

A self-adaptable distributed CBR version of the EquiVox system

Julien Henriet, Christophe Lang, Ronnie Muthada Pottayya, Karla Breschi

*Université de Franche-Comté, FEMTO-ST DISC, UMR 6174 CNRS
16 Route de Gray, 25000 Besançon, France*

*julien.henriet, christophe.lang, ronnie.muthada_pottayya, karla.breschi
@univ-fcomte.fr*

Abstract

Three dimensional (3D) voxel phantoms are numerical representations of human bodies, used by physicians in very different contexts. In the controlled context of hospitals, where from two to ten subjects may arrive per day, phantoms are used to verify computations before therapeutic exposure to radiation of cancerous tumours. In addition, 3D phantoms are used to diagnose the gravity of accidental exposure to radiation. In such cases, there may be from ten to more than a thousand subjects to be diagnosed simultaneously. In all of these cases, computation accuracy depends on a single such representation. In this paper, we present EquiVox which is a tool composed of several distributed functions and enables to create, as quickly and as accurately as possible, 3D numerical phantoms that fit anyone, whatever the context. It is based on a multi-agent system. Agents are convenient for this kind of structure, they can interact together and they may have individual capacities. In EquiVox, the phantom's adaptation is a key phase based on ANN interpolations. Thus, ANNs must be trained regularly in order to take into account newly capitalised subjects and to increase interpolation accuracy. However, ANN training is a time-consuming process. Consequently, we have built Equivox to optimize this process. Thus, in this paper, we present our architecture, based on agents and ANN, and we put the stress on the adaptation module. We propose, next, some experimentations in order to show the efficiency of the EquiVox architecture.

Keywords: distributed case-based reasoning, multi-agent system, 3D numerical phantoms, adaptation.

1. Introduction

Three dimensional (3D) voxel phantoms are numerical models of human bodies and are used by physicians and medical doctors in very different contexts: to diagnose the gravity of accidental exposure to radiation (radiation protection) and also to control the calibration of Treatment Planning Systems used to eradicate cancerous tumours in hospital departments (radiotherapy). These 3D organ and body representations must fit those of the subject as well as possible. They can be generated from ionizing images (scanners), Magnetic Resonance Imaging and/or ultrasounds. However, these methods require expensive and highly specialized equipment accessible only by appointment; they also need preparation and surveillance. Such constraints significantly increase delays before any accurate image can be obtained. Moreover, some of these methods expose the already weakened subjects to additional radiation. Consequently, the generation of 3D phantoms without recourse to expensive and potentially harmful equipment is a useful and safe alternative in all cases of exposure to radiation.

EquiVox is a tool implementing artificial intelligence concepts in order to find and customize numerical phantoms of human beings [1]. This paper aims at presenting and analyzing the different improvements which have been made in this tool. Case Based Reasoning (CBR) is a problem solving method that uses similar solutions from similar past problems in order to solve new problems [2]. The EquiVox platform uses the CBR approach to find the most similar phantom(s) within any set of phantoms and then attempts to adapt them to the characteristics of the target case (the subject). EquiVox adaptation tool uses Artificial Neural Networks (ANN). The EquiVox case-base can be supplied with 3D representations created in different hospital departments or even in field hospitals established temporarily near the sites of nuclear accidents. Thus, knowledge can be distributed over different sites or can be shared and improved by different users situated in different places [3]. Globally, EquiVox follows the path of any other CBR tool. Whenever a person has to be modeled, the most similar phantom is found. This phantom is then automatically adapted by the system after the retrieval phase. The adaptation phase implements an ANN [1, 4] : the target case coordinates of each point of each organ contour is interpolated by taking into account the coordinates of the same point on the closest phantom and having the

same size as that of the target case. The adapted phantom is then revised by the expert, then capitalized. During the capitalization phase, the ANN is always trained so that the next adaptation phase takes into account this new phantom. We have optimized the accuracy of interpolations by introducing a weighting vector for each phantom [5, 6]. This optimization has led to a counter effect on the capitalization phase making the latter a very long time consuming process. Hence, forcing us today to distribute the system.

The problem is that EquiVox may be used when less than ten new subjects must be represented (in a normal hospital context) one day, and the next day, when ten to a hundred of human bodies must be provided (just after accidental exposures to radiations, for example). This important variation of arrivals introduces many difficulties, and the last version of EquiVox was not always capable of providing the most accurate representations for all the subjects in an acceptable time. Therefore, we turned ourselves towards a multi-agent model: each agent supports one of the phases of the CBR. Through a probabilistic study [7], we initially evaluated the theoretical performances of this first multi-agent model. The results demonstrated that the multiple instances of all agents were not necessary and that only the multiple instances of the agent in charge of the capitalization were sufficient in order to improve the performance of the EquiVox system.

Besides, the simple multiplication of all agents cannot be a viable solution as all hospitals do not have access anytime to clusters to parallelize ANN learning. Hence, we had to think about a system that does not systematically capitalize, which is able to postpone the long capitalization phases, while maintaining the highest level of precision for the supplied solutions. So, we tested, compared, and analyzed various capitalization strategies. Any capitalization strategy (or learning, if one takes the point of view of the ANN) must consider the input flow of the system, and be put forward automatically in real time by the system. Therefore, this EquiVox distributed version differs from other CBR by the implementation of an adaptive capitalization, deferrable, and distributed, rather than a linear and systematic capitalization. In addition, no human intervention is required, EquiVox modifying the capitalization process by itself in real time.

Thus, this paper presents and evaluates the original modifications we have made to this distributed architecture in response to this specific problem.

2. Materials and Methods

The first part of this section presents EquiVox, its objectives, the inherited difficulties and the original architecture we have designed in order to deal with them. The difficulties posed by the previous non-adaptative version of EquiVox are also highlighted through an example in this first subsection. The second part of this section presents the detailed new agents of EquiVox and their behaviors.

2.1. The EquiVox objectives

3D phantoms provide solutions to many situations. In case of accidental exposures, the computed impacts of both the external and internal (inhaled) doses are reported. EquiVox 3D phantoms are used in this particular case, to estimate the dose due to the quantity inhaled. These representations are of precious help for dosimetric reports of inhaled substances since in this particular case, even if the radiation source is not well known, an accurate idea of the inhaled dose can be computed.

Other non-ionizing imaging methods such as Magnetic Resonance Imaging and ultrasound may provide accurate images as well. These methods, however, require expensive and highly specialized equipment which may not be readily available after the accident. If it is available anywhere nearby, it may already be in use by local hospitals and accessible only on appointments, which significantly increases delays before any accurate images can be obtained. In addition, radiotherapy services today treat an increasing number of cancerous lung tumors, with physicians and medical doctors using Treatment Planning Systems to treat tumors. The equipment necessary for this requires preparation and control. Its suppliers use 3D phantoms to control their calibration, though it would be reassuring to also use 3D representations from independent sources.

Finally, the use of this equipment requires additional appointments, with further and inconvenient testing, thus creating additional constraints. The already weakened subjects (elderly subjects in the case of cancer treatment and subjects in shock in the case of accidental exposure) are thus under greater stress and in need of quick reassurance. Consequently, 3D phantoms are useful alternatives and also comprise preliminary models in all cases of exposure to radiation.

2.2. Case modeling and retrieval phase

We have studied the list of useful characteristics furnished by the physicians of the French Institute of Radiation and Protection (IRSN). Thus, in EquiVox, a problem is described as a set of r descriptors $\{d_1, \dots, d_r\}$. These descriptors are external measures of the human body: height, gender, age, under bust circumference and chest circumference. Table 1 presents three examples of phantoms stored in the initial case base of EquiVox. There are two adult women phantoms and an adult man phantom of different ages and sizes.

Each expert has its own set of n phantoms: $SP = \{P_1, \dots, P_n\}$.

Each P_i is the solution part of a case and represents the contours of m organs: $P_i = \{P_i^1, \dots, P_i^m\}$.

Each organ O is a set of q points joined by a Delaunay mesh [8]: $P_i^O = \{C_1^{i,O}, \dots, C_q^{i,O}\}$ where $C_j^{i,O}$ denotes the 3D coordinates of point j of organ O of phantom P_i , $O \in \{\text{lung, heart, liver, sternum, ribs, scapulae, spine, breasts, skin, oesophagus, thorax}\}$.

Finally, a case i is: $i = \{d_1^i, \dots, d_r^i, P_i\}$.

The retrieval phase lies in sorting the source cases according to their similarity with the target case t . We implemented a classical version of the *k-Nearer Neighbour* (kNN) algorithm [9] which computes the distances between each descriptor of the problem parts of target and source cases. Thus, for each case i , a similarity index S_i is computed as follows:

$$S_i = \frac{\sum_{k=1}^n \frac{\Delta_k - |d_k^i - d_k^t|}{\Delta_k}}{n}$$

where Δ_k is the difference between the maximum and the minimum known values that the descriptor d_k can take. The S_i value is always between 0 and 1. The greater the similarity of i to t , the closer the S_i value to 1.

2.3. The agents in EquiVox

Hence, the actual system aims at assisting the physician in choosing and customizing the most similar phantom from among the existing available ones. The EquiVox platform uses the CBR approach to find the most similar phantom(s) within any set of phantoms and then attempts to adapt them to the characteristics of the target case (TC) (the subject). The EquiVox adaptation tool uses ANN to adapt the stored phantoms in accordance with the subject. As previously explained, different experts may use this platform

in very different contexts. In the context of hospitals, 2 to 6 new subjects can arrive every day. In the context of large hospitals or in case of minor radiation contamination accident, we can consider that 10 to 15 new subjects usually arrive every day. If we consider the use of EquiVox in the context of accidental exposure to radiations, EquiVox must provide 30 up to 80 (massive radiation contamination) new 3D representations every day. The main purpose of this study is to present EquiVox with a distributed and cooperative architecture in order to merge and share the adaptation knowledge (AK) and phantoms drawn by all the users.

The first version of the distributed architecture was designed to suit the material architecture [7]. Figure 1 represents the intra-agent control model over the EquiVox architecture. The entire set of phantoms is stored under Rhino3D files on a central server. The corresponding biometric data is stored under an ontology scheme on the same server.

Whenever a new subject arrives, a new phantom is required. The Descriptor Agent (DA) is in charge of describing the target case. Once this phase completed, the described target case is transmitted to a Reminder Agent (RmA). The latter determines the similitude index by taking into account the source cases found on the central server (in the case base). This phase is known as the retrieval phase and as is can be seen in the figure, the RmA is responsible of contacting the server and retrieving back the phantom. The retrieved source case and the target case from the RmA then undergo an adaptation phase. An Adaptation Agent (AA) considers the characteristics of the source case and calls the interpolation module. This module loads the trained ANNs and the coordinates of the organ contours in question in order to create interpolated contours suited to the target case and which are finally combined to create the interpolated organ contours. At this stage, there are two possibilities. Whether the Expert Agent (EA) accepts the phantom as it is or else requires that a revision phase be performed. According to the EA, if the phantom doesn't match the subject, it then revises and finalizes the adapted phantom and sends this revised solution for capitalization.

As mentioned above, training of an ANN module is a time consuming process and when the latter is in this state, it cannot be used by any AA for interpolation purposes. Hence, it has been necessary to find procedures and techniques which will enable the system of being used and trained at the same time. Besides, when used, an ANN must have the latest data in order to be the more optimal possible. The architecture we have designed can cope with these two challenges. Firstly, it is guaranteed that whenever

the ANN is used, the latter always dispose of the latest data and secondly during its training phase, the system can still be used. When an expert sends the revised case for capitalization, these new data are buffered temporarily on the the central server. According to the strategy employed for training ANNs, when a certain amount of capitalized cases is reached or it is time for launching an ANN training, it is the Strategy Agent (SA) which is put forward. In fact, this agent acts like a coordinator and will orchestrate the activation and deactivation of Retainer Agents (RtA). New RtAs will be activated if the SA estimates that there is not enough RtAs to perform ANN trainings. Else, RtAs will be deactivated. As already mentioned, RtAs will take all the phantoms awaiting for capitalization from the buffer and will perform the capitalization phase. It is to be known that whenever a training process is in progress, there always exists a copy of a latest operational ANN module; the SA guarantees this aspect of the system.

Figure 2 depicts the different agents that implement our system. We have mapped agents on the functions described in Figure 1. The DAs handle functions surrounding TCs. The RmAs deal with the retrieval phase. The AAs are mandatory for adaptation. The experts are considered as agents (*Expert Agents*, EA) in our model. They describe and revise the solutions. Finally, the RtAs deal with the global knowledge.

We have two main categories. The first one concerns human agents, the expert and the subject who is being studied (they inherit from the general class). Despite that they are human beings, we can represent them by means of agents in a simulation software tool. The second category concerns simulation software agents which are programs designed and implemented in distributed EquiVox. They will both act upon data represented by resource class named General Knowledge. On this figure, we can clearly see the influence that the strategy agent will have on retainers. At last, we have a class that represent the General Knowledge. It is independent on an AML class diagram since in these diagrams we only represent inheritance links. But, in the MAS System each agent can access, at least in theory, to this resource.

2.4. The adaptation module of EquiVox based on ANN and its limitations

To interpolate the 3D contours of the organs, the subject's height must be known. Indeed, as concluded by I. Clairand *et al.* study [10], organ shapes and volumes depend on the subject's height.

In fact, bust circumferences are added during the revision phase when the target is an adult woman. These parameters have no influence on the shape of

organs. They help to determine breast models to add to corresponding male phantoms and generated by the ANN. It should be noted that the ANN will always provide male phantoms. These phantoms are then adapted (addition of a breast model in respect to bust circumferences) to fit female phantoms.

Actually, this is one of the features of the EquiVox target and source cases. Let us note h_i the feature corresponding to the height of the case i and h_t , the height of the TC.

An ANN with 5 inputs and 3 outputs was simulated, and the source case SIM which height h_{SIM} is the closest to the TC is considered.

The trained ANN interpolates the 3 coordinates of each point of the organ contours separately. Thus, the 5 inputs enabling interpolation of the coordinates $C_k^{t,O}$ of point k of organ O of t are:

- The 3 coordinates of point k of O contours of SIM : $C_k^{SIM,O}$;
- The height of the TC: h_t ;
- The difference between h_t and h_{SIM} : $\Delta h = h_{SIM} - h_t$.

For example, suppose that a person measuring 1m70 must be modeled and that the most similar phantom of a given hospital’s knowledge base is the phantom P_4 measuring 1m65. During EquiVox adaptation phase, each contour point of each organ will be interpolated by a trained neural network available in the knowledge base. Each point C_k^t of the phantom representing the person measuring 1m70 will hence be equal to $C_k^t = Interpolate_{RNA}(C_k^4, 170, -5)$ (-5 parameter representing the difference between the subject’s phantom size and the most similar retrieved phantom’s size).

A cross validation has been performed with this approach [4]. This approach gave satisfying results since the errors were generally inferior to the spatial resolution of the initial representation and have been validated by IRSN experts.

The adaptation knowledge is thus composed of an ANN that must be previously trained over the set of known phantoms. This training step, assumed by the set of RtAs, is time-consuming; we first analyzed the learning duration according to the number of phantoms at our disposal. As reported in Figure 3, these results tend to prove that the learning delay follows an affine function: $time(nbPhantoms) = 4080 \times nbPhantoms + 10500$ seconds. Thus, if there is only one retainer agent, the whole system is locked during each ANN training. This is the reason why we must find a solution that

allows the use of the system even if an ANN training is in progress. This solution consists in distributing and multiplying the agents. Nevertheless, this implies that RtAs are able to launch ANN trainings concurrently, and thus generating concurrent versions of the AK.

Figure 4 helps to understand the difficulties implied by this time-consuming learning process of the ANN. In this figure, the strategy of one agent that performs an ANN learning act every day at 8 o'clock is compared to the strategy of one agent that performs an ANN learning act for every 5 new cases when 8 new cases arrive every day. Let us call A_8 the agent that performs the ANN learning every day at 8 o'clock, and A_5 the agent that performs the ANN learning for every 5 new cases. Let us designate 8 o'clock as $T1$ (time 1), when agent A_8 launches an ANN learning, a process that will end at $T2$. Thus, at $T2$, an ANN based on the learning experience gained from 3 subjects is available. At $T3$, since 5 new subjects have just been modelled, A_5 launches an ANN learning that will end at $T5$. Before that, A_8 has already launched (at $T4$) another process of adaptation knowledge that will end at $T6$. At $T7$, A_5 performs an ANN learning based on 10 subjects. This ANN will be usable at $T9$. Since there is only one agent available for this strategy, and since this agent A_5 is already involved in performing an ANN learning, it cannot perform another one before the end of this second day when 15 subjects will have arrived (this situation occurs sometime between $T7$ and $T8$). Finally, at $T11$, A_5 launches an ANN learning based on 20 subjects. Nevertheless, at this timestamp, A_5 has provided an ANN based on only 10 subjects, whereas A_8 has provided an ANN based on 18 subjects (launched at $T8$ and ended at $T10$). Consequently, at $T11$, the strategy of A_8 is better than that of A_5 . Actually, the learning delays penalize the A_5 strategy when 8 new cases arrive every day.

For this reason, the strategy of learning must be a real-time process adapted to the variations of the context of use of EquiVox.

2.5. Evolution and originality of this distributed version of EquiVox

As a matter-of-fact, it is necessary to adapt the number of agents (system topology) and the learning strategies (load balancing) to the arrival flow in real-time. The distributed platform must evolve into a self-adaptable system that (1) observes its environment, (2) analyzes the situation, (3) schedules the tasks and (4) undertakes any changes necessary in order to balance the charge of the retainer agents that support the most time-consuming tasks. This is precisely the task of the strategy agent.

Figure 5 represents the communication flows between the agents of the distributed platform. Each target case is taken in charge by one expert within a pool of experts, which in turn communicates directly with one descriptor agent within a pool of descriptor agents. The chosen descriptor agent transmits the considered target case to one of the reminder agents and also signals to the unique strategy agent that a new target case has arrived. The chosen reminder agent then communicates the most similar retrieved source case together with the target case to the adaptation agent which performs the adaptation process in order to have an adapted phantom of the target case. Afterwards, the adaptation agent sends this adapted phantom to one of the expert agents who may or may not revise it before sending the final revised target case to one of the activated retainers.

Though there is no need for communication between agents such as descriptor, reminder and adaptation agents, retainer agents, they must, on the other hand, share information. Actually, the retainers must share some information in order to know if it is activated, when to launch ANN training, and not to undertake the same training sequence twice for the same training set. These agents must all know what the current strategy is, which retainers are activated, which agent is launching ANN training, with how many new cases, at which launch date, etc. To divide the memory among the set of retainers, we have chosen the Pilgrim protocol which is an efficient and secure protocol for concurrent, cooperative and collaborative works with shared memory [11]. The agents are dispatched over a ring and a token is exchanged. The originality in this protocol is the fact that the token carries each stored modification. Each agent has a copy of the shared memory. In this protocol, the token is a structured entity that is transmitted from agent to agent and is dispatched over a logical ring in an order determined at the beginning of the cooperative work. When an agent wants to modify the shared memory, it must place a reservation flag above the token if there is not yet another flag above it. The token continues its course over the ring and when the agent again receives the token, it is allowed to write the modifications. Thus, all the other agents receive these modifications during the next token revolution, after which all the agents have exactly the same version of the shared memory even if two or more agents want to modify it at the same time. Thus, before launching ANN training, each retainer must verify whether it is activated and whether another agent has not already launched a training.

The strategy agent is in charge of the choice of the retaining strategy according to the entry rate fluctuation and thus, activates and deactivates

the required retainers with the chosen learning strategy.

Figure 6 shows by means of a simplified sequence diagram how a retainer is activated by the strategy agent. In fact, when the descriptor agent signals to the unique strategy agent that a new case has arrived, the latter increments the actual number of cases being treated and calculates the entry rate fluctuation. If this rate is higher than the tolerance threshold, meaning that there are not enough activated retainers, the strategy agent signals to an inactivated retainer agent that it will now be activated.

In contrast, Figure 7 shows how a retainer is deactivated by the strategy agent. When the entry rate fluctuation is calculated lower than the tolerance threshold, meaning that there are more activated retainers than required, the strategy agent signals to one of the activated retainer agents that it is to stop all activities and disable itself. Upon receiving this message, the selected retainer finishes its learning sequence - if one had been in progress - waits for the token and writes the data to be shared with the other retainers in the token and passes into a disabled state. The strategy agent is then aware that the system possesses one less retainer.

3. Results

In this section, we have first evaluated the different possible strategies of capitalisation separately, and we have then evaluated the new adaptative strategy.

3.1. Evaluation of the strategies separately

The study conducted previously [7] showed that the multiplication of all agents was not necessary in order to improve the performance of the system. Only the multiplication of agents responsible of capitalization affects positively the performance of EquiVox.

Using this architecture, we then evaluated the different configurations at the disposal of the strategy agent. These configurations are characterized by three parameters:

- the arrival frequency;
- the number of retainer agents;
- the learning frequency.

In order to evaluate the possible configurations and to program the choices to be made by the strategy agent, we studied different cases, combining different possible values for each configuration parameter. We then counted the number of phantoms our distributed platform was capable of taking into account.

We considered different arrival frequencies:

- 2 subjects arrive each day (hospital context);
- 6 subjects arrive each day (hospital context);
- 12 subjects per day (large hospital or a minor radiation contamination accident);
- 30 subjects per day (context of an accidental exposure to radiations);
- 80 subjects per day (context of a massive radiation contamination).

We implemented a varied number of retainer agents: 1, 2, 3, 4, 5 or 10 agents. We tested the different learning frequencies: concerning the learning delays, we assumed that performing one learning after the capitalization of each case would quickly overload the entire system. Consequently, we tested the following learning strategies for these retainer agents:

- one learning process performed every 8 hours;
- one learning process performed every evening at 8 p.m;
- one learning process performed every 3 new cases capitalized;
- one learning process performed every 4 new cases capitalized;
- one learning process performed every 5 new cases capitalized;
- one learning process performed every 10 new cases capitalized;
- one learning process performed every 15 new cases capitalized;
- one learning process performed every 20 new cases capitalized;
- one learning process performed every 50 new cases capitalized.

For this study, we assumed that there were 0 to 5 phantoms stored in the initial case-base. Hence, we can assume this initial situation corresponds to most of the initial situations in which EquiVox will be used. Indeed, when a lot of phantoms are stored in the case-base, we can assume that the remaining and adaptation phases are effective (a case very close to the subject can be found in this case-base), and there are few revisions to be performed on the adapted phantom proposed by the system.

Experiments have been made on a simulator developed with the GAMA platform. GAMA is a modelling and simulation development platform that is more especially used to spatially explicit agent-based simulations. It offers a graphical interface and a specific language, GAML, which enables the development of the Multi-Agent System. As the result of experiments, GAMA can produce useful charts. In our case, agents are not situated in a space environment, but our system is composed of distributed entities; GAMA can handle this kind of system.

Figures 8, 9 and 10 show the number of phantoms that can be taken into account in the customary hospital context and Figures 11 and 12 in the deteriorated context of disasters. Since these phantoms are used by hospital physicians, the latter's work schedules (8 a.m. to 8 p.m. 5 days per week) are taken into account. The tests were performed until a change took place in the number of new subjects arriving. As long as 2 subjects per day arrived, we stopped the series of tests at 30 subjects per strategy. When six new subjects began arriving every day, we ran tests up to a total of 100 subjects. For 12 new subjects who arrived every day we tested up to 150 subjects, and in disaster-context, we reached 300 arrivals.

Figure 8 shows that with the arrival of 2 new cases every day, the highest number of cases taken into account in the adaptation knowledge is obtained when one or more retainer agents undertake a learning act every day at 8 p.m. or for every 4 new cases, whereas a lower number of phantoms is taken into account when one or more agents perform an adaptation knowledge update for every 10 new cases. According to Figure 9, when 6 new cases arrive every day, the best results are obtained when one or more agents perform an update every 8 hours. Again, these best results are obtained, when at least 2 retainer agents perform an update every day at 8 p.m., or when more than 4 agents perform an ANN learning act for every 3 new cases. In contrast, the worst results are obtained when a learning act is attempted for every 5 or 10 new cases. Figure 10 shows that when there are 12 new arrivals per day, all strategies give more or less optimum results when ten agents are employed.

However, this optimum is obtained from one agent that performs one ANN learning every day at 8 p.m. or every 8 hours.

Figure 11 shows that when 30 new subjects arrive every day, the best results are obtained when one agent or more perform an ANN learning act every day at 8 p.m. Again, the best results are obtained when more than 4 agents perform adaptation knowledge updates every 8 hours. Best results are also obtained when more than 5 agents perform learning acts at every 15 new cases or when 10 agents launch ANN learnings at every 10 new cases. In contrast, performing learning acts every 3, 4 or 5 new cases are the least efficient strategies, whatever the number of retainer agents activated. According to Figure 12, when 80 new subjects arrive every day, best results are obtained with more than 2 agents that learn every 50 cases, more than 4 agents every 20 cases, more than 5 agents every 4 cases, 10 agents every 3 cases, or 10 agents every 15 cases. However, the worst results are obtained with only a single agent, whatever the strategy, and learning acts launched every day at 8 p.m. or every 8 hours, whatever the number of subjects.

As shown in Figures 8, 9 and 10, one agent that learns every day at 8 p.m. or 2 agents that learn every 8 hours will always take into account a maximum number of phantoms in all customary hospital contexts, whereas the other strategies, in order to be optimal, require 10 agents. Furthermore, in this particular context they are not always efficient as just two agents. Indeed, these strategies are well suited to physicians' work schedules.

In contrast, Figures 11 and 12 show that when arrivals are more numerous, these two strategies are progressively of lower efficiency than strategies based on 5 agents or more and learning acts performed at every 3, 4, 15 or 50 new cases. Hence, it is important to retain that the most efficient strategies in hospital contexts are the worst in the case of disasters, whereas the strategies that are the most efficient in a disaster context are the worst when used in a normal hospital context.

3.2. Evaluation of the adaptative strategy

These results drive us to the conclusion that there is not a single configuration that can handle all the situations. Indeed, the number of agents and their learning strategies must be adapted as soon as a variation of the number of arrivals is detected. Thus, a new type of agent has been introduced into the system and the RtAs now communicate, share information and act according to a concurrency management protocol. Considering these results, we have finally chosen to implement the configurations and adaptations of

Table 2: when there are less than 4 arrivals per day, the strategy agent activates 2 RtAs that launch ANN trainings every day at 8 p.m., when the strategy agent detects 4 to 9 arrivals per day, it activates 4 RtAs that launch ANN trainings every 8 hours and deactivates all the other RtAs. One more RtA is activated when the strategy agent detects 9 to 40 new subjects per day. Finally, when more than 40 arrivals per day are detected, this strategy agent activates 10 RtAs that launch ANN trainings every 15 new cases and deactivates all the other RtAs. As presented in Figures 8, 9, 10, 11 and 12, we could have chosen other optimized configurations.

In order to validate these choices, we conducted further investigations on our proposed system through three specific scenarios corresponding to three use cases whereby all the characteristics of these use cases differ from each other. In order to measure the performance of the system, we have recorded the number of subjects taken into account by the ANN in the different proposed strategies. We recall that the interpolations of the ANN will be better provided that a more significant number of source cases already exists in the case-base. The numbers of subjects to treat are shown on these curves and we have assumed that every subject is to be revised and retained by the system.

In the first scenario (Figure 13), the number of subjects arriving is at first low, then increases very rapidly after ten hours before becoming low again. This scenario corresponds to the case of an occasional radiological accident involving twenty subjects and occurring after a moderate utilization of the system. In this scenario, the strategy, "a learning every 8 hours" is more efficient than the one based on "a learning every 15 cases" up to 37 hours of utilization. The trend then reverses until the 61th hour of utilization before becoming strictly identical. The introduction of a Strategy Agent helps to provide an agreement and be always very close to the optimal number of subjects for the exploited ANN (identical to the strategy "a learning every 8 hours" until the 33th hour of utilization, then a subject less than the strategy "a learning every 15 cases" until the end of the simulation).

In the second scenario (Figure 14), the system is utilized moderately for several days before a radiological accident involving twenty subjects occurs. The strategy "a learning every 8 hours" is better than the one proposing "a learning every 15 cases". Again, the introduction of a "strategy agent", helps to stick to this optimum and to propose during the entire ANN scenario based on as much, even more, subjects than this optimal strategy.

Finally, in the third scenario (Figure 15), the system is put into operation

when a major radiological accident happens. The system's knowledge base contains initially five cases and is enriched by the modeled subjects involved in this major radiological accident. The strategy "a learning every 15 cases" offers more often trained ANNs on much more subjects than the strategies "a learning every 8 hours" and "a learning every day at 8 a.m.". However, as soon as the number of subjects arriving decreased, the strategy based on "a learning every 8 hours" becomes better. Here again, the introduction of a Strategy Agent optimizes the number of subjects taken into account by the operational ANN from the beginning till the end of the scenario, despite the fact that the inflow decreases.

4. Discussion and related works

These evaluations, therefore, show that the introduction of a Strategy Agent optimizes the number of subjects taken into account by operational ANNs, thus optimizing adaptation accuracies made by EquiVox even in case of big variations of the input stream.

EquiVox is a distributed CBR [3]. According to the classification proposed by E. Plaza and L. McGinty, the distributed version of EquiVox enables increasing the capabilities of calculation following the *Multiple Agents* model, provided that a centralized knowledge in a *Single Case Base* be used.

A large number of CBR systems designed for Health Science (CBR-HS) combined with Artificial Intelligence (AI) tools are now available [12, 13, 14]. Moehrle and Raskob propose a CBR approach which helps to identify solutions and cons-measures in case of nuclear disasters [15]. The special feature of their proposal rely on the fact that it is not only based on cases but also on scenarios. Like other systems [16, 17, 18], EquiVox provides a prototypical case for under-represented cases of a class using an ANN [19] based on its adaptation process [4].

A. Pla *et al.* propose a distributed version of their previous CBR tool designed for the medical sector [20]. Like in the real life, where multidisciplinary meetings, are used by physicians to concert on a particular case, the goal of their tool is the same here. When a physician has to pronounce a diagnostic on a complicated case, he has to take into consideration the opinions of his fellow physicians working on the same case. In their tool, physicians are agents. Each agent will use its own case-base. Once, all diagnostics gathered, they are transmitted to a coordinator agent which is responsible to take the

final decision. The difference between their architecture and ours lies in the fact that there isn't any adaptation process in theirs.

-Following the example of J. Hu *et al.* [21], we have designed a process that increases adaptation accuracy when the set of phantoms stored in the knowledge database grows [5]. We have proposed and have implemented an original method in order to optimise the accuracy of the adapted phantoms [6]. The main drawback of this method resides in the fact that it is time-consuming. This is one of the reasons why we propose here a distributed version of EquiVox based on the use of agents [7].

GI Hawe *et al.* designed an application where agents are used to simulate emergency responses in order to determine resource allocation in case where two major incidents would occur at the same time but situated at different locations [22]. An important point to highlight is that resources are limited and hence we need to make some compromises resulting in a solution closest to the optimal one. Furthermore, maintaining the case base and updating the entire knowledge base during the capitalization process is of the utmost importance for CBR-systems [23].

Most of the designed distributed CBR systems are composed of CBR-based agents that compare solutions provided by different CBR-engines and a decision policy is defined to decide which solution is best-suited to the proposed problem [24, 25, 26, 27, 28, 29]. In some approaches, a single agent coordinates, selects and delivers the tasks to the others [30]. The agent assuming this role can be chosen by the distributed system itself according to the situation [31, 32].

EquiVox differs from other approaches in several points. First of all, our platform combines several AI concepts therein: general operation according to the CBR, adaptation phase performed by an ANN, and process distribution following an agent-based model. One of the originalities of EquiVox lies in the ability of the system to postpone the knowledge capitalization phase at the end of a CBR cycle, while maintaining an optimum level of performance. Indeed, in other approaches of distributed CBR or systems integrating ANNs, the capitalization phase is systematically done at the end of each CBR cycle. In EquiVox, capitalization of adaptation knowledge is initiated depending in the system input stream. Several capitalization phases have been defined and tested. Optimal strategies have been identified according to the contexts in which EquiVox will be used. Finally, EquiVox can automatically change its knowledge capitalization strategy without any human interaction.

However, in these distributed platforms, either the system is capable of

treating a large number of problems though its actual solutions based on knowledge that may not always be up-to-date or is perhaps unshared, or the system can furnish a highly accurate solution but uses a time-consuming process. Actually, both of these abilities must be treated by the distributed version of EquiVox, since most hospitals dispose of fewer than ten new subjects each day with which to generate and accurately capitalise 3D phantoms. In contrast, in the cases of massive exposure to radiation, hundreds of subjects must be treated as soon as possible. In addition to these approaches, an interesting multi-agent system is also proposed by E. F. de O. Sandes *et al.* in which agents must balance the charge of the different nodes working in parallel [33]. Both this approach and ours treat the system load balancing by means of a multi-agent system (MAS), but in two different manners. In the system of E. F. de O. Sandes *et al.*, a set of balancers modifies the job queue of each executor agent, whereas our study makes use of a unique agent to regulate the frequencies of time-consuming computations and also the number of agents devoted to these tasks. The resulting distributed platform is self-adaptable since it is capable of observing its environment, analyzing the situation, scheduling the necessary changes in topology and agent charges, and finally of implementing these changes in real-time without any human intervention.

A future improvement of the EquiVox platform may consist in multiplying the number of case-bases in order to increase the initial number of phantoms and thus increase the accuracy of the adaptation process. Another option may consist in adapting 3D voxel phantoms and transferring them afterwards to the central server. For that purpose, the security and robustness of the images transmission becomes a central concerns. Hence, a robust protocol like REPRO [34] including techniques of reduction and expansion of medical images, the saving of bandwidth used for images transmission and also an optimisation of the adaptation of the received images to different types of terminals will have to be integrated.

5. Conclusion

Our work is based on a simple proposition: the distribution of our CBR system called EquiVox. We describe how a multi-agent paradigm has been used in order to design our distributed version of EquiVox. In this study, we demonstrated the performances of our system, considering different environments and constraints (number of known cases and speediness of arrivals).

In this article, we have introduced a new agent: the strategy agent. The strategy agent is in charge of the choice of the retaining strategy according to the entry rate fluctuation and thus activates and deactivates the required retainers with the chosen learning strategy. Moreover, we have created a group of retainers to be activated or deactivated depending on the strategy agent's policy.

We have shown that the strategy agent brings consequent improvement to our platform in terms of adaptability. Thus the system now reacts in function of the number of subjects arriving.

Acknowledgements

The authors wish to thank the *Institut National de la Santé et de la Recherche Médicale*, the *Ligue Contre le Cancer*, the *Pays de Montbéliard Agglomération* and the *Société Française de Radioprotection* for their financial aid and the *National Institute of Radiological Protection and Nuclear Safety* for their expertise. They also thank Mr. John Olsen for his help with the English language.

- [1] Henriet J, Chebel-Morello B, Salomon M, Farah J, Laurent R, Sauget M, Broggio D, Franck D, Makovicka L, EquiVox: an example of adaptation using an artificial neural network on a case-based reasoning platform, *Bio Eng: Appl, Bas Com* **25**:1350027, 2013.
- [2] Kolodner J, Case-based reasoning, *CA Morgan Kaufmann*, 1993.
- [3] Plaza E, McGinty L, Distributed case-based reasoning, *The Knowl Eng Rev* **20**:261, 2005.
- [4] Henriet J, Leni PE, Laurent R, Salomon M, Case-Based Reasoning adaptation of numerical representations of human organs by interpolation, *Exp Sys With Appl* **41**:260, 2014.
- [5] Henriet J, Chatonnay P, Introduction of a combination vector to optimise the interpolation of numerical phantoms, *Exp Sys With Appl* **40**:492, 2013.
- [6] Henriet J, Chatonnay P, Leni PE, An iterative precision vector to optimise the CBR adaptation of EquiVox, *Eng Appl Art Intell* **35**:158, 2014.

- [7] Henriet J, Lang C, Introduction of a multiagent paradigm to optimize a case-based reasoning system designed to personalize three-dimensional numerical representations of human organs, *Bio Eng: Appl, Bas Com* **26**:1450060, 2014.
- [8] Christensen GE, Deformable shape models for anatomy, *Wash Univ St Louis, Mississippi*, 1994.
- [9] Cover T, Hart P, Nearer neighbour pattern classification, *IEEE Trans Info Theo* **13**:21, 1967.
- [10] Clairand I, Bouchet LG, Ricard M, Durigon M, Di Paola M, Aubert B, Improvement of internal dose calculations using mathematical models of different adult heights, *Phy Med Bio* **45**:2771, 2000.
- [11] Garcia E, Guyennet H, Henriet J, Lapayre JC, Towards an optimistic management of concurrency: a probabilistic study of the pilgrim protocol, *Comp Supp Coop Work Des* **2**:51, 2006.
- [12] Diaz F, Fdez-Riverola F, Corchado J M, Gene-CBR: a case-based reasoning tool for cancer diagnosis using microarray data sets, *Comp Intell* **22**:254, 2006.
- [13] Montani S, Case-based reasoning for managing noncompliance with clinical guidelines, *Comp Intell* **25**:196, 2009.
- [14] Marling C, Montani S, Bichindaritz I, Funk P, Synergistic case-based reasoning in medical domains, *Exp Sys With Appl* **41**:249, 2014.
- [15] Moehrle S, Raskob F, Structuring and reusing knowledge from historical events for supporting nuclear emergency and remediation management, *Eng Appl Art Intell* **46**:303, 2015.
- [16] Perner P, Attig A, Using prototype-based classification for automatic knowledge acquisition, *Pat Rec Mach Intell Biomet*:197, 2011.
- [17] Attig A, Perner P, Incremental Learning of the Model for Watershed-Based Image Segmentation, *Comb Ima Anal*:209, 2012.
- [18] Dieterle S, Bergmann R, A Hybrid CBR-ANN Approach to the Appraisal of Internet Domain Names, *Case-Bas Reas Res Dev*:95, 2014.

- [19] Minsky M, A Neural-Analogue Calculator Based upon a Probability Model of Reinforcement, *Harv Univ Psy Lab*, 1952.
- [20] Pla A, Lopez B, Gay P, Pous C, eXiT* CBR. v2: Distributed case-based reasoning tool for medical prognosis, *Dec Supp Sys* **54**:1499, 2013.
- [21] Hu J, Qi J, Peng Y, New CBR adaptation method combining with problem-solution relational analysis for mechanical design, *Comp Ind* **66**:41, 2015.
- [22] Hawe GI Coates G, Wilson DT, Crouch RS, Agent-based simulation of emergency response to plan the allocation of resources for a hypothetical two-site major incident, *Eng Appl Art Intell* **46**:336, 2015.
- [23] Chebel-Morello B, Haouchine MK, Zerhouni N, Reutilization of diagnostic cases by adaptation of knowledge models, *Eng Appl Art Intell* **26**:2559, 2013.
- [24] Sebestyenova J, Case-based reasoning in agent-based decision support system, *Acta Poly Hung* **4**:127, 2007.
- [25] Faez F, Ghodsypour SH, O'brien C, Vendor selection and order allocation using an integrated fuzzy case-based reasoning and mathematical programming model, *Int J Prod Eco* **121**:395, 2009.
- [26] Al-Sakran H, Software cost estimation model based on integration of multi-agent and case-based reasoning, *J Comp Sci* **2**:276, 2006.
- [27] Kumar KA, Singh Y, Sanyal S, Hybrid approach using case-based reasoning and rule-based reasoning for domain independent clinical decision support in ICU, *Exp Sys With Appl* **36**:65, 2009.
- [28] Fu J, Fu Y, Case-based reasoning and multi-agents for cost collaborative management in supply chain, *Proc Eng* **29**:1088, 2012.
- [29] Floyd MW, Esfandiari B, Lam K, A Case-Based Reasoning Approach to Imitating RoboCup Players, *FLAIRS Conf*:251, 2008.
- [30] Rishi OP, Govil R, Sinha M, Agent based student modeling in distributed CBR based intelligent tutoring system, *Proc World Cong Eng Comp Sci*, 2007.

- [31] Ontanon S, Mishra K, Sugandh N, Ram A, Case-based planning and execution for real-time strategy games, *Case-Based Reas Res Dev*:164, 2007.
- [32] Ros R, Arcos JL, De Mantaras RL, Veloso M, A case-based approach for coordinated action selection in robot soccer, *Art Intell* **173**:1014, 2009.
- [33] Sandes EFO, Ralha CG, De Melo ACMA, An agent-based solution for dynamic multi-node wavefront balancing in biological sequence comparison, *Exp Sys With Appl* **41**:4929, 2014.
- [34] Amri H, Hanna F, Lapayre JC, Khalfallah A, Bouhlel MS, REPRO: A new reduction/expansion protocol to increase the performance of image transmission in medical telediagnosis platforms, *Bio Eng: Appl, Bas Com* **27**:1550054, 2015.

Highlights

EquiVox provides 3D representations of human bodies in very different contexts.

This multi-agent system uses case-base reasoning and artificial neural networks.

The training of artificial neural networks is a time-consuming process of EquiVox.

We explored different learning strategies depending on the context of use of EquiVox.

We designed a new version that can automatically change the learning strategy.

Figures and Tables

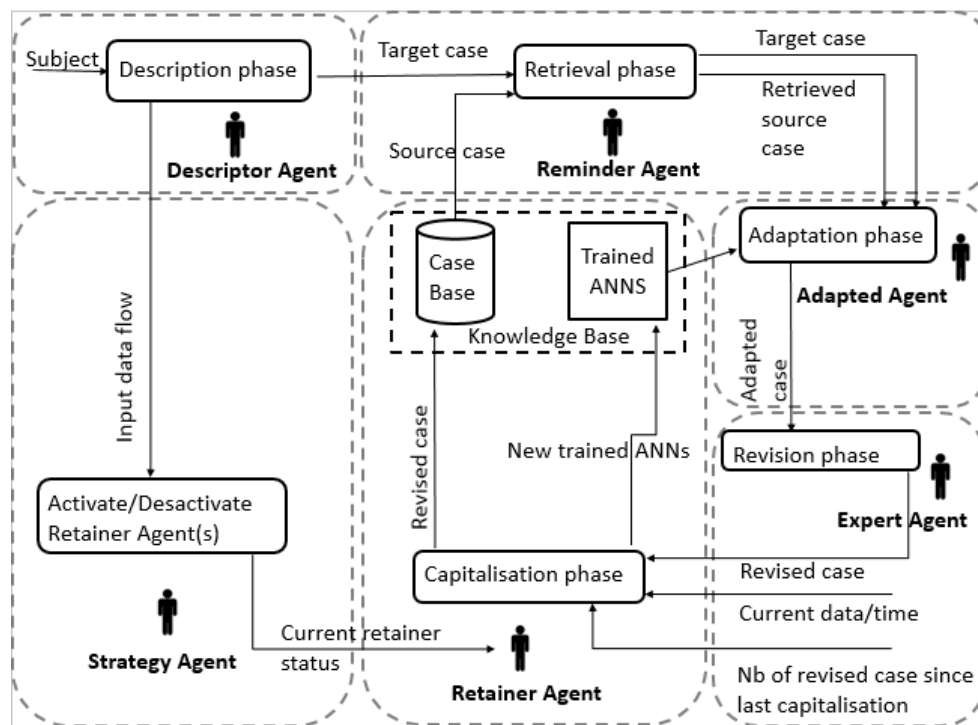


Figure 1: Data flow diagram over the EquiVox architecture.

6

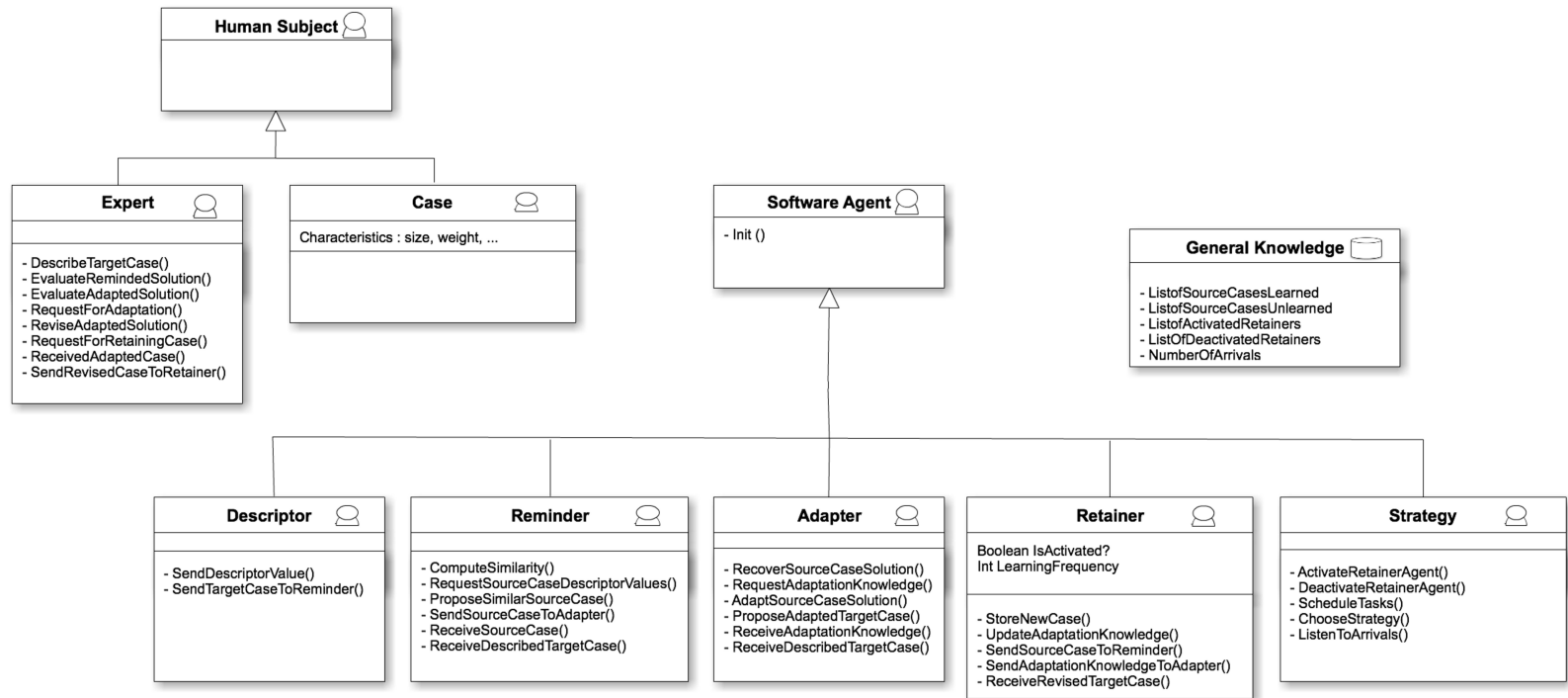


Figure 2: Distributed EquiVox AML Diagram.

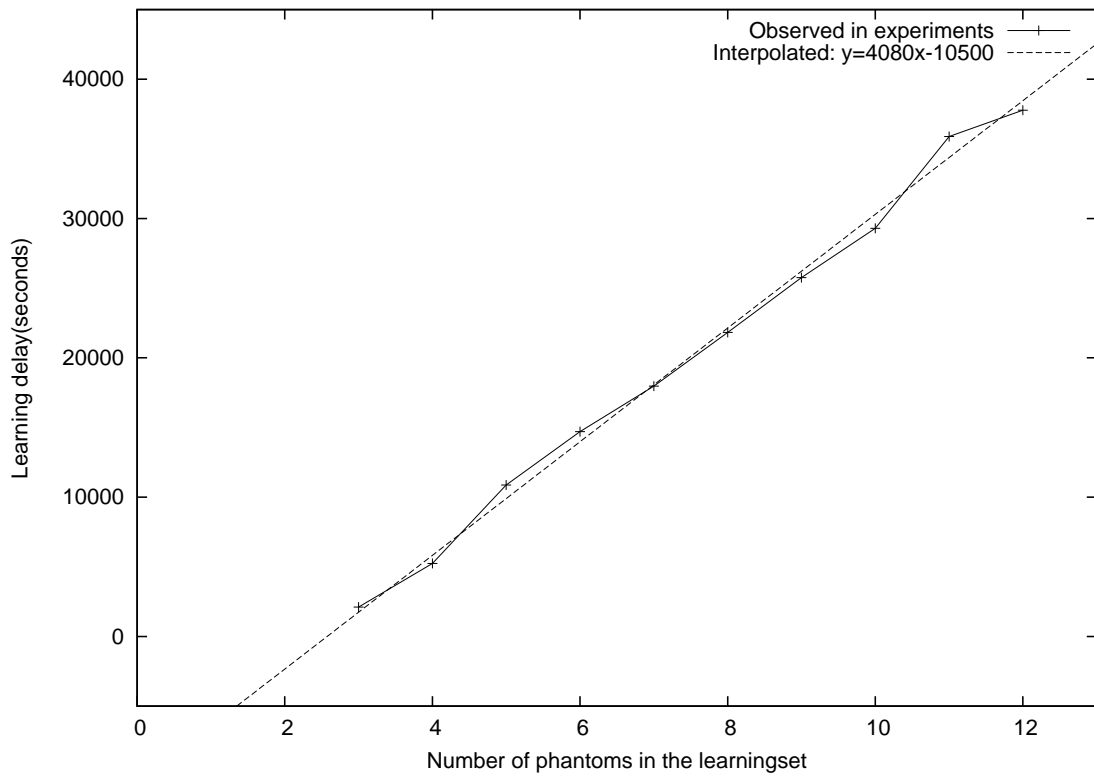


Figure 3: Learning delays observed according to the number of phantoms in the training set.

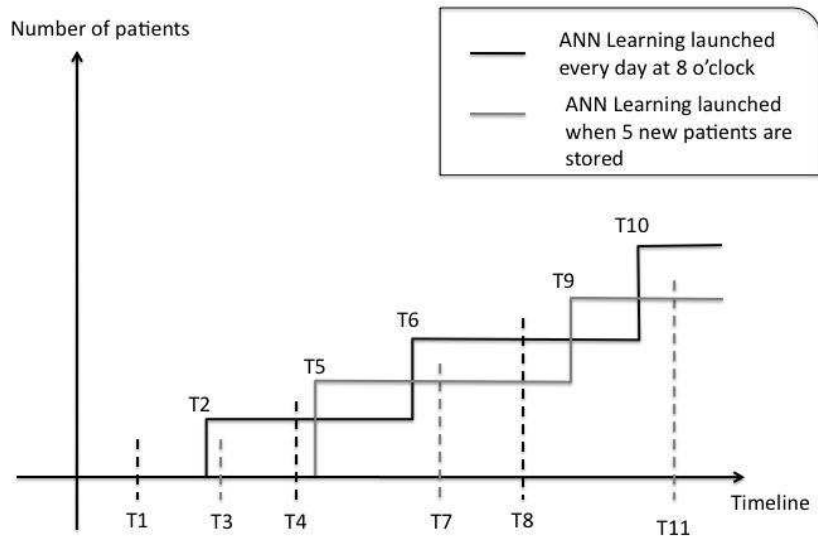


Figure 4: Comparison of strategies for updating adaptation knowledge.

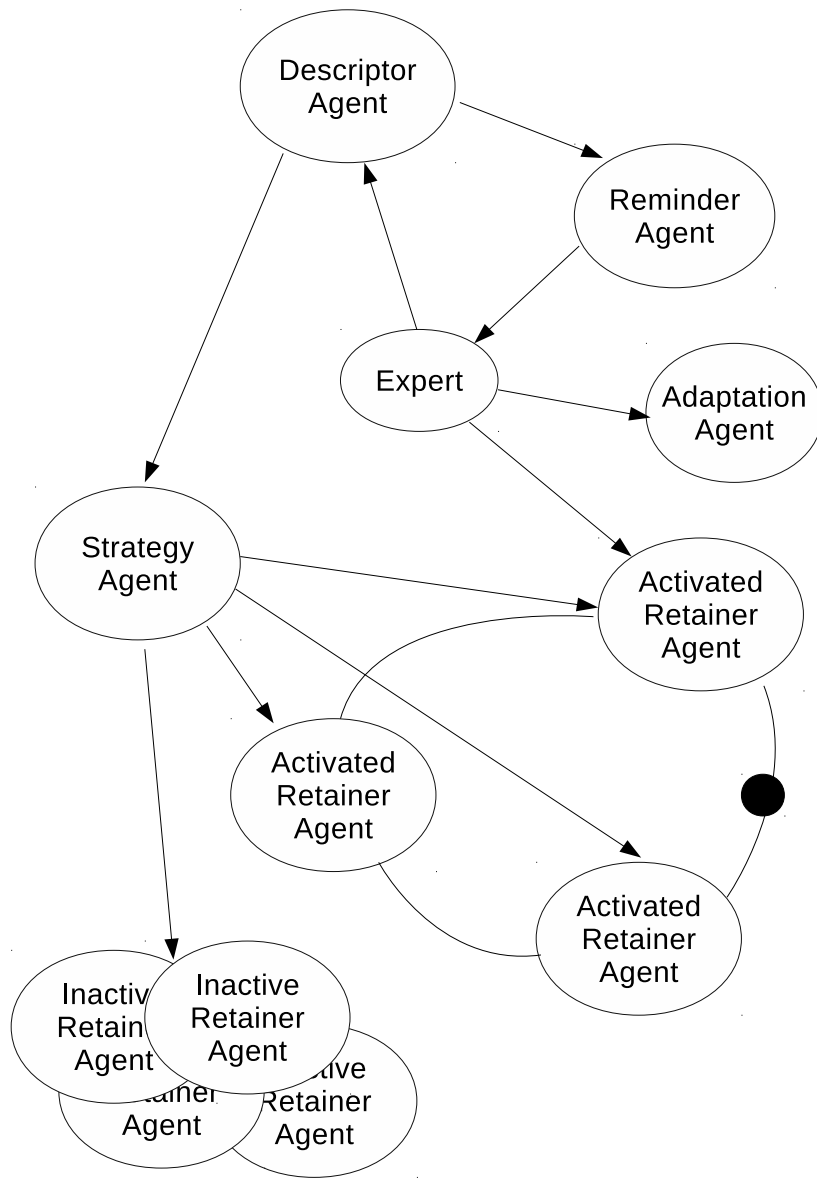


Figure 5: Communication between the agents of EquiVox.

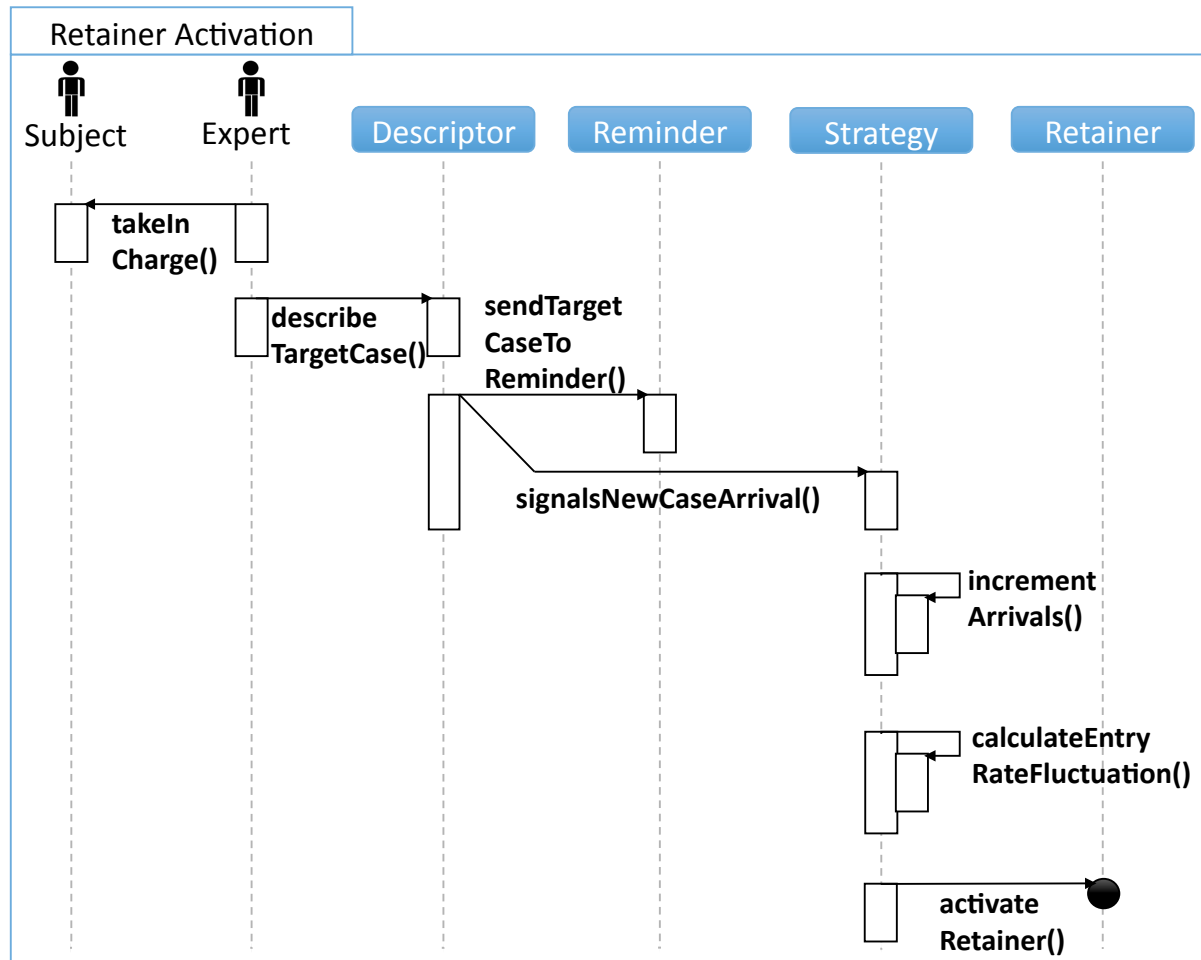


Figure 6: Activation of a retainer by the EquiVox strategy agent.

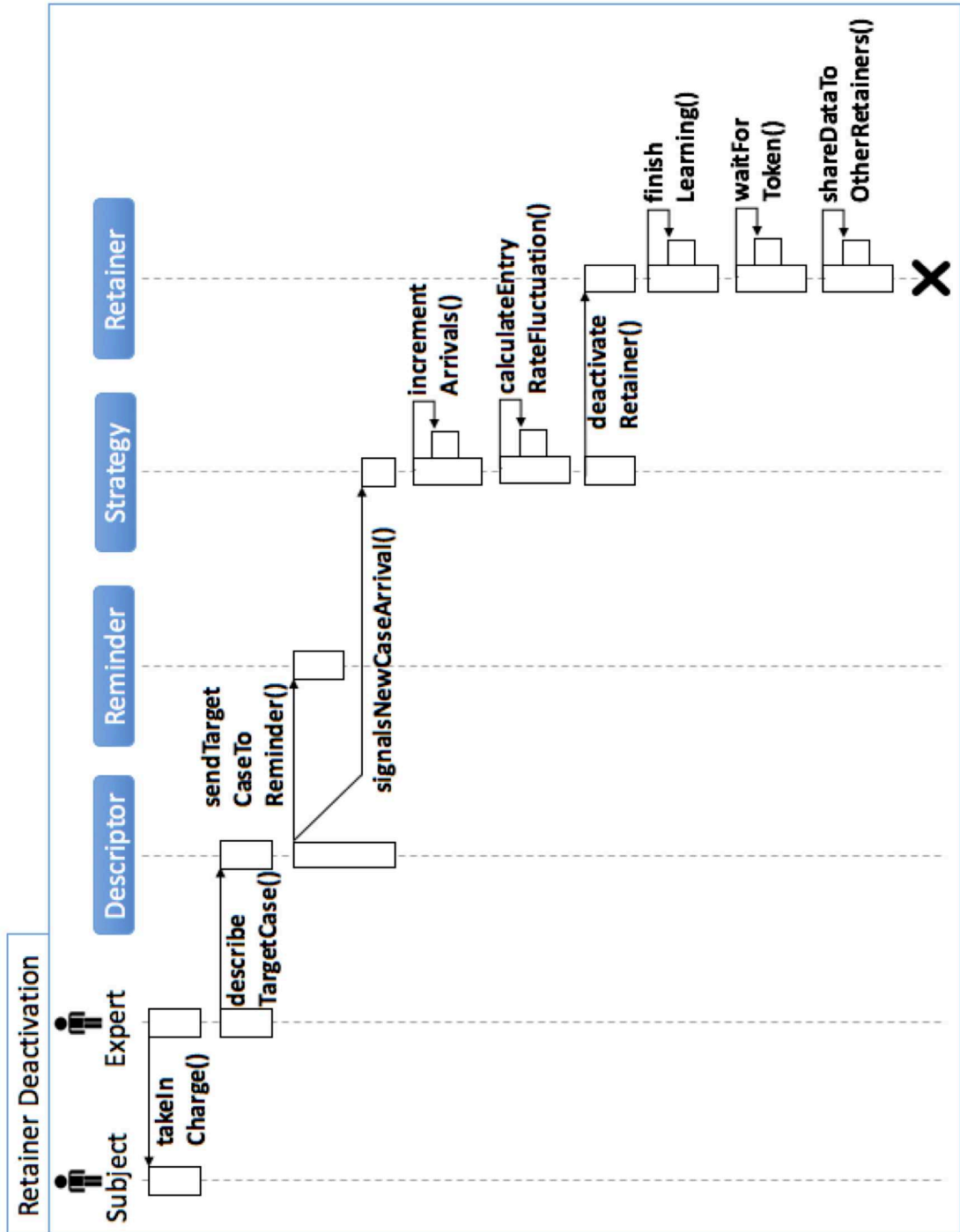


Figure 7: Deactivation of a retainer by the EquiVox strategy agent.

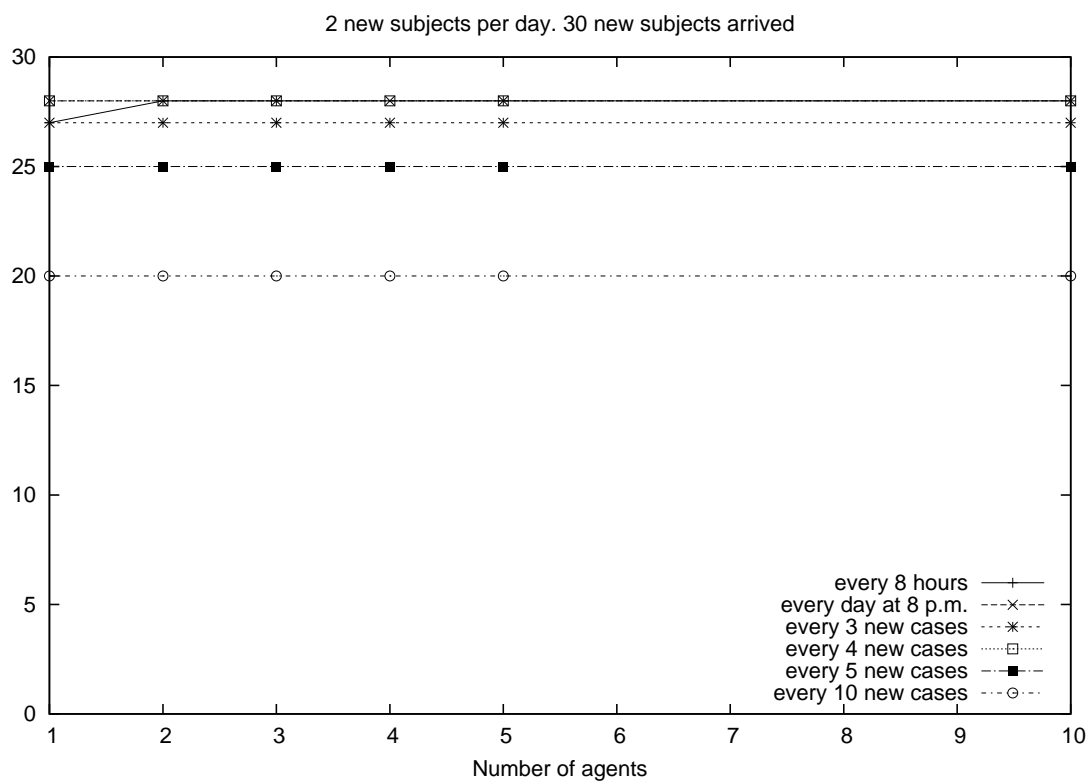


Figure 8: Number of phantoms taken into account during the adaptation knowledge in the customary context of hospitals (2 subjects per day).

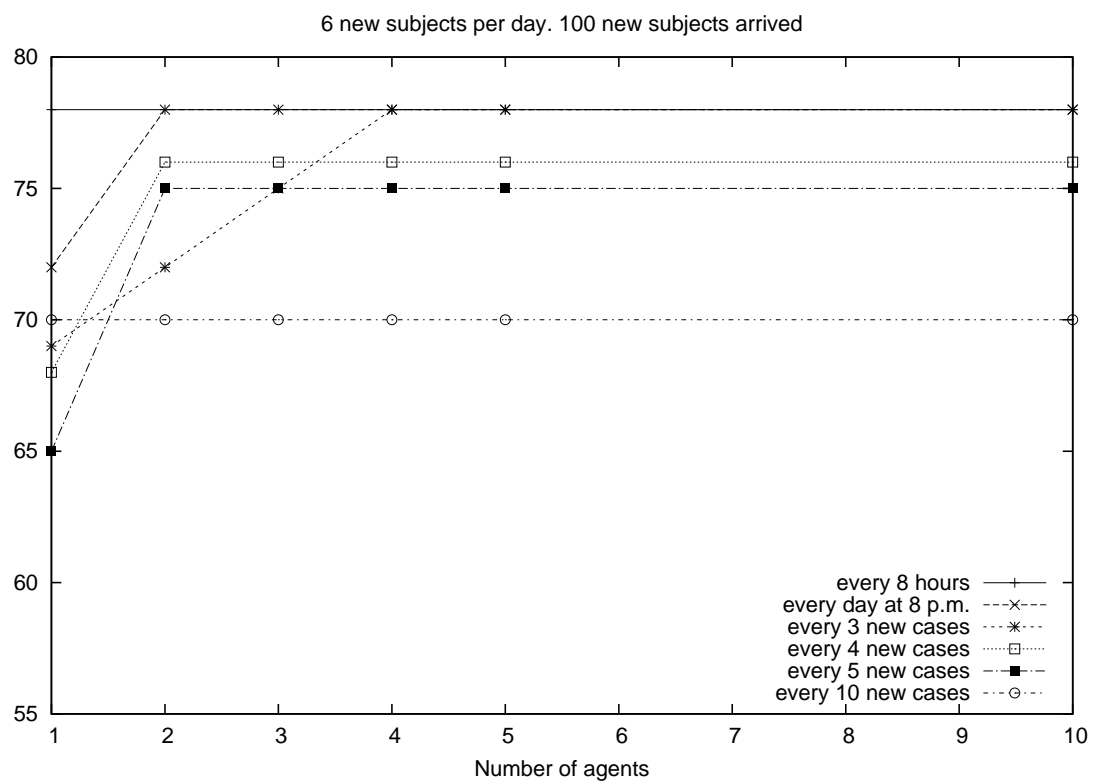


Figure 9: Number of phantoms taken into account during the adaptation knowledge in the customary context of hospitals (6 subjects per day).

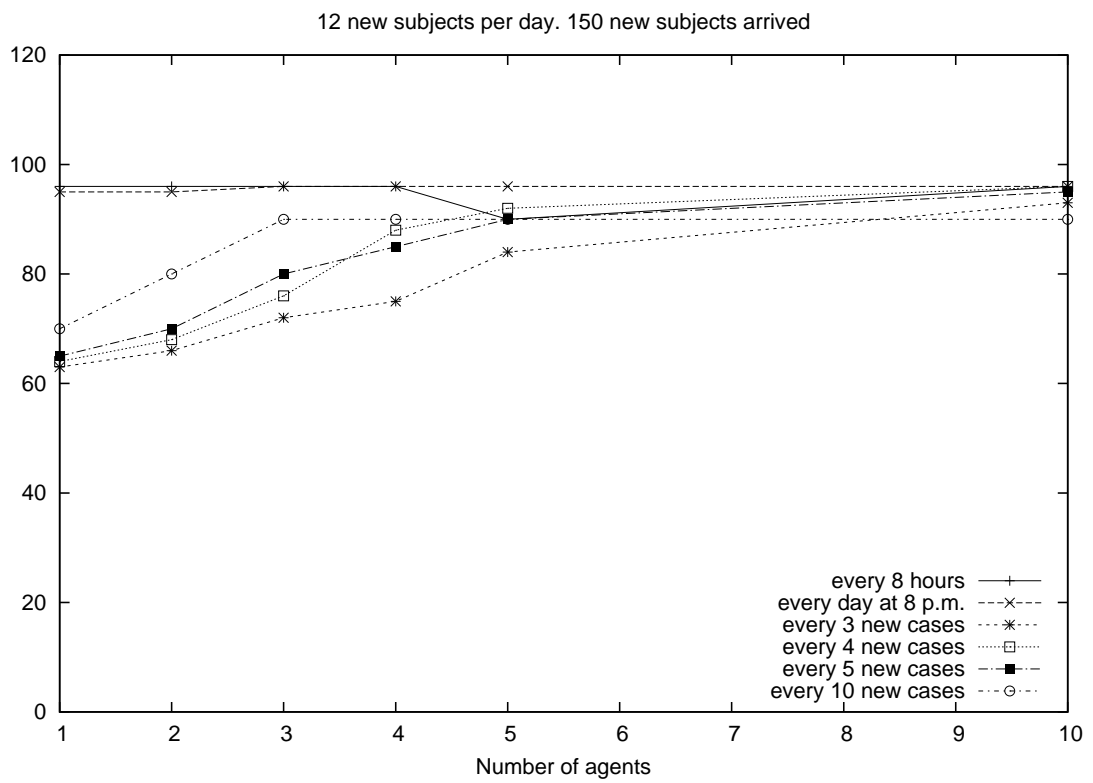


Figure 10: Number of phantoms taken into account during the adaptation knowledge in the customary context of hospitals (12 subjects per day).

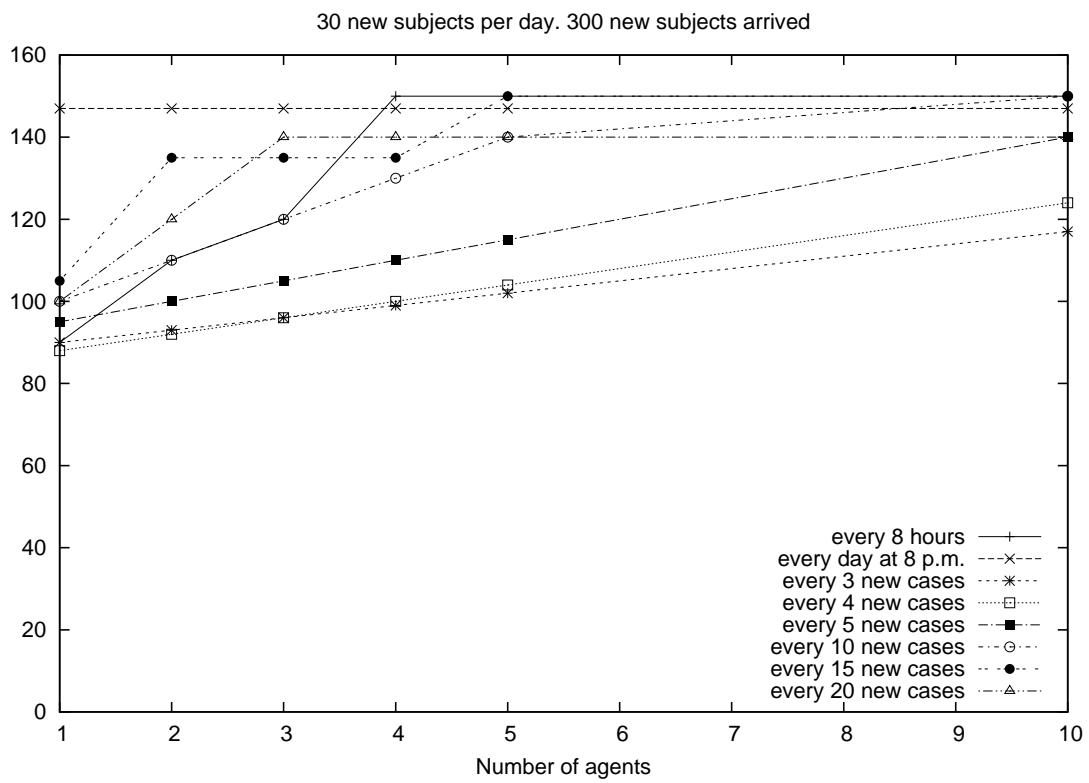


Figure 11: Number of phantoms taken into account during the adaptation knowledge in disaster contexts (30 subjects per day).

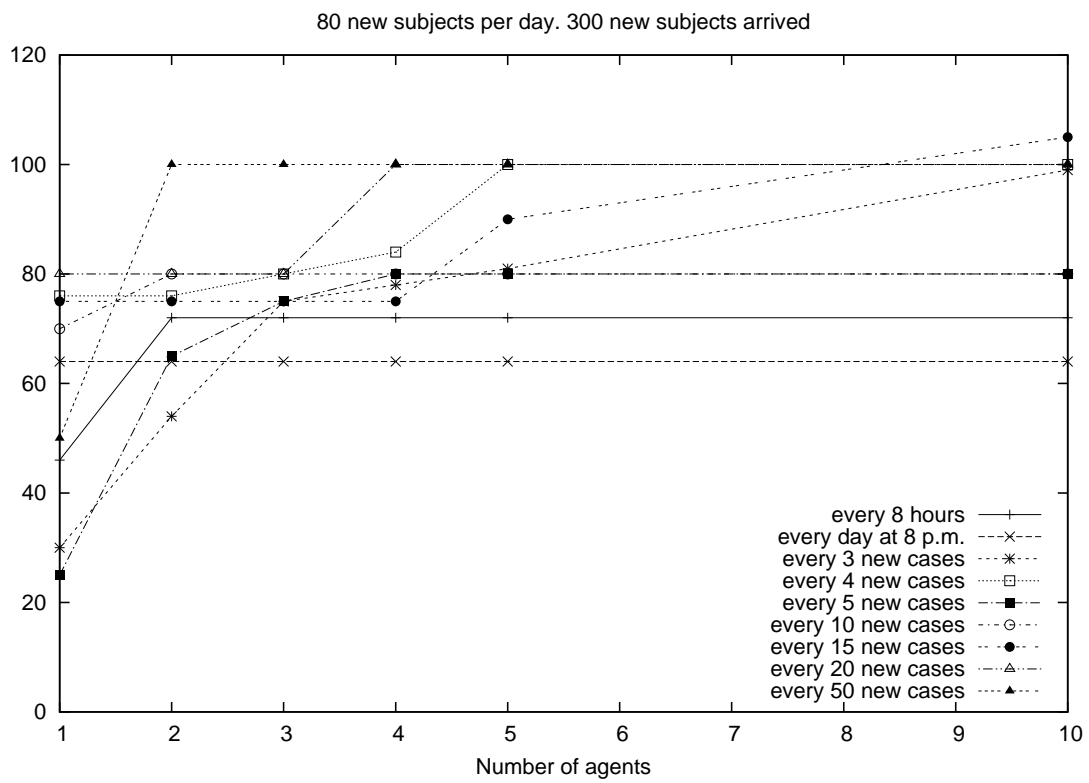


Figure 12: Number of phantoms taken into account during the adaptation knowledge in disaster contexts (80 subjects per day).

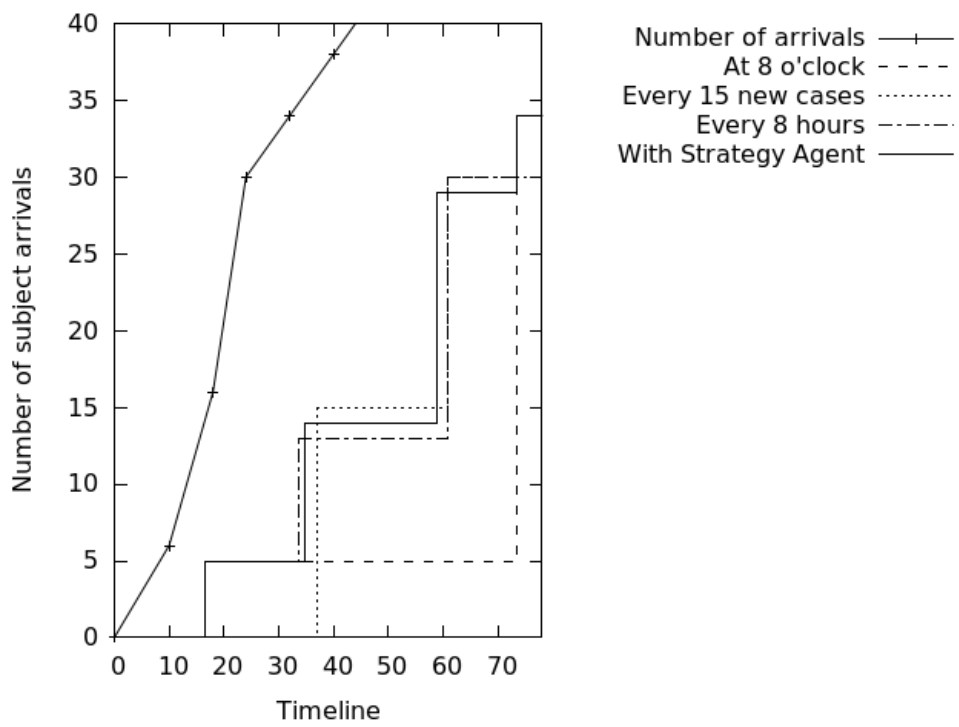


Figure 13: Number of subjects taken into account by the EquiVox adaptation module in scenario 1.

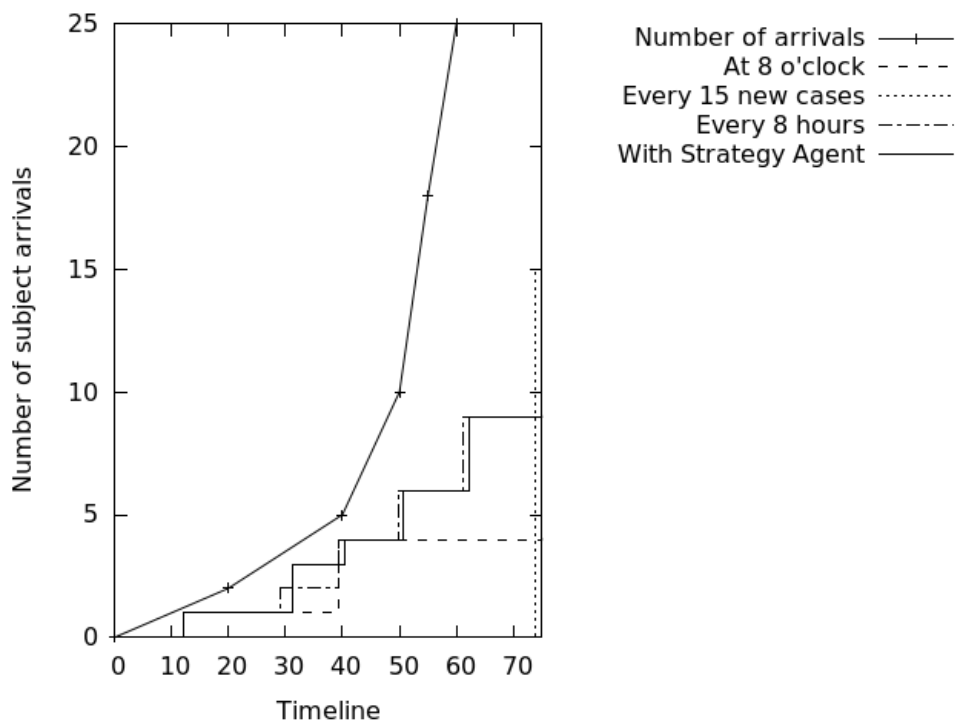


Figure 14: Number of subjects taken into account by the EquiVox adaptation module in scenario 2.

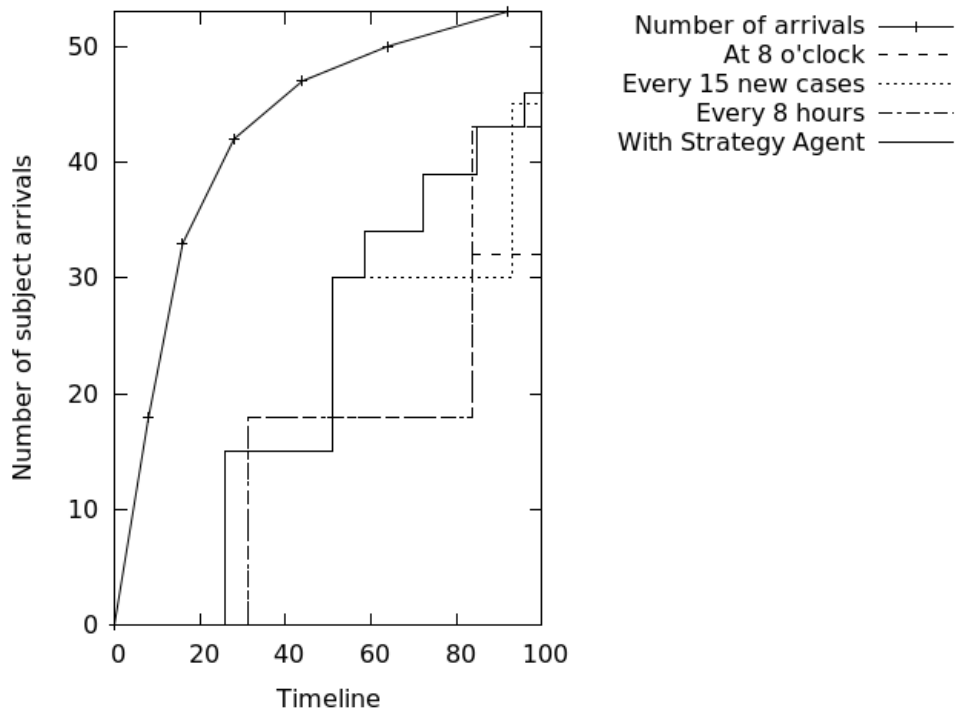


Figure 15: Number of subjects taken into account by the EquiVox adaptation module in scenario 3.

Phantom ID	height (cm)	age (years)	gender	under chest circumference (cm)	chest circumference (cm)
P_2	167.54	48	Female	85.6	99.9
P_4	173.18	26	Female	74.7	90.7
P_9	185.25	31	Male	80.7	88.3

Table 1: Examples of phantoms initially stored in the EquiVox case base.

Arrival frequencies (subjects per day)	Number of activated RtAs	ANN training frequencies
less than 4	2	every day at 8 p.m.
4 to 9	4	every 8 hours
9 to 40	5	every 8 hours
more than 40	10	every 15 new cases

Table 2: Implemented adaptation strategies.