version 1 - 14 Feb 2011

the number of simultaneous communications must be as small as possible. For this reason we have not used a protocol of the Q-routing family.

Reinforcement learning requires finding a good equilibrium between exploitation and exploration [2] in order to be able to converge as quickly as possible towards optimal or near optimal solutions. Exploitation consists in choosing the action, i.e. the neighbor of a node, with the best-expected return. However a learning process with too much exploitation, leads to sub-optimal solutions. Exploration is required to improve the quality of the best found solution. Exploration consists in choosing an action at random, i.e. choosing the neighbor of a node at random. However, too much exploration results in a waste of resources and low returns.

It is impossible to have an a priori estimation of the packet QoS involved in the exploration process. The exploration packets should then be used only for the computation of the $\mathbf{Q}_t$ vectors. When an exploration packet is sent by source s towards destination d, router s chooses a neighbor randomly. The other routers of a path choose their best neighbor according to their $\mathbf{Q}_t$ vectors.

IV. Experiments

A. The case study

An access method allows several users to share the same medium, i.e. electromagnetic waves for wireless communications. The Interleaved Division Multiple Access (IDMA) method [14] is a new access method similar to CDMA: users can send the data simultaneously and in the same band. The separation of the different users is based on the attribution of different interleavers but the code remains the same. Recent studies [15] have shown that it is even possible to use the same interleaver for all the users provided that the communications are asynchronous. This property is well suitable for MANETs. However, the greater the number of simultaneous transmitting nodes in the neighborhood, the lesser the signal to noise ratio and the greater the error rate. That involves that the quality of links between nodes is not stable and changes quickly. It is a challenge for the conception of good routing policies.

The experiments aim to estimate the efficiency of the Russian doll multicriteria decision method. The case study involves 20 boats in a rectangular region 100×150 km^2 that transmit data towards a terrestrial station located at coordinates (0, 75km). All the boats use the IDMA method.

During communications, boats are considered as stationary because of their low speed compared to distance between them. However, the features of communication channels change quickly because of the variability of the signal to noise ratios.

B. Simulations

The integration of the physical layer on a standard network simulator involves a high software complexity. There is no current simulator able to offer this feature. To overcome this difficulty, we have decided to work with the combination of two simulators that we have developed: one dedicated for the physical layer and IDMA access method simulation and the other for the network layer in charge of the routing tasks. The source nodes are chosen at random. Node 0, which is the terrestrial station, is the only destination of data packets.

A time period is a set of 40 successive time slots. For every time slot, a random number of simultaneous messages from 1 to 5 are sent through the network. A half of the messages is used to train the routers. The other half is used to provide performance results.

In the frame of the presented experiments, $\mathbf{Q}_t(a, d, r, n)$ is a vector $(\delta_t, \varepsilon_t)$ composed of the expected delay δ_t and expected error rate ε_t at time period t. These values have to be minimized by choosing adequate neighbors $n*$ for every router r and destination 0. Four application classes are defined for these experiments. Class 1 requires minimizing ε only. Class 2 requires the minimization of δ only. Class 3 requires simultaneous minimization of δ and ε if $\delta \leq 3.5$. If $\delta > 3.5$, δ is minimized only. Class 4 requires simultaneous minimization of δ and ε if $\delta \leq 4.5$. If $\delta > 4.5$, δ is minimized only. Two experiments are performed.

Experiment 1 uses standard MCDM methods. However, such methods are unable to meet the exact requirements for application classes 3 and 4. So, they are simplified by dropping the thresholds 3.5 and 4.5 for classes 3 and 4 respectively. We differentiate class 3 and 4 by using two MCDM methods with the same objectives in order to compare them with the Russian doll method. The MCDM method for class 3 minimizes the weighted Chebyshev distance $d^{\infty}_{\mathbf{w}}(x, \rho)$ between solution $x = (\delta, \varepsilon)$ and the reference point $\rho = (1, 0)$ with weight 0.1 for δ and weight 0.9 for ε:

$$d^{\infty}(x, \rho) = \max\left(0.1(\delta - 1), 0.9\varepsilon\right)$$

The weights have been chosen as normalizing coefficients since no indication was given about the importance of criteria. The MCDM method for class 4 minimizes the weighted sum of δ and ε with the same weights as for class 3.

Experiment 2 uses the Russian doll method. Now, the specification of classes 3 and 4 can be fully implemented by an appropriate choice of shapes for the nested boxes. According to section II, box 0 is defined as the whole criteria domain defined by the extreme points (0, 0) and (10, 1) where the first coordinate stands for δ and the second one stands for ε.

The nested boxes for the different application classes are represented in figure 2. The box upper bounds are given in table II.

TABLE II
Box upper bounds according to the application classes

Class	box 1	box 2
1	(10, 0)	
2	(1 , 1)	
3	(3.5, 1)	(1, 0)
4	(4.5, 1)	(1, 0)

In summary, box 1 implements the threshold 3.5 or 4.5 on

hal-00565838, version 1 - 14 Feb 2011

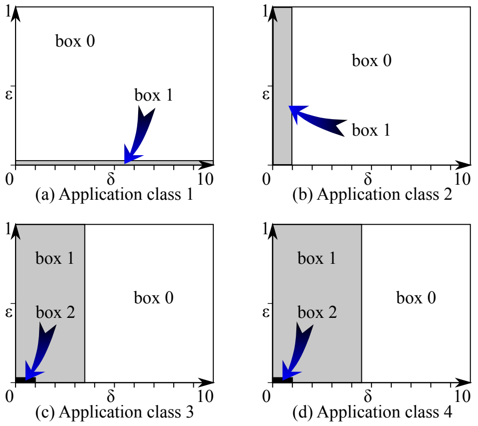

Fig. 2. Nested boxes of the Russian doll method for the four application classes

δ for classes 3 or 4 respectively. Definition of box 2 allows the MCDM algorithm to minimize simultaneously δ and ε.

C. Results

Each simulation gives 4 sets of numbers $d_t(a)$, $s_t(a)$, $\delta_t(a)$ and $\varepsilon_t(a)$ for each application class a, where t is the time period number:

$d_t(a)$: average of measured delays over all packets sent during time period t for application class a.

$s_t(a)$: average of success rates, $s_t(a) = 1 - e_t(a)$, where $e_t(a)$ is the measured error rate.

$\delta_t(a)$: average of $\mathbf{Q}_t(a, 0, s, n^*)$ first coordinate (expected delay) at source nodes. For each packet, source node s is randomly chosen. Its destination is always 0.

$\varepsilon_t(a)$: average of $\mathbf{Q}_t(a, 0, s, n^*)$ second coordinate (expected error rate) at source nodes.

Fig. 3 and 4 give the delays $d_t(a)$ and success rates $s_t(a)$ vs. time period t, for experiment 2. The figures show that $d_t(a)$ and $s_t(a)$ are clearly differentiated according to application class a. Thus, the network is able to configure itself to route simultaneously data packets according to the specifications of the different application classes.

In order to show how the learning process adapts the routing policy following a failure of some nodes, the success rates and delays for two nodes close to the destination (over 4) are suddenly and strongly degraded from the 300-th time period. Figures show that the routes are changed after some time periods to prevent the routers to choose these nodes. The network is then able to almost recover its previous performance.

Figure 5 presents the histograms of the expected delays estimations $\delta(a)$ for the Russian doll decision making. They are not displayed as bar charts as usual, but as continuous functions for the sake of readability. The width of each bin is 0.1. The frequency at abscissa x is obtained from the number of values in interval $(x - 0.1, x]$.

The effect of the boundaries of box 1 of the Russian doll method are clearly visible on fig. 5 for application classes 3 and 4. Indeed, the difference of frequencies between bins 3.4 and 3.5 for the application 3 histogram is the greatest among

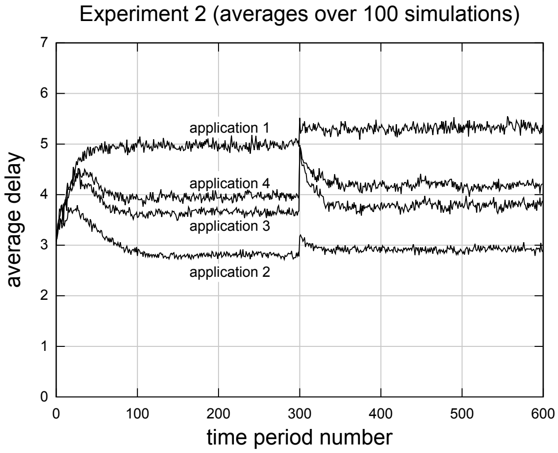

Fig. 3. Average delays vs. time period number according to application classes for experiment 2

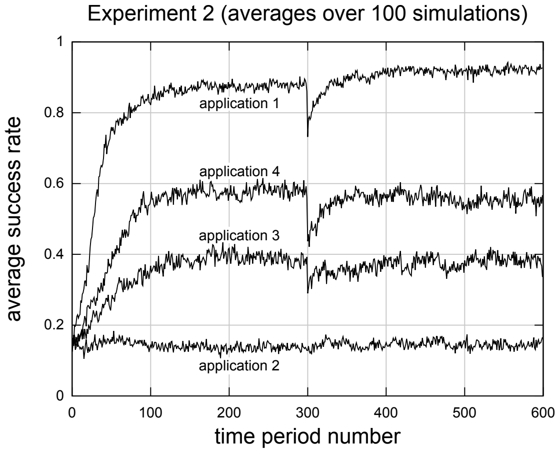

Fig. 4. Success rates vs. time period number according to application classes for experiment 2

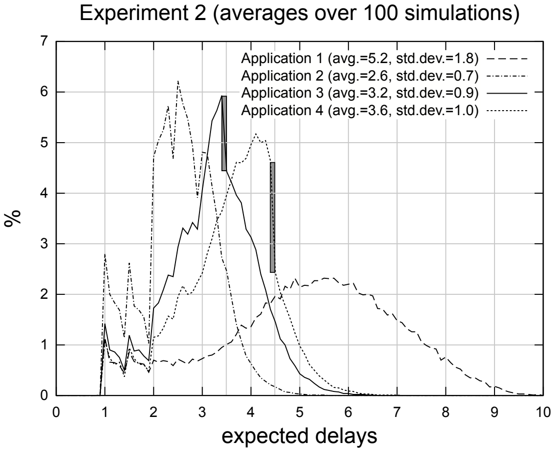

Fig. 5. Histograms of the estimation of expected delays according to application classes for experiment 2 (bin width = 0.1)

hal-00565838, version 1 - 14 Feb 2011

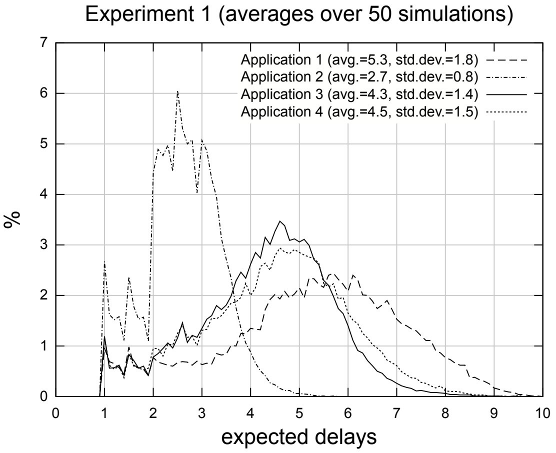

Fig. 6. Histograms of the estimation of expected delays according to application classes for experiment 1 (bin width = 0.1)

consecutive bins, while 3.5 is the δ boundary for box 1 of application class 3. Similarly, the same phenomenon occurs between bins 4.4 and 4.5 for the application 4 histogram, while 4.5 is the δ boundary for box 1 of application class 4.

The standard deviation for application class 1 is quite high as expected because the only minimized criterion is ε. For the other classes, standard deviations of $\delta(a)$ are notably lower. These values show that the δ boundaries of box 1, for $a = 3$ or 4, behave like an attractor. These values will be compared below with results of experiment 1.

Experiment 1 carries out standard MCDM methods only. Figure 6 presents the histograms of the expected delays estimations $\delta(a)$ for the weighted Chebyshev distance minimization (application class 3) and for the weighted sum minimization (application class 4).

Compared to experiment 2, standard deviations for application classes 3 and 4 are notably higher. The histograms of $\delta(a)$ for these classes are not very different from the one of application class 1. So, the "attraction effects" towards desired objectives are much less marked than for experiment 2.

V. Conclusion

Setting the parameters of most MCDM methods to fit the preferences of a decision maker is a difficult task. This paper has presented a method able to choose the best multicriteria solution according to a context defined beforehand, whatever the Pareto front. This context is given by a set of nested boxes in the criteria space, the shapes of which are defined by objective facts as technical standards, technical specifications, etc. This kind of method is well suited for self-adaptive systems because it does not need the intervention of a decision maker to adjust parameters and to assess the quality of the method results.

The Russian doll multicriteria decision method has been used with a reinforcement learning to optimize the routing in a wireless network. The results on a case study show that the routing can be finely controlled because of the possibility to include as much parameters as desired to adjust the search of the best solution on Pareto fronts a priori unknown. These results was clearly better than those obtained with the minimization of a weighted sum or a Chebyshev distance to a reference point.

More work has to be done to refine the Russian doll multicriteria decision method for self-adaptive systems. In particular, the method could be improved when the Pareto front is strongly concave by using polyhedral nested boxes instead of hyper-rectangular boxes only. Its applicability should also be evaluated in multiple contexts.

Acknowledgment

This work is a part of the OPERRA project supported by the "Institut TELECOM", France.

References

[1] Mehran Abolhasan, Wysocki, T., and Dutkiewicz. E., "A review of routing protocols for mobile ad hoc networks". *Ad Hoc Networks*, 2:1, January 2004.

[2] Sutton, R., Barto, A., *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, Massachusetts, 1998.

[3] Watkins, C.J.C.H., Dayan, P., *Machine Learning*, 8:279-292, 1992

[4] Tao, T., Tagashira, S., and Fujita, S., "LQ-Routing Protocol for Mobile Ad-Hoc Networks". In *Proc. of the 4th Annual ACIS International Conference on Computer and information Science* (July 14 - 16, 2005).

[5] Di Caro, G., Ducatelle, F., and Gambardella, L.M., "Anthocnet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks", *European Trans. on Telecom.*, *Special Issue on Self-organization in Mobile Networking*, vol. 16, pp. 443455, 2005.

[6] Deb. K., *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley, 2001.

[7] Ishibuchi, H., Nojima, Y., Narukawa, K., Doi, T., "Incorporation of decision maker's preference into evolutionary multiobjective optimization algorithms", in *Proc. of GECCO '06*, 741–742, 2006.

[8] Roy, A., Das, S.K., "QM2RP: a QoS-based mobile multicast routing protocol using multi-objective genetic algorithm. *Wirel. Netw.* 10, 3 (May), 271-286, 2004.

[9] Guerriero, F., De Rango, F., Marano, S., Bruno, E., "A biobjective optimization model for routing in mobile ad hoc networks", *Applied Mathematical Modelling*, Volume 33, Issue 3, March 2009, Pages 1493.

[10] Egoh, K., De, S., "A Multi-Criteria Receiver-Side Relay Election Approach in Wireless Ad Hoc Networks", *MILCOM 2006*, pp. 1-7, 2006

[11] Setmaa-Krkkinen, A., Miettinen, K., Vuori, J., "Best compromise solution for a new multiobjective scheduling problem", *Comput. Oper. Res.* 33, 8 (Aug.), 2353-2368, 2006

[12] Mahmoud, Q., (Ed.) *Cognitive Networks: Towards Self-Aware Networks*, Wiley-Interscience, 2007

[13] Bowman, V. J., "On the relationship of the Tchebycheff norm and the efficient frontier of multiple-criteria objectives", *Lecture Notes in Economics and Mathematical Systems*, Vol. 135, pp.76-85, 1976

[14] Ping, L., WU, K.Y., Liu, L., Leung, W.K., "On Interleave-Division Multiple-Access" in *Proc. of IEEE ICC Conference*, pp 2869-2873, 2004

[15] Houcke, S., Sicot, G., Debbah, M., "Blind Detection for block coded interleaved Division Multiple Access" *GLOBECOM*, San Fransisco, USA, 2006