

Elsevier Editorial System(tm) for Expert Systems With Applications  
Manuscript Draft

Manuscript Number:

Title: Integrating Case-based Planning and RPTW Neural Networks to Construct an Intelligent Environment for Health Care

Article Type: Full Length Article





Keywords: case-based planning; case-based reasoning; RFID; health care; RPTW neural network

Corresponding Author: Dr. Javier Bajo, Ph.D.

Corresponding Author's Institution: Pontifical University of Salamanca

First Author: Javier Bajo, Ph.D.

Order of Authors: Javier Bajo, Ph.D.; Juan F de Paz, Engineering; Yanira R de Paz, Ph.D.; Juan M Corchado, Ph.D.

	<p><b>Javier Bajo</b> (PhD.). Received a PhD. in Computer Science and Artificial Intelligence from the University of Salamanca in 2007. At present he is Assistant Professor at the University of Salamanca (Spain). He obtained an Information Technology degree at the University of Valladolid (Spain) in 2001 and an Engineering in Computer Sciences degree at the Pontifical University of Salamanca in 2003. He has been member of the organising and scientific committee of several international symposiums such as CAEPIA, IDEAL, HAIS, etc. and co-author of papers published in recognized journal, workshops and symposiums.</p>
	<p><b>Juan Francisco de Paz</b> (PhD. Student). At this moment, he's completing his studies of PhD in Computer Science at University of Salamanca (Spain). He obtained a Technical Engineering in Systems Computer Sciences degree in 2003, an Engineering in Computer Sciences degree in 2005 at the University of Salamanca and at this moment he is finishing Statistic in the same university. He has been co-author of published papers in several journals.</p>
	<p><b>Yanira de Paz</b> (PhD. Student). At present she holds a scholarship provided by the Spanish Minister of Education to complete a PhD. program at the University of Salamanca (Spain). She obtained a Mathematics degree in 2002 and a Statistic degree in 2003 at the University of Salamanca (Spain). She is Assistant Professor at the Faculty of Economy at the University of Salamanca and co-author of several mathematical and statistical books. She has also been lecturer in the Faculty of Mathematics at the Complutense University of Madrid.</p>
	<p><b>Juan M. Corchado</b> (PhD.). Received a PhD. in Computer Science from the University of Salamanca in 1998 and a PhD. in Artificial Intelligence (AI) from the University of Paisley, Glasgow (UK) in 2000. At present he is Associate Professor, Director of the Intelligent Information System Group (<a href="http://bisite.usal.es">http://bisite.usal.es</a>) and Director of the MSc programs in Computer Science at the University of Salamanca (Spain), previously he was sub-director of the Computer Science School at the University of Vigo (Spain, 1999-00) and Researcher at the University of Paisley (UK, 1995-98). He has been a research collaborator with the Plymouth Marine Laboratory (UK) since 1993. He has led several Artificial Intelligence research projects sponsored by Spanish and European public and private institutions and has supervised seven PhD. students. He is the co-author of over 130 books, book chapters, journal papers, technical reports, etc. published by organisations such as Elsevier, IEEE, IEE, ACM, AAAI, Springer Verlag, Morgan Kaufmann, etc., most of these present practical and theoretical achievements of hybrid AI and distributed systems. He has been President of the organising and scientific committee of several international symposiums.</p>

## Integrating Case-based Planning and RPTW Neural Networks to Construct an Intelligent Environment for Health Care

Javier Bajo<sup>1</sup>, Juan F. de Paz<sup>2</sup>, Yanira de Paz<sup>2</sup> and Juan M. Corchado<sup>2</sup>,

<sup>1</sup> Escuela Universitaria de Informática. Universidad Pontificia de Salamanca  
Compañía 5, 37002, Salamanca, Spain  
jbajope@upsa.es

<sup>2</sup> Departamento Informática y Automática. Universidad de Salamanca  
Plaza de la Merced s/n, 37008, Salamanca, Spain  
{fcofds;yanira;corchado}@usal.es

### Abstract

*This paper presents an intelligent environment developed for monitoring patients' health care in execution time in hospital environments. The CBPMP (Case-Based Planner for monitoring patients) is an autonomous deliberative case-based planner designed to plan the nurses' working time dynamically, to maintain the standard working reports about the nurses' activities, and to guarantee that the patients assigned to the nurses are given the right care. The planner operates in wireless devices and is integrated with complementary software into an intelligent environment, named AmI-P (Ambient Intelligence for Patients). CBPMP description, its relationship with the complementary technology, and preliminary results of system prototype in a real environment are presented.*

*Keywords:* case-based planning; case-based reasoning; RFID; health care; RPTW neural network

### 1. Introduction

Ambient Intelligence (AmI) proposes a new way to interact between people and technology, where this last one is adapted to individuals and their context [14]. The objective of Ambient Intelligence is to develop intelligent and intuitive systems and interfaces capable to recognize and respond to the user's necessities in a ubiquitous way, providing capabilities for ubiquitous computation and communication, considering people in the centre of the development, and creating technologically

complex environments in medical, domestic, academic, etc. fields [29]. Ambient Intelligence requires new ways for developing intelligent and intuitive systems and interfaces, capable to recognize and respond to the user's necessities in a ubiquitous way, providing capabilities for ubiquitous computation and communication [2]. The multi-agent systems [33] have become increasingly relevant for developing distributed and dynamic intelligent environments. A case-based reasoning system [1] has been embedded within a deliberative agent and allows it to respond to events, to take the initiative according to its goals, to communicate with other agents, to interact with users, and to make use of past experiences to find the best plans to achieve goals. The deliberative agent works with the concepts of Belief, Desire, Intention (BDI) [5], and has learning and adaptation capabilities, which facilitates its work in dynamic environment.

With the appearance of AmI-based systems, one of the most benefited segments of population will be the elderly and people with disabilities. It will improve important aspects of their life, especially health care [12]. There is an ever growing need to supply constant care and support to the disabled and elderly [27] and the drive to find more effective ways to provide such care has become a major challenge for the scientific community. Today, the number of Europeans over 60 years represents more than 25% of the population and it is estimated that in 20 years this percentage will rise to one third of the population [6]. In the United States of America it is expected that in 2020 people over 60 will represent about 1 of 6 citizens [20]. Furthermore, over 20% of people over 85 years have a limited capacity for independent living, requiring continuous monitoring and daily care. The Institute of Medicine has studied the role of information technology in improving health care delivery in the US. In [20], the Institute presents a strategy and an action plan to foster innovation and improve the delivery of care. The need to reinvest in the system is underlined and as such six health care aims are defined; to be safe, effective, patient-centered, timely, efficient and equitable. Ten guidelines for the redesign of the system are given attention on the role of the patient and improvements in knowledge, communication and safety mechanisms. Moreover, the Institute proposes a strategy to improve safety

in health care based on the study of medical errors. The proposed system presented here has been conceived and developed taking these considerations into account.

The importance of developing new and more reliable ways to provide care and support to the elderly is underlined by this trend [6], and the creation of secure, unobtrusive and adaptable environments for monitoring and optimizing health care will become vital. Some authors [27] consider that tomorrow health care institutions will be equipped with intelligent systems capable of interacting with humans. Multi-agent systems and architectures based on intelligence devices have recently been explored as supervision systems for medical care for the elderly [9]. These intelligence systems aim to support them in all aspects of their daily life, predicting potential hazardous situations and delivering physical and cognitive support.

Radio Frequency IDentification (RFID) [30] is an automated data-capture technology that can be used to electronically identify, track, and store information about products, items, components or people. It is most frequently used in industrial/manufacturing, transportation, distribution, and warehousing industries, however, there are other growth sectors including health care. The proposed system uses microchips mounted on bracelets worn on the patient's wrist or ankle, and sensors installed over protected zones, with an adjustable capture range up to 2 meters. The microchips or transponders help locate the patients, which can be ascertained by consulting the CBR agents installed in personnel PDAs.

Nowadays, a great amount of problems can be considered as planning problems. In this kind of problems the aim is to obtain a plan, as well as obtaining a scheduling of the actions along the time, assigning the necessary resources and taking into account the restrictions within the problems (possibly using optimality criteria and guarantee of the execution). Planning has experienced significant progresses since its origins in the decade of 1970 in terms of efficiency and sophistication of the algorithms and representations used. Moreover, planning has been adapted to real problems and talking about practical planning. Practical planning has been applied in many different fields and has

been combined with different artificial intelligence techniques, i.e. neural Networks. One of the main challenges has been the development of intelligent systems focused on the improvement of health care sciences that can be applied to hospitals, residences, etc. [13]. Most of the applications developed try to use practical planning techniques in order to monitor patients, supervise treatments or to obtain knowledge from the data handled. Lanzola *et al.* [23], for instance, presents a methodology that facilitates medical applications and proposes a generic computational model for its implementation. Such a model could be specialized to manage all kind of information and knowledge in a hospital environment. However, the method proposed by Lanzola *et al.* [23] is very abstract and doesn't take into account the possibility of modelling certain technologies common in hospital environments (as shown in this paper) as wireless or RFID. Another lack of the method is the use of dynamic planning, commonly required in this kind of applications. Others such as Decker and Li [10], propose a system to increase hospital efficiency using global planning and scheduling techniques. They propose a multi-agent solution using the generalized partial global planning approach that preserves the existing human organization and authority structures, while providing better system-level performance (increased hospital unit throughput and decreased patient stay time). To do this, they extend the proposed planning method with a coordination mechanism to handle mutually exclusive resource relationships, using resource constraint scheduling. This system does not use dynamic planning, it uses a static task assignment, and it does not work on wireless devices and does not use location information or RFID technology.

In this work, we propose CBPMP, a dynamic planner which is integrated within AmI-P intelligent environment, which is a dynamic system for the management of different aspects of a geriatric center. This distributed system uses Radio Frequency Identification (RFID) [30] technology for ascertaining patients' location in order to maximize their safety or to generate medical staff plans. The development of such intelligent environment has been motivated for one of the more distinctive characteristics of hospital environment, which is their dynamism, in the sense that the patients change

very frequently (new patients arrive and others pass away), while the staff rotation is also relatively high and they normally work in shifts of eight hours. AmI-P provides the personnel of the hospital with updated information about the center and the patients, provides the working plan, information about alarms or potential problems and keeps track of their movements and actions within the center. Dynamic problems require the dynamic solutions provided by this technology. From the user's point of view the complexity of the solution has been reduced with the help of friendly user interfaces and a robust and easy to use technology.

The proposed planner CBPMP uses a case-based reasoning (CBR) [1] architecture, that allows it to respond to events, to take the initiative according to its goals, to communicate with other agents, to interact with users, and to make use of past experiences to find the best plans to achieve goals. This particular planner uses a special type of CBR systems which we call Case-Base Planning (CBP) system, specially designed for planning construction. Case-based planning allows us to retrieve past experiences when a new plan is created which lends the system a large capacity for learning and adaptation [15]. The planning mechanism has been implemented by means of a novel RPTW (Routing Problems with Time Windows) neural network. The neuronal networks proposed within this research framework are self-organised, based on Kohonen networks [21], but which present certain improvements (RPTW neural Neural Network) [25]. These improvements allow the network to reach a solution much more rapidly. Furthermore, once a solution has been reached, it makes it possible to make new modifications taking restrictions into account (in this study, specifically time restrictions). In this way, the CBPMP planner has learning and adaptation capabilities, which facilitates its work in dynamic environment and allows it to learn from initial knowledge, to interact autonomously with the environment as well as with users and other agents within the system, and to have a large capacity for adaptation to the needs of its surroundings.

This chapter is organized as follows: the following section presents the new RPTW neural network. Section 3 shows the new planning mechanism, obtained by means of integrating case-based planning

and RPTW neural networks. Section 4 presents a case study in a hospital environments and, finally, Section 5 presents results and conclusions obtained after the implementation of a prototype into a real scenario.

## **2. Defining Self-Organising Neural Networks in a Novel Way: RPTW Neural Networks**

The basic Kohonen network [21] cannot be used to resolve dynamic problems since it attempts to minimize distances without taking into account any other type of restriction, such as time limits. In the present study a planner based on Kohonen networks is described but with a number of improvements (RPTW Neural Network) [25] that allow us to reach a solution more rapidly. Furthermore, once a solution has been reached, it is re-modified in order to take restrictions into account.

### **2.1 Objective 1: Reach a Solution more Rapidly**

As such, for this modification of the basic algorithm (RPTW), the aim is to make the solution search more agile and in order to achieve this, the basic vicinity function used in the Kohonen network is modified. The number of neurons in the output layer corresponds to the places that the user (or traveller) wishes to visit. The topology of the neural network being considered is described below. The input layer is formed by two neurons, each one of those receives the coordinates of the place presented as input. A vector of neurons is used of size the same as the number of places to visit in the problem in the output layer, as shown in [19], [24]. The number of neurons in the output layer isn't modified. Let  $x_i \equiv (x_{i1}, x_{i2}) \quad i = 1, \dots, N$  the coordinates from the place  $i$  and  $n_i \equiv (n_{i1}, n_{i2}) \quad i = 1, \dots, N$  the coordinates of the neurons  $i$  in  $\mathfrak{R}^2$ .  $N$  places will be visited by the traveller. Consequently there will be: Two neurons in the input layer and  $N$  neurons in the output layer. It will be considered a vicinity function decreasing with the number of iterations.



$$g(k, h, t) = \text{Exp} \left[ \left( -\frac{|k-h|}{N/2} \right)^{\frac{\text{Máx}_{\substack{i,j \in \{1, \dots, N\} \\ i \neq j}} \{f_{ij}\}} - \sqrt{(n_{k1} - n_{h1})^2 + (n_{k2} - n_{h2})^2}}{\text{Máx}_{\substack{i,j \\ i \neq j}} \{f_{ij}\}} - \lambda \frac{|k-h|t}{\beta N} \right] \quad (1)$$

$\lambda$  and  $\beta$  are determined empirically, their respective values are: 5 and 50.  $t$  is the current iteration.  $\text{Exp}[x] = e^x$ .  $N$  is the number of places that are visited by a traveller and  $f_{ij}$  is the distance given by the Floyd Algorithm [26].

The radius of final vicinity should be near to 0 so that only the winner is upgraded. Iteratively the group of places will be presented, so that the weights of the neurons approach the coordinates of the places. When concluding the process, there will be a neuron associated to each place. To determine the route to follow, we will leave the place associated to the neuron  $i$  to the associated to  $i+1$ , for  $i=1, 2, \dots, N$ , passing the whole vector of neurons. To close the road, the last tract will be given by the route that joins the place associated to the neuron  $N$  with the associated to the neuron  $1$ . The distance of the road will be given by the sum of the distances between the successive couples of rooms of the road. The learning rate is a decreasing function as shown in (2):

$$\eta(t) = \text{Exp} \left[ -\sqrt[4]{\frac{t}{\beta N}} \right] \quad (2)$$

The activation function of the neurons corresponds to the identity function. When the system stops, the route to continue will be given by the weights of the neurons that will be very close to the coordinates of the places. To know which following place to visit, we need to study the following element of the vector of neurons. The neurons are stored in a vector that contains the weights of each neuron at present, so that the vector defines a ring, the neuron  $n_1$  is the following to the  $n_N$  considered.

With a big radius of vicinity, in the first iterations of the algorithm the victory of a neuron affects a great part of the map, so that a global self-organization takes place. If the radius decreases, the effect of a victory affects every time a smaller part of the map, so that the criterion to stop the learning of the network is that the distance among places cannot be optimized more. The initial number of total iterations is of  $T_1 = \beta N$  (first phase). When  $t = \beta N$ , all the couples of possible neurons are exchanged (exchanging their weights) in the obtained ring of neurons and if the distance is optimized then the learning is not finished.

In general, in the phase  $Z$ , the total number of interactions to carry is:

$$T_Z = T_{Z-1} - \frac{T_{Z-1}}{Z} \quad (3)$$

The aim of these phases is to eliminate the crossings. Concluded the iterations of each phase is proven if the distance is already optimized, in such a way that in the phase that stops the learning, the distance is minimum. The primary objective - to achieve artificial neural networks that are faster than basic Kohonen networks, applied to the problems that the basic networks resolve - has been achieved. In the section below the necessary modifications are introduced into the algorithm so that the network can take restrictions into account, and therefore be able to resolve other problems that cannot be resolved by basic Kohonen networks.

## 2.2 Objective 2: Taking Time Restrictions into Account

Instead of using the distance calculated by means of the Floyd algorithm [26], a different distance is used that we call “temporal distance”. Without loosing any generality it can be supposed that a distance unit is equivalent to a time unit.

$$dt_{ij} \equiv dt(x_i, x_j) \triangleq \text{Máx} \{ f_{ij} + t_i, b_j \} \quad (4)$$

Where  $t_i$  is the time hended to get to place “ $i$ ” from the previous place plus the time taken on tasks to be carried out within place “ $i$ ” (in other words, the service time in “ $i$ ”) and  $b_j$  is the time limit for carrying out the tasks in place “ $j$ ”. In this way, the vicinity function of the network modified from the RPTW network is:

$$g(k, h, t) = Exp \left[ \left( -\frac{|k-h|}{N/2} \right)^{\frac{\underset{i \neq j}{\text{Máx}} \{d_{ij}^*\} - \sqrt{(n_{k1} - n_{h1})^2 + (n_{k2} - n_{h2})^2}}{\underset{i, j}{\text{Máx}} \{d_{ij}^*\}} - \lambda \frac{|k-h|t}{\beta N}} \right] \quad (5)$$

With:

$$d_{ij}^* \equiv d^*(x_i, x_j) \triangleq \frac{f_{ij} + t_i}{dt_{ij}} \quad (6)$$

### 3. RTPW Neural Network and Case-based Planning.

CBPMP is an autonomous deliberative case-based planner developed for integration within a system named Aml-P. The goal of this planner is to provide efficient working schedules, in execution time, for hospital environments staff and therefore to improve the quality of health care and the supervision of patients in a hospital. Each of the CBPMP planners is assigned to a nurse or a doctor of a hospital, and it also provides information about patients’ locations, historical data and alarms. As the members of the staff are carrying out their duties (following the plan provided by the planner) the initial proposed plan may need to be modified, for example due to delays or alarms. In this case the planner is capable of re-planning in execution time.

Case-based Reasoning (CBR) is a type of reasoning based on the use of past experiences to solve new problems [1]. The purpose of a CBR system is to solve new problems by adapting solutions that have been used to solve similar problems in the past. A case can be defined as a past experience, and is composed of three elements: A problem description which describes the initial problem, a solution

which provides the sequence of actions carried out in order to solve the problem, and the final state which describes the state achieved once the solution was applied. A CBR system manages cases (past experiences) to solve new problems. The way in which cases are managed is known as the CBR cycle. The CBR cycle shown in Figure 1 consists of four sequential phases: retrieve, reuse, revise and retain. The retrieve phase starts when a new problem description is received. Similarity algorithms are applied in order to retrieve from the cases memory the cases with a problem description more similar to the current one. Once the most similar cases have been retrieved, the reuse phase begins. In this phase the solutions of the cases retrieved are adapted to obtain the best solution for the current case. The revise phase consists of an expert revision of the solution proposed. Finally, the retain phase allows the system to learn from the experiences obtained in the three previous phases and updates the memory case in consequence.

AmI proposes new possibilities for solving a wide scope of problems and a new way of interaction between people and technology, where the last one is adapted to individuals and their context, showing a vision where people are surrounded by intelligent interfaces merged in daily life objects [12]. This adaptation requires creating computing-capable environments with intelligent processing and communication services. One of the possibilities for executing complex computational tasks is the use of CBR systems, which are capable of learning from their initial knowledge and autonomously adapting themselves to environmental changes. As CBR systems use their experiences from previous similar situations to resolve a new problem, it will be possible to personalize a user access if we take into account similar past cases. These similar past cases can contain memories about users with similar profiles trying to resolve a problem in similar context conditions.

The CBR adaptation and learning mechanism needs updated information from the environment and communication paths, with both the system users and the system elements. The multiagent technology [33] can be used for solving distributed systems in which CBR systems may act. In this paper it is proposed that the use of agents reason and solve problems using embedded CBR systems [8].

Case-based planning (CBP) is the idea of planning as remembering [7], [16]. CBP is a specialization of case-based reasoning (CBR) which is a problem solving methodology based on using a library of solutions for similar problems [16]. In CBP, the solution proposed to solve a given problem is a plan, so this solution is generated taking into account the plans applied to solve similar problems in the past. The problems and their corresponding plans are stored in a plans memory. In practice, what is stored is not only a specific problem with a specific solution, but also additional information about how the plans have been derived. As well as in case-based reasoning, the case representation, the plans memory organization and the algorithms used in every stage of the case-based planning cycle are essential to define an efficient planner. The formal description of a case-based planner [28] can be formalized as a 3-tuple  $\langle I, G, Op \rangle$ :

- $I$  is a set of formulae describing the initial state.
- $G$  is a set of formulae describing the goal specification.
- $Op$  is the set of operators (also called actions) that can be applied in a plan. Every action  $a \in Op$  is described in terms of pre-conditions  $Ca$  (what has to be fulfilled in order to the action can be executed) and port-conditions  $Ea$  (what has to be fulfilled after the execution of the action).

A plan  $P$  is a tuple  $\langle S, B, O, L \rangle$ :

- $S$  is the set of plan actions. There are two special actions:  $t_I$ , those whose effects are  $I$ , that is, the initial state; and  $t_G$ , those actions whose pre-conditions are  $G$ , that is, the goal specification.
- $O$  is an ordering relation on  $S$  allowing to establish an order between the plan actions.  $t_I$  is always the first action and  $t_G$  is the last action. If the ordering relation is total,  $P$  is a linear plan, whereas if it is a partial-order relation,  $P$  is a non-linear plan.
- $B$  is a set that allows describing the bindings and forbidden bindings on the variables appearing in  $P$ .

- $L$  is a set of casual links of the form  $s \xrightarrow{p} s'$ , where  $s, s' \in S$ ,  $p \in E_s$  and  $p \in C_{s'}$ . That is, relations allowing to establish a link between plan actions.

A plan  $P$  constitutes the solution generated to solve a planning problem when for each action  $s \neq t_i$ , for each  $p \in C_s$  there exists a causal link  $s \xrightarrow{p} s'$  and for each action  $s \neq t_G$  there exists at least a causal link  $s \xrightarrow{a} s''$ . In the case that the planner is interested in retaining the failures or unexpected situations during the plan, these failures or situations are represented as a set of formulae  $F$ .

### 3.1 The Planning Model of the CBPMP Planner

A CBP planner is specialized in generating plans in execution time [7]. The purpose of the CBP planner is to solve new problems by adapting solutions that have been used to solve similar problems in the past [22], and the CBP are a variation of the CBR systems, based on the plans generated from each case. The CBPMP planner requires dynamic planning systems that allow them to respond to changes in the environment and to provide efficient plans in execution time for optimising the working rotas. The CBP planner used by the CBPMP identifies a plan, for a given nurse, to provide daily nursing care in the hospital. It is very important to maintain a map with the location of each patient at the time of planning or replanning, which is why RFID technology is used to facilitate the location and identification of patients, nurses and doctors.

Let  $E = \{e_0, \dots, e_n\}$  the set of tasks that the nurse is assigned to.

$$a_j : \begin{matrix} E \\ e_i \end{matrix} \rightarrow \begin{matrix} E \\ a_j(e_i)=e_j \end{matrix} \quad (7)$$

A plan is the name given to a sequence of actions (1) that, from a current state  $e_0$ , defines the path of states through which the agent passes in order to offer the nurses the optimum path according to their characteristics. Below, in (8), the dynamic relationship between the behaviour of the agent and the changes in medium are modelled. The behaviour of planner A can be represented by its action

function  $a_A(t) \forall t$ , defined as a correspondence between one moment in time  $t$  and the action selected by the planner,

$$\text{Agent } A = \{a_A(t)\}_{t \in T \subseteq N} \quad (8)$$

From the definition of the action function  $a_A(t)$  it is possible to define a new relationship that collects the idea of an agent's action plan (9),

$$p_A : \begin{matrix} TXA \\ (t, a_A(t)) \end{matrix} \rightarrow \begin{matrix} A \\ p_A(t) \end{matrix} \quad (9)$$

in the following way,

$$p_A(t_n) = \sum_{i=1}^n a_{iA}(t_i - t_{i-1}) \quad (10)$$

Given the dynamic character desired for the planner, for a definition of the CBPMP plan, the continuous extension of the previous expression (10) is proposed, in other words (11)

$$p_A(t_n) = \int_{t_0}^{t_n} a_A(t) dt \quad (11)$$

The variation of the agent plan  $p_A(t)$  will be provoked essentially by: The changes that occur in the environment and that force the initial plan to be modified, and the knowledge from the success and failure of the plans that were used in the past, and which are favoured or punished via learning.

The planning is carried out through a neural network based on the Kohonen network [21]. The neurons are organized in a two-layer unidirectional architecture. The learning method is presented as follows: (The equations are presented in the order they should be executed).

- To present the input vector  $X^p = (x_1^p, \dots, x_i^p, \dots, x_N^p)^T$  in the input layer.
- The weightings initially take random weightings in (0,1).
- To calculate the intensity of the neurons of the output layer. The Euclidean distance:

$$y_k = \sqrt{\sum_{i=1}^N (x_i - w_{ki})^2} \quad (12)$$

- To determine the winning neuron that will be that of smaller Euclidean distance.
- To upgrade the weights of the neurons that connect the input layer with the output neuron:

$$w_{ki}(t+1) = w_{ki}(t) + \eta(t)g(k, h, t)(x_i(t) - w_{ki}(t)) \quad (13)$$

Where:

$$g(k, h, t) = e^{\frac{-|k-h|^2}{2R(t)^2}} \quad (14)$$

(Gaussian function), therefore the formula is the following:

$$w_{ki}(t+1) = w_{ki}(t) + \eta(t)e^{\frac{-|k-h|^2}{2R(t)^2}} (x_i(t) - w_{ki}(t)) \quad (15)$$

Where  $w_{ki}$  is the weight of the connection between the input neuron  $i$  and the output neuron  $k$ ;  $t$  is the iterations;  $\eta$  is the learning rate;  $h$  is the position of the winning neuron;  $k$  is the neuron of the output layer; and  $i$  is the neuron of the input layer. In  $k-h$  a distance is calculated between the neurons. The Euclidean distance has been used.

The CBP mechanism follows the 4 stages of a CBR system (Retrieval, Reuse, Review and Retain). During each reasoning stage the CBP performs the following actions:

Retrieval stage. During this step the CBP mechanism selects the plans from the plan-base, which are similar or contain similar information to the problem case that describes the aims of a given user and his/her restrictions and profile. For the recovery of similar cases to the current problem, the multivariate technical of conglomerates analysis was used, that is the hierarchical method. The reason why the method of conglomerates analysis was chosen is because the recovered cases will be more similar to the current problem than it would be with other methods, since it is an analysis type in



which the variables are considered from a vectorial point and the comparisons are made keeping in mind the components of that vector. If there are two cases that have all the same components except for one, this method no longer considers them the same. The general idea consists of distributing  $n$  data, of dimension  $p$ , in "conglomerates" or groups (clusters) formed by data that are "similar to each other". To define the conglomerates, it is necessary to choose distances and measures of previous similarity. The usual distances are the euclidean and the distance of Mahalanobis. The algorithm of grouping hierarchical begins with  $n$  conglomerates (one for each individual observation). In successive steps the conglomerates unite for likeness. The process of the algorithm can be graphically represented in a dendrogram. The number of conglomerates can decide the starting from the dendrogram, stopping the process of mass at the moment that the application of the algorithm takes to unite conglomerates that are very distant [25].

Reuse stage. Each nurse is assigned as set of tasks taking into account the set of daily tasks needing to be accomplished, nurse's skills and maximum service times estimated for each task and nurse. This assignation has to fulfill two conditions: i) the working day of each nurse is limited to 8 hours and ii) the number of hours assigned to each of the nurses has to be similar (a fair distribution of hours is expected).

$$\text{Max } \sum_{ij} g_{ij} e_{ij}$$

$$\sum_j t_{ij} \leq 8 \quad \forall i$$

$$\sum_j t_{ij} \cong \sum_j t_{kj} \quad \forall i \neq k$$

$$\sum_{ij} e_{ij} = \#\{\text{Pending tasks}\}$$

$$\text{Where } e_{ij} = \begin{cases} 1 & \text{if nurse } i \text{ is assigned task } j \\ 0 & \text{otherwise} \end{cases}$$

$g_{ij} \in [0,1]$ , 1 is the higher value for specialization and it indicates the level of specialization (or skill) of the nurse  $i$  to carry out the task  $j$ .

$t_{ij}$  is the maximum time employed by nurse  $i$  to accomplish the task  $j$  for those of the retrieved cases.

Once the nurse has been chosen all the tasks similar to the ones assigned to the nurse are retrieved. A task is retrieved only if its service time is less or equal to the time interval bound to the task type in the problem considered.

The neural network receives and inputs the coordinates corresponding to the rooms or wards that the nurse has to visit, according to the assignation previously explained, the starting time and finish time for each of the tasks and an estimation of the service times. The estimation of the service time for a task is obtained as the median of the service times of the retrieved cases for the task.

The result provided by the neural network is a route containing a schedule of the working day for the nurse. The nurse has to accomplish the tasks assigned, taking into account the starting and finish times for each of them.

Review stage. During the revision stage the user confirms the plan, indicating his/her satisfaction degree.

Retain stage: During the retain stage the plans are stored depending on their efficiency.

In Figure 2 it is possible to observe the steps carried out in each of the stages of the CBP system. The reuse stage implements the replanning mechanism. When an interruption occurs, the system initiates a new CBP cycle, taking into account the tasks previously accomplished. That is, in the new retrieval stage plans with a similar problem description to the current situation (after the interruption) will be recovered. The replanning mechanism guarantees that at least some of the plans closest to the initial plan will be recovered (the rest of the plans are not valid anymore because of the restrictions,

that the tasks have already accomplished, etc.) together with new plans. The proposed CBP mechanism has been tested in a real environment as shown in the next section.

When the network obtains as a result the optimum plan  $p^*$ . If the plan  $p^*$  is not interrupted, the planner will reach a desired state  $e_j \equiv e^*$ . In the learning phase, a weighting  $w_j(p)$  is stored. With the updating of weighting  $w_j(p^*)$ , the planning cycle of the CBP motor is completed. Let's suppose that the planner has initiated a plan  $p^*$  but at a moment  $t > t_0$ , the plan is interrupted due to a change in the environment. The solutions given by the neural network meet the conditions of the Bellman Principle of Optimality [3], in other words, each one of the plan's parts is partially optimum between the selected points. This guarantees that if  $g_0$  is optimum route for interrupted  $e_0$  in  $t_1$ , because  $e_0$  changes to  $e_1$ , and  $g_1$  optimum route to  $e_1$  that begun in the state where  $g_0$  has been interrupted, it follows that:  $g = g_0 + g_1$  is optimum route to  $e = e_0 (t_1 - t_0) + e_1 (t_2 - t_1)$ .

The dynamic process follows the CBP cycle recurrently: each time a plan finds itself interrupted; it generates from the state reached so far, the surroundings of the plans from the case base and adjusts them to the new problem.

#### **4. Case Study: Intelligent Environment for Hospitals**

The hospital Santísima Trinidad of Salamanca has been interested in improving the services offered to its patients and has collaborated in the development of the technology presented here, providing their know-how and experimenting with the prototype developed. This hospital is intended for people over 65 years old, and has the following services and facilities among others: TV room, geriatric bathroom, hairdressing salon, medical service, religious attention, occupational therapy, technical assistance, terrace, garden, laundry service, clothes adjustment, infirmary, reading room, living room, room of visits, cafeteria, social worker, chapel, elevator, customized diet, and multipurpose room.

#### 4.1 Hospital Intelligent Environment

Figure 3 shows a diagram of the first floor of the Santísima hospital of Salamanca containing the main facility rooms, while all the patients' rooms are located in the second floor. This hospital has capacity for 60 patients, an average of 6 nurses, one social worker and 5 more employees with other responsibilities. We selected 30 patients to test the system, so the hardware implemented at the hospital basically consisted of 42 ID door readers (Hitag HT RM401 and mobile WorkAbout Pro RFID), one on each door and elevator, 4 controllers, one at each exit, one in the first floor hall and another in the second floor hall, and 36 bracelets (Sokymat ID Band Unique Q5 with a chip Hitag S 256), one for each patient and the nurses. The ID door readers get the ID number from the bracelets and send the data to the controllers which send a notification to the Manager agent.

The Aml-P system is a distributed system of a relatively high dimension. Studying the requirements of the problem, four roles have been chosen: the Patient role manages the patient's personal data and behaviour (monitoring, location, daily tasks, and anomalies); the Doctor role treats patients; the Nurse role schedules the nurse's working day obtaining dynamic plans depending on the tasks needed for each assigned patient; the Manager role controls the patients' location and manages locks and alarms; Besides, the Manager role manages the medical record database and the doctor-patient and nurse-patient assignment. The dependences and relations between roles are also described. For the roles involved in the system a number protocols have been considered: request a treatment, inform about monitoring data, inform about care results, request a doctor assignment, request a nurse assignment, inform about assignment, request a patient's daily plan, inform about a patient's daily tasks, request a patient location, inform a nurse about a lock activation, report alarm activation, request doctor situation, doctor reports on his schedule, request a nurse situation, nurse reports situation, patient reports an anomaly, patient reports on personal data and previous medical records. For example, when the nurse wants to know the tasks required for the patient, the Nurse role executes a protocol RequestPatientPlanif through which is able to make a request to the Patient role. The Patient

role acts to give a suitable response to the Nurse role and executes the InformPlanif protocol to communicate the planned tasks to the Nurse role. The conclusions obtained after the analysis and design process let us conclude that AmI-P is composed of four different types of roles, as can be seen in Figure 4:

- Patient role manages the patient's personal data and behaviour (monitoring, location, daily tasks, and anomalies). Every hour validates the patient location, monitors the patient state and sends a copy of its memory base (patient state, goals and plans) to the manager role in order to maintain backups. The patient state is instantiated at execution time as a set of objects and these objects are controlled through goals that must be achieved or maintained. The objects that were seen to define a general patient state at the Santísima Trinidad hospital of Salamanca were: weight, temperature, blood pressure, feeding (diet characteristics and next time to eat), oral medication, parenteral medication, posture change, toileting, personal hygiene, and exercise. The objects and goals used for every patient depend on the plan (treatment) or plans that the doctors prescribe. The patient role monitors the patient state by means of the goals. To know if a goal has been achieved or has failed, it is necessary to maintain continuous communication with the rest of the AmI-P roles. At least once per day, depending on the corresponding treatment, the patient role must contact the nurse role. The patient role must have periodic communication with the doctor role. Finally the patient role must ensure that all the actions indicated in the treatment are taken out.
- Manager role controls the patients' location and manages locks and alarms as well as the management of the medical record database and the doctor-patient and nurse-patient assignment. It must provide security for the patients and medical staff and the patients, doctors and nurse assignment must be efficient.
- Doctor CBPMP role treats patients. The Doctor role needs to interact with the Patient role to order a treatment and receive periodic reports, with the Manager role to consult medical records and assigned patients, and with CBPMP role to ascertain the patient evolution.

- CBPMP role schedules also the nurse's working day obtaining dynamic plans depending on the tasks needed for each assigned patient. CBPMP manages nurses' profiles, tasks, available time and resources. The generated plans must guarantee that all the patients assigned to the nurse are given care. The nurse can't exceed 8 working hours. Every role generates personalized plans depending on the nurse's profile and working habits.

Manager and Patient roles run in a central computer, but CBPMP roles run on mobile devices, so a robust wireless network has been installed as an extension to the existing wired LAN. With respect to the question of failure recovery, a continuous monitoring of the system is carried out. Every role saves its memory (personal data) onto a data base. The most sensitive roles are patient roles, so these roles save their state every hour. When a role fails, another instance can be easily created from the latest backup. The database and server used must have redundancy and failure recovery, so a RAID (Redundant Array of Inexpensive Disks) server is used. In the case of a server failure, an alarm is generated and all the plans and information required for nurses and doctors to carry out their working day are automatically printed. A secure and authenticated access to the patient's data is provided. The use of different authorisations for users, logins and passwords, and the encryption of messages using a public key infrastructure and SSL (Secure Socket Layer) have already been implemented. Moreover, the RFID tag only contains the identification number

#### **4.2 Tasks Planning System**

The objectives of CBPMP planners are: to plan the nurses and doctors working time dynamically, to maintain the standard working reports about their activities, and to guarantee that the patients assigned to the nurses are provided with suitable care. Thus the CBPMP planner schedules the working days obtaining dynamic plans depending on the tasks needed for each assigned patient.

Given a set of memories  $B$  compatible with the problem  $E$ , it is possible to generate a plan base CBP that contains all the possible plans produced by the combinations of compatible memories. The

memories available for the CBPMP planner are tasks, resources and time. A task is a java object that contains the data of the patient who requested the service, the description of the service and the time limits to carry it out, as can be seen in Table 1. For each task one or more goals are established, in such a way that the whole task is eventually achieved. A goal is also a java object, that identifies what the CBPMP planner wants to achieve (complete a task) and under which conditions (restrictions). For this, a goal can contain parameters and define creation conditions (that allow CBPMP to define the conditions for achieving the goal), context conditions (the conditions that must be fulfilled) or drop conditions.

The CBP system constructs plans as a sequence of tasks that need to be carried out by a nurse. A description of the problem will be formed by the tasks that the nurse needs to execute, the resources available, and the times assigned for their shift. In the retrieval stage, the descriptions of similar problems are recovered. In order to do this, the CBPMP planner allows the application of various similar algorithms (cosine, clustering etc.). In this step, those problem descriptions found within a range of similarity close to the original problem description are recovered from the beliefs base. In our case, a tolerance of 20% has been permitted.

Once the most similar problem descriptions have been recovered, the reuse stage recovers the solutions associated with them. One solution contains all the plans (sequences of tasks) that are carried out in order to achieve the objectives of CBPMP for a problem description (assuming that replanning is possible) in the past, as well as the efficiency of the solution being supplied. At this time CBPMP takes control of the processing of the plan (scheduling). The reuse stage is centred around the objectives and resources needed by each task, as well as on the objectives that the nurse needs to perform and the resources available in order to carry out the global plan. The objectives or global plans that each nurse has are to attend to the patients and not to work for over eight hours. The time available is a problem restriction. Reuse stage takes care of incidents and interruptions that may occur during replanning.

In order to illustrate how the planner works, let's take a significant example. In the first place it is necessary to take into account that each nurse has a different profile according to their qualification and the tasks that they usually carry out. Let  $pr = \{pr_1, \dots, pr_{10}\} \subset I$  define the stored profiles of the nurses at the hospital. It is considered appropriate to manage the profiles of the nurses because there are some nurses who perform tasks with greater skill or who carry out tasks in less time. On the other hand, the CBMP planner maintains a close relationship with the Manager. The Manager has as one of its tasks the assignation of nurses to patients and doctors to patients. This assignation is carried out through the CBR reasoning motor of the Manager. When the new assignation of tasks needs to be carried out to the nurses or to the doctors, both past experiences, such as the profile of the nurse or doctor, and the needs of the current situation are retrieved. In this way tasks are allocated to a nurse. These tasks may correspond to the same patient or to a number of patients. Moreover, as mentioned above, the profile of each nurse is taken into account. For example, not all nurses are equally qualified for rehabilitation. If one nurse is more qualified in the area, she will be allocated the patients whose need for rehabilitation is greater, always taking into account that the nurse cannot work more than 8 hours, so that the number of patients assigned depends on the time needed to carry out the rehabilitation. The Manager takes into account how those patients who receive rehabilitation are improving, the arrival of new patients, holiday rotas etc. As such, the allocation of tasks needs to be set on a daily basis.

Secondly, it is necessary to store within the beliefs base the time that each task takes, described as  $t_j = \text{Max}_{j,k} \{t_{jk}^i\}$ , where j indicates the type of task, k, the nurse with the most suitable profile to carry it out (since it is only possible to assign on each task type to the nurses who are qualified to carry it out) and i, the patient that requires the task,  $t_j \subset I$ .

The first step when working with CBRs is obtaining a definition of the case structure. A case structure is composed of three elements: a problem description, the solution applied to solve the



problem and the result obtained for the solutions applied [4]. For the current problem the case is described as follows:

- **Problem description:** Describes the initial information available for generating a plan. As can be seen in Table 2 the problem description for planning a nurse's working day consists of a case identification, the list of tasks that the nurse has to accomplish, the nurse's profile, the nurse's preferences and temporal restrictions.

Where the structure of a Task can be observed in Table 7, and the structure corresponding to a nurse profile can be observed in Table 6.

- **Solution:** Describes the actions carried out in order to solve the problem description. As can be seen in Table 3 it contains the case identification, the timetables and the priorities for each task and the route or sequence of tasks.

Where the TimeTables field contains a list of Timetable elements. A Timetable element, as shown in Table 4, indicates the identification of a task, the priority of a task, the desired starting and finishing times for a task and the room identification if required.

The Route field contains a description of the activities carried out in a room of the hospital, as shown in Table 5.

- **Efficiency:** contains the case id and a number indicating the plan efficiency

For each of the rooms in the hospital, the system manages information related to the room location (in coordinates), access (doors) location (in coordinates), room type and resources available.

In our case study, at the beginning of each working day, every nurse receives a list of tasks that must be completed. The tasks must be assigned to the nurses. It is necessary to take into account that each nurse has a different profile according to their qualification and the tasks that they usually carry out. It is considered appropriate to manage the profiles of the nurses because there are some nurses who perform tasks with greater skill or who carry out tasks faster. The structure of a nurse profile can

be observed in Table 6. Table 6 shows the abilities of a nurse, her service time and timetable preferences and restrictions.

One of the Manager agent's duties is the assignation of nurses to patients. This assignation is carried out through a CBR reasoning motor integrates within the Manager Agent. When a new assignation of tasks needs to be carried out to the nurses, both past experiences, such as the nurse profile, and the needs of the current situation are retrieved. In this way tasks are allocated to a nurse. These tasks may correspond to the same patient or to a number of patients. The task structure is shown in Table 7. Moreover, as mentioned above, the profile of each nurse is taken into account. For example, not all nurses are equally qualified for rehabilitation. If one nurse is more qualified in the area, she will be allocated the patients whose need for rehabilitation are greater, always taking into account that the nurse cannot work more than 8 hours, so that the number of patients assigned depends on the time needed to carry out the rehabilitation. The Manager agent takes into account how those patients who receive rehabilitation are improving, the arrival of new patients, holiday rotas etc. As such, the allocation of tasks needs to be set on a daily basis.

Once the assignation of tasks to a nurse has been completed, the assignation is communicated to the corresponding Nurse agent. That is, the Nurse agent receives a new problem description needing to be resolved. From this moment on, the CBP planning process begins. The Nurse agent must take into account the time that the nurse has available and the time required for each task. Moreover, the resources available and the location of the patients involved are also taken into account.

The location of the patients is a factor which significantly influences the decision as to whether a plan should be interrupted. For example, in the case that a nurse should go to a given room to take dinner to the patient and the patient is actually in a different room, the nurse's plan will need to be interrupted. As mentioned above, the location of the patients within the hospital is defined through a reference system in  $\mathfrak{R}^2$ . In the location system, it is fundamental that RFID devices are used. These devices make it possible to rapidly assess the possibility or need to replan.

A plan can be interrupted for different reasons. Those which have been taken into account within the hospital are: that a resource fails, that a patient suffers some sort of crisis and requires unforeseen attention, that the patient has an unexpected visit or that visits to the patient have gone on over the permitted time allowed and an emergency situation. If the planner finds itself in a situation where the plan is interrupted, it rejects the initial plan and seeks an alternative one. The first thing that needs to change is the task order, attempting to maintain the assignation originally allocated by the Manager. The new plan must meet the initial objectives. In the event that this is impossible, the nurses will need to be reassigned. This reassignment will attempt to limit changes to a minimum. For reassignment it is necessary to take into account the tasks that were assigned to the nurses, the development of the plans (which tasks have been carried out and which still need to be done) and the profiles of the nurses (prioritising preparation for the task that cannot be covered). The nurse who is assigned the task should replan in order to include the new task. In the event that the replanning is positive (the tasks that still need to be done and the new task can be carried out) the process is complete. If the replanning is negative, the next nurse down in the ranking will be used.

Lastly, depending on the efficiency of the plan, it will be stored together with its level of efficiency within the beliefs base. In the paragraphs below, we give a specific example in detail.

Let  $E^i = \{e_0^i, \dots, e_h^i\}$  the task carried out on patient  $i$ , put in order of priority. We have the following problem  $E = \bigcup_i E^i = \{e_0, \dots, e_n\}$ , that is updated on a daily basis, where  $E$  denotes the complete set of tasks being carried out and therefore has no superscript.

Selecting a nurse  $k \in \{1, \dots, 10\}$  at random (in particular,  $k=3$ ), the assignation of tasks according to the profile is shown in Table 8.

Calculation of the tasks assigned verifies that the total time allocated does not go over 8 hours. As may be noted, when the tasks are being assigned, the location of the patients is not taken into account

(the patients are given the best treatment possible). But the location of the patients is taken into account when the plan is generated (in order to minimise the total time taken to carry out the tasks).

Once the assignation of tasks is complete, each CBPMP planner carries out a plan for its nurse. They retrieve similar assignations from the beliefs base, and the corresponding plans that were used. A plan is made and supplied to the nurse. The nurse then carries out the plan in sections, in other words, task by task (The current task is shown in the PDA and the nurse has to introduce the result obtained after the task has been accomplished). Each task has a series of objectives which must be reached for the part of the plan to have been completed successfully. In order to carry out each task the nurse must have a number of resources available.

The data obtained after the execution of the example can be observed in Table 9. In the case that any interruption occurs, the process previously explained is repeated again, that is, replanning is required. In the event that it had been impossible to reorganise the tasks that remained, communication would be made with the Manager. The Manager would need to reassign the tasks according to the level of expertise of the nurses.

## **5. Results and Conclusions**

From the point of view of Ambient Intelligence [11], [14], the present work evaluates the capabilities of ubiquitous computing, ubiquitous communication and intelligent user interaction that the proposed system presents. The system facilitates carrying out tasks in a ubiquitous way, such as plans generation for nurse's working day, or patient and task assignation. These tasks are executed in a distributed way inside the intelligent environment and are independent of the location of the processing devices due to the use of mobile devices and wireless networks. In a similar manner the system provides wireless communication by means of Wi-Fi and RFID technologies, allowing location-independent access. Finally, the system incorporates intuitive interfaces which facilitate usability and automatic adaptation to the user profile. We had certain problems implementing the system, partly because the nurses and workers were not familiar with the use of PDA devices, so some

courses were given to introduce them to these technologies and teach them how to use the system interface. After that and with some difficulties with the installation of the wireless access points (with the propagation of the signal) and the collocation of the RFID door readers, the system was running smoothly, with only minor problems.

In regard to efficiency and improvement in the quality of the services provided by the hospital, the implantation of the intelligent environment has allowed an increment of the productivity as well as providing novel additional services, such as security or automatic patient monitoring. Figure 5 shows the average number of nurses working simultaneously (each of the 24 hours of the day) at the hospital before and after the implantation of the system prototype. As can be seen in Figure 5, the blue line represents the average number of nurses required in the hospital each hour of a day without the AmI-P. The violet slides represents the same measure but after the implementation of the intelligent environment. As can be seen, the intelligent environment helps the nurses to gain time, which can be dedicated to the care of special patients, to learn or to prepare new activities. The time spent on supervision and control tasks has been reduced substantially, as well as the time spent attending false alarms, while the time for direct patient care has been increased. The system also facilitates the more flexible assignation of the working shifts at the hospital; since the workers have reduced the time spent on routine tasks and can assign this time to extra activities, such as exercising the patients, learning, carrying out leisure activities or just talking with the patients or with their families. Their work is automatically monitored, as well as the patient's activities. The stored information may be analysed with knowledge discovery techniques and may help to improve the quality of life for the patients and the efficiency of the centre. The security of the centre has also been improved in three ways: the system monitors the patients and guarantees that each one of them is in the right place; secondly, only authorised personnel can gain access to the hospital protected areas, and thirdly, the information is stored in a more secure way using redundance and generating continuous backups. The access to information has been protected in order to guarantee confidentiality.

The tasks executed by nurses were divided in two categories, direct action tasks and indirect action tasks. Direct action tasks are those which require the nurse acting directly on the patient during the whole task (medication, posture change, toileting, feeding, etc). In the indirect action tasks the nurses do not need to act directly on the patients all the time (reports, monitoring, visits). CBPMP planners can take care of some of these indirect actions, so nurses can dedicate more time to personal patients care. During the first testing period the problem was analysed and data was collected. The average time spent by nurses carrying out their duties with a given patient was obtained, having into account the patient type, its dependency level and the nurse professional level. For the direct action tasks, the following times were obtained for each patient: 35 minutes cleaning, 18 minutes feeding, 8 minutes oral medication, 30 minutes parenteral medication, 25 minutes posture change, 8 minutes toileting, 60 minutes exercise and 10 minutes' others for patients with a dependence degree of 1; and 45 minutes cleaning, 28 minutes feeding, 11 minutes oral medication, 42 minutes parenteral medication, 50 minutes posture change, 30 minutes toileting, 90 minutes exercise and 10 minutes others for patients with a dependence degree of 2. We are especially interested on time spent on indirect tasks; daily times obtained before and after the implementation. Table 10 shows how the implementation of the AmI-P reduces the time spent on indirect task. For example, the average number of minutes spent by a nurse on monitoring patients has been reduced from 167 daily minutes to 95 daily minutes without reducing the care level and the patients safety. The novel neural network based mechanism improves the results obtained with a previous geodesic planner [8].

The third point taken into account to evaluate the system presented within this paper focuses on the evaluation of the novel neural network-based planning mechanism. This dynamic planning mechanism allows replanning in execution time, which in turn improves patient care. To evaluate the efficiency as well as the quality of the proposed planning mechanism, our planner has been compared to a series of different planners considered as significant in the literature for this kind of problems. This comparison can be considered as a good reference to evaluate the behaviour and benefits of the proposed

replanning mechanism. The brute force technique allows obtaining a solution by exploring all the possible combinations of solutions. That is why the calculus time exponentially increases as the number of tasks increases. Another well known possibility is the use of heuristic planning techniques, as the A\* algorithm, which allows efficient searches in the solution space, trying to reduce the number of possible combinations to explore. However, in spite of this search, the number of combinations grows exponentially [17]. Classical planning is based on production system, but its main disadvantage is the search of a equilibrium between the cost of rules selection and the cost of rules application. Depending on the election, the algorithm efficiency is very variable [32]. Finally, the evolutionary computing techniques, specifically genetic algorithms, allow a great amount of efficient planning algorithms. The main advantage of the genetic algorithms is the possibility of allowing parallel solution searches, which are refined in the successive generations. Moreover genetic algorithms allow incorporating different heuristic techniques within the different existing operators along the evolution cycle. However, the main disadvantage lies in the efficiency, it is considerably reduced as the number of parallel solutions grows as well as in the difficulty to find a good encoding for the chromosome [18]. Once the planners used to evaluate our dynamic planner have been presented, the process followed to get an evaluation consisted on a first comparison to a brute force planner and to a A\* [17]. The next step consisted on comparing our planner to a classical planner by means of the creation of production systems which determine the order for the tasks to be applied. Finally, we compared our planner to an evolutionary computing planner based on genetic algorithms.

To obtain the same test conditions for all the planners, the method we followed consisted of the assignment of the same tasks to nurses with the same profile and similar conditions in the environment. In order to observe the efficiency of the planners looking at the number of tasks assigned to the nurse, the number of tasks was changed. The limits for the number of tasks assigned to a nurse were established between 10 and 30. The results obtained can be observed in Table 11. As can be seen in Table 11 the times obtained for the brute force planner are estimated from 15 points up since the

calculus implies excessive and impracticable computation times. Something similar happens for the A\* algorithm-based planner. The estimations have been done taking into account the number of combinations required as well as the time employed as a function of the length of the route. The number of total combinations for this problem is obtained as  $N!$  and the time employed is obtained as a comparison to the time employed for the estimation of the aptitude for a plan composed of 10 tasks. In the case of the A\* planner the results correspond to the estimation for an intermediate situation, in which it is only necessary to spread out the middle of the branches of each of the nodes. Basically to implement this algorithm we need to define an evaluation function in the way  $f^*(n)=g^*(n)+h^*(n)$ , where  $g^*(n)$  is the minimum cost to go from the origin node to the node  $n$  and  $h^*(n)$  is the minimum cost to reach the final node from the node  $n$ . As these functions are unknown, we need to estimate them by means of  $f(n)=g(n)+h(n)$  (A algorithm), imposing the restriction  $h(n)\leq h^*(n)$ .  $H(n)$  has been defined as  $h(n)=0$  and  $g(n)$  as the value for the accumulated route following the branches of the tree. Initially we begin from a state without any task assigned, so in the next level we are going to add so many branches as tasks have to be added. After this, the distances for the routes are calculated, as well as the fulfilment of the time restrictions and the valid branch with a minor distance  $f(n)$  is chosen. The classical planner was implemented trying to imitate the A\* algorithm behaviour. The genetic algorithms-based planner was designed using the procedures used to resolve the TSP problem. The TSP problem basically consists of a traveller who has to visit a number of cities with the restriction of passing only one time each city. The cross operators are modified in order to take into account the time restrictions only when the chromosomes are combined; the mutation operators are modified to allow the combined use of various of them; and the aptitude function is defined as positive for valid solutions and as negative for invalid solutions. In this way it can be guaranteed that all valid solutions obtain a higher aptitude value than invalid solutions.

As can be seen in the results shown in Table 11, the CBPMP planner has a better behaviour with respect to the increment of the number of tasks than the rest of the planners. The distances calculated,



although not optimum, are very close to the result, so they can be considered as acceptable. The rest of the planners, except the genetic algorithm-based one, are impracticable for 20 tasks up. That's the reason why it is impossible to replan in execution time for a number of tasks higher to 15 elements. However, the genetic algorithm-based planner needs an important time to provide a solution, and this is a serious problem in real environments where taking decisions need to be efficient related to the response time.

The CBP planning mechanism which allows replanning in execution time, integrated within an intelligent agent, has been tested over the last few months. To test the system 30 patient agents, 10 nurse agents, 2 doctor agents and 1 manager agent were instantiated. During the testing period the systems usefulness has been evaluated from different points of view. The system, which in turn improves patient care, facilitates the more flexible assignation of the working shifts at the hospital; since the workers have reduced the time spent on routine tasks and can assign this time to extra activities, such as exercising the patients, learning, carrying out leisure activities or just talking with the patients or with their families. Their work is automatically monitored, as well as the patients' activities. The stored information may be analysed with knowledge discovery techniques and may help to improve the quality of life for the patients and the efficiency of the centre. The security of the centre has also been improved in three ways: the system monitors the patients and guarantees that each one of them is in the right place; secondly, only authorised personnel can gain access to the hospital protected areas, and thirdly, the information is stored in a more secure way using redundancy and generating continuous backups. The access to information has been protected in order to guarantee confidentiality.

The agent technology provides a framework for distributed problem solving and data communication. The characteristics of agents, such as autonomy, reactivity, pro-activity, social abilities, reasoning, learning and mobility [33] are adequate to fulfil the AmI necessities. Moreover, the incorporation of case-based reasoning mechanisms [8] facilitates ubiquitous computation

capabilities. An agent can act as an interface between the user and the environment. In this way, it is possible to get that the services provided by the multiagent system and the technology in the environment can be adapted to the user necessities. Finally, agents can be executed on mobile devices, which facilitate ubiquitous capabilities. These characteristics make the proposed architecture appropriate for developing AmI [14] scenarios. The manager agent assigns tasks and monitors the patients and nurses locations. Nurse and Doctor agents run on handheld devices and provide easy-to-use interfaces.

The planning mechanism has been evaluated taking into account the characteristics of a CBR system [1]. In this sense, the parameters that influence the efficiency of each of the stages of the proposed CBP cycle have been studied in detail. In Figure 6 it is possible to see the interaction between the variables corresponding to the average satisfaction degree for a nurse related to the plan success and the number retrieved from the base case in order to provide the plan. As explained in Section 2, one of the key concepts in the efficiency of a CBR system consists of the retrieval algorithm implemented, and specifically, in the similitude metric applied. The success or efficiency of a plan can be calculated through the results obtained for each of the tasks of the plan and the evaluation provided by the nurse that has completed the plan. On the other hand, the number of retrieved cases allows evaluating the strategy proposed in the retrieval stage. The results presented in Figure 6 show how for a larger number of retrieved cases from the cases memory, the satisfaction degree indicated by nurses increases. That is, the nurses can better achieve their initial objectives when the number of similar past experiences available is larger. The relationship between the number of retrieved cases and the nurse satisfaction degree is stabilized for a number of cases around 110 and 200.

The case-based planning mechanism proposed in the frame of this research comes characterized by its capability for dynamic replanning in execution time. In order to obtain a better understanding of the success of the CBP planning system, it is of interest to take into account the average number or replanning actions executed for a plan. Figure 7 shows the interaction between the number of retrieved

cases, the average satisfaction degree (percentage) and the average number of replans per plan in the proposed CBP system. In Figure 7 it can be observed how the number of retrieved cases increases, the satisfaction degree grows and the average number of replannings required per plan decreases.

In Figure 8 it has been represented in the interaction that exists between the average number of cases retrieved in order to solve a plan and the average number of replannings required. It is possible to detect an inverse relationship between both variables, that is, when the average number of retrieved cases increases the average number of replannings required decrease and vice versa. Moreover, the graph presents an inflection point of 50 retrieved cases, corresponding to an average of 7 replannings. This fact let's us conclude that from 50 retrieved cases on, the number of replannings decreases smoothly. It is necessary to notice that a CBR system needs initial knowledge and a certain learning period to provide good results.

Another good indicator to evaluate the CBP is the success of the plans suggested. The plans recommended to the nurses can be successfully completed (replanning can be necessary or not) or failed. In the same way as the average number of replannings and the nurse satisfaction degree was influenced by the number of cases retrieved. The number of plans successfully completed depends on the retrieval strategy implemented. Figure 9 presents a 3x3 matrix representing the interaction between the average number of retrieved cases, the average number of completed plans and the average number of failed plans. Figure 9 let's us conclude that as the number of cases retrieved increases, the number of plans completed successfully increases and the number of plans failed decreases. In Figure 10 we can see that exits an inflection point in the row indicating the average number of retrieved cases and the column indicating the average number of plans completed. Such a point indicates, that from a certain number of retrieved cases on, the number of plans successfully completed stabilizes. Even, if the number of retrieved cases increases dramatically, the success for the proposed plans decreases.

Finally, the influence of the similitude degree obtained for the retrieved cases has been evaluated. Figure 11 presents the interaction between the percentage of plans successfully completed and the

similitude degree obtained during the retrieval stage. In Figure 11 it can be seen that, generally, as more similar the retrieved cases are to the current problem, the possibility of plan success increases. CBP is based on the use of past experiences to resolve new problems. Obviously, if we can obtain past plans containing similar tasks and these plans were executed by nurses with similar skills, the probability of success increases. However, the CBP is highly affected by environmental changes, and the success for a plan depends not only on previous experiences but also on the incidents that can happen during the execution of the plan, which cause events of interruption and the corresponding replanning actions.

In the future, health care for hospital patients, the elderly and people with other disabilities will require the use of new technologies that allow medical personnel to carry out their tasks more efficiently. Weick [31] describes the fundamental problems of knowledge transfer and sense making in digital/computer based environments. We have shown the potential of deliberative CBPMP agents in a distributed multi-agent system focused on health care, providing a way to respond to some challenges of health care, related for example to the identification, control and health care planning. In addition, the use of RFID technology on people provides a high level of interaction among users and patients through the system and is fundamental in the construction of the intelligent environment. Furthermore, the use of mobile devices, when used well, can facilitate social interactions and knowledge transfer.

### **Acknowledgements**

This work has been supported by the Spanish Ministry of Science and Technology project TIN2006-14630-C03-03 and the Spanish Ministry of Labor and Social Security project “Plataforma inteligente para la gestión integral de residencia geriátricas”.

### **References**

- [1] Aamodt, A., & Plaza, E. (1994). Case-Based Reasoning: foundational Issues, Methodological Variations, and System Approaches. *AI Communications*, 7, 39-59.

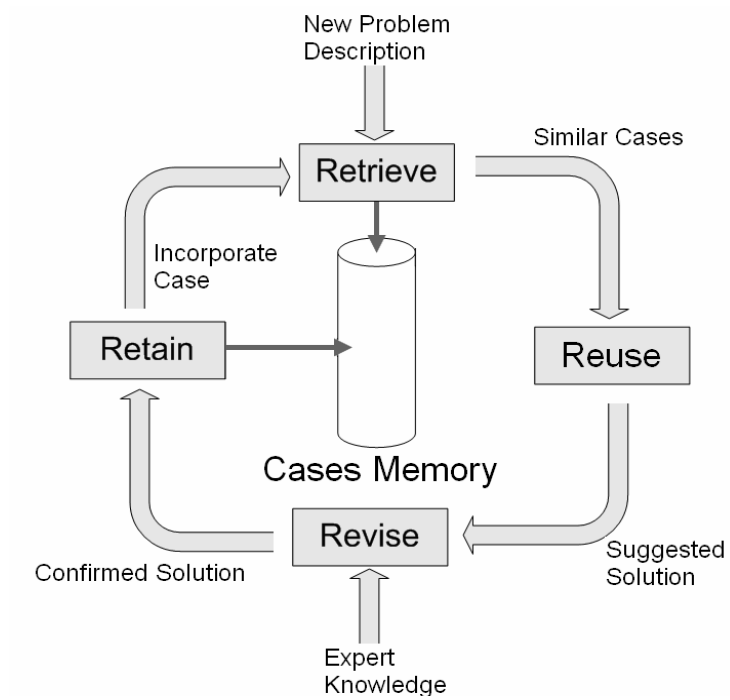
- [2] Aarts, E., & Marzano, S. (Eds.) (2003). *The New Everyday: Views on Ambient Intelligence*. Rotterdam: Uitgeverij 010 Publishers.
- [3] Bellman, R.E. (1957). *Dynamic Programming*. Princeton, New Jersey: Princeton University Press.
- [4] Bergmann, R., Kolodner, J., & Plaza, E. (2005). Representation in case-based reasoning. *The Knowledge Engineering Review*, 00 (0), 1–4.
- [5] Bratman, M.E., Israel, D., & Pollack, M.E. (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4, 349-355.
- [6] Camarinha-Matos, L., & Afsarmanesh, H. (2002). Design of a Virtual Community Infrastructure for Elderly Care. *Proceedings of the Third Working Conference on Infrastructures for Virtual Enterprises: Collaborative Business Ecosystems and Virtual Enterprises* (pp. 635). Sesimbra, Portugal.
- [7] Corchado, J.M., & Laza, R. (2003). Constructing Deliberative Agents with Case-based Reasoning Technology. *International Journal of Intelligent Systems*, 18, 1227-1241.
- [8] Corchado, J.M., Bajo, J., de Paz, Y., & Tapia, D.I. (2008). Intelligent Environment for Monitoring Alzheimer Patients, *Agent Technology for Health Care*. *Decision Support Systems*, 44 (2), 382-396.
- [9] Cox, M.T., Muñoz-Avila, H., & Bergmann, R. (2005). Case-based planning. *The Knowledge Engineering Review* 00 (0), 1–4.
- [10] Decker, K., & Li, J. (1998). Coordinated Hospital Patient Scheduling. *Proceedings of the Third International Conference on Multi Agent Systems* (pp. 104-111). Paris, France.
- [11] Ducatel, K., Bogdanowicz, M., Scapolo, F., Leijten, J., & Burgelman, J.C. (2001). That's what friends are for Ambient Intelligence (AmI) and the IS in 2010. In *Innovations for an e-Society*. Congress Pre-prints, *Innovations for an e-Society*. Challenges for Technology Assessment, Berlin, Germany.

- [12] Emiliani, P.L., & Stephanidis, C. (2005). Universal access to ambient intelligence environments: opportunities and challenges for people with disabilities. *IBM Systems Journal*.
- [13] Foster, D., McGregor, C., & El-Masri, S. (2005). A Survey of Agent-Based Intelligent Decision Support Systems to Support Clinical Management and Research. *Proceedings of the First International Workshop on Multi-Agent Systems for Medicine, Computational Biology, and Bioinformatics* (pp. 16-35). Utrecht, Netherlands.
- [14] Friedewald, M. and Da Costa, O. (2003). *Science and Technology Roadmapping: Ambient Intelligence in Everyday Life (AmI@Life)*. Seville: Institute for Prospective Technology Studies IPTS.
- [15] Glez-Bedia, M., & Corchado, J.M., (2002). A planning Strategy based on Variational Calculus for Deliberative Agents. *Computing and Information Systems Journal*, 10 (1), 2-14.
- [16] Hammond, K.J. (1989). *Case-Based Planning: Viewing Planning as a Memory Task*. Academic Press Inc.
- [17] Hart, P.E., Nilsson, N.J., & Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths Graphs. *IEEE Trans. on Systems Science and Cybernetics*, 4 (2), 100-107.
- [18] Haupt, R.L., & Haupt, S.E. (2004). *Practical Genetic Algorithms*. Wiley.
- [19] Jin, H.D., Leung, K.S., Wong M.L., & Xu, Z.B. (2003). An Efficient Self-Organizing Map Designed by Genetic Algorithms for the Traveling Salesman Problem. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 33 (6), 877-888.
- [20] Kohn, L.T., Corrigan, J.M., & Donaldson, J. (1999). *To Err is human: Building a Safer Health System*. Committee on Quality of Health Care in America Institute of Medicine, Washington, D.C.: National Academy Press.
- [21] Kohonen, T. (2001). *Self-Organising Maps*. Berlin: Springer-Verlag.
- [22] Kolodner, J. (1993). *Case-Based Reasoning*. San Mateo: Morgan Kauffman.

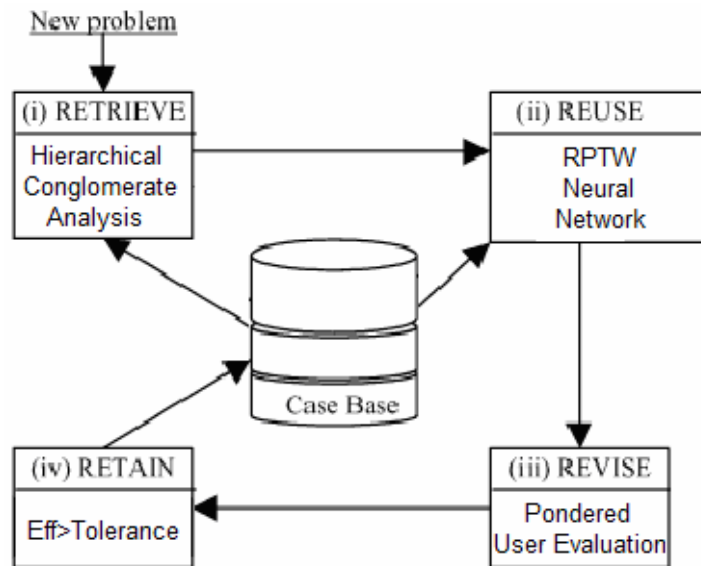
- [23] Lanzola, G., Gatti, L., Falasconi, S., & Stefanelli, M. (1999). A Framework for Building Cooperative Software Agents in Medical Applications. *Artificial intelligence in medicine*, 16 (3), 223-49.
- [24] Leung, K.S., Jin H.D., & Xu, Z.B. (2004). An Expanding Self-Organizing Neural Network for the Traveling Salesman Problem. *Neurocomputing* 62, 267-292.
- [25] Martín, Q., Santos M.T., & de Paz, Y. (2005). *Investigacion Operativa: Problemas y Ejercicios* : Pearson Educacion.
- [26] Martín, Q., de Paz, J.F., de Paz, Y., & Pérez, E. (2008). RETRACTED: Solving TSP with a modified kohonen network. *European Journal of Operational Research*. In Press.
- [27] Nealon, J., & Moreno, A. (2003). Applications of Software Agent Technology in the Health Care domain. In *Whitestein series in Software Agent Technologies*. Berlin: Birkhauser.
- [28] Spalazzi, L. (2001). A Survey on Case-Based Planning. *Artificial Intelligence Review*, 16 (1), 3-36
- [29] Susperregi, L., Maurtua, I., Tubío, C., Pérez, M.A., Segovia, I. & Sierra, B. (2004). Una arquitectura multiagente para un Laboratorio de Inteligencia Ambiental en Fabricación. *Proceedings of the first Taller de Desarrollo de Sistemas Multiagente* (pp. 23-35). Málaga, Spain.
- [30] U.S. Department of Commerce (2005). *Radio Frequency Identification: Opportunities and Challenges in Implementation*. Technical Report..
- [31] Weick, K.E. (1985). *Cosmos vs. Chaos: Sense and Nonsense in Electronic Contexts*. R.L. Ruggles, Ed., *Knowledge Management Tools*, 3.
- [32] Wilkins, D.E. (1988). *Practical Planning: Extending the Classical AI Planning Paradigm*. San Francisco, CA, USA.: Morgan Kaufmann Publishers Inc.

- [33] Wooldridge, M., & Jennings, N.R. (1995). Agent Theories, Architectures, and Languages: a Survey. In: Wooldridge and Jennings (Eds.), *Intelligent Agents* (pp. 1-22). Berlin: Springer 1-22.

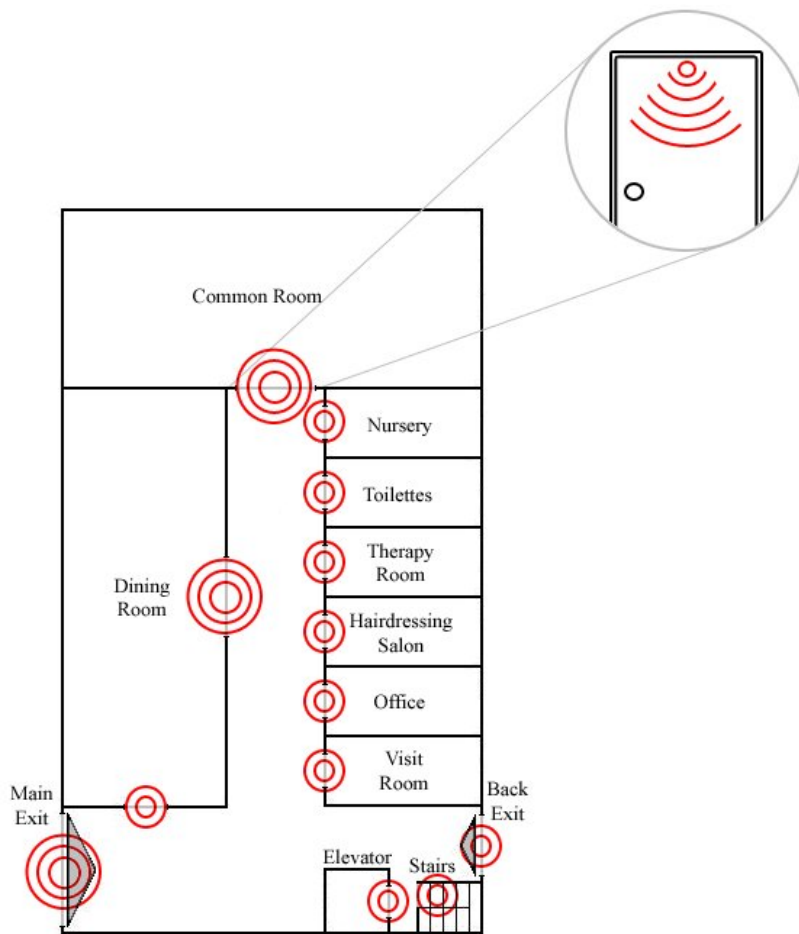




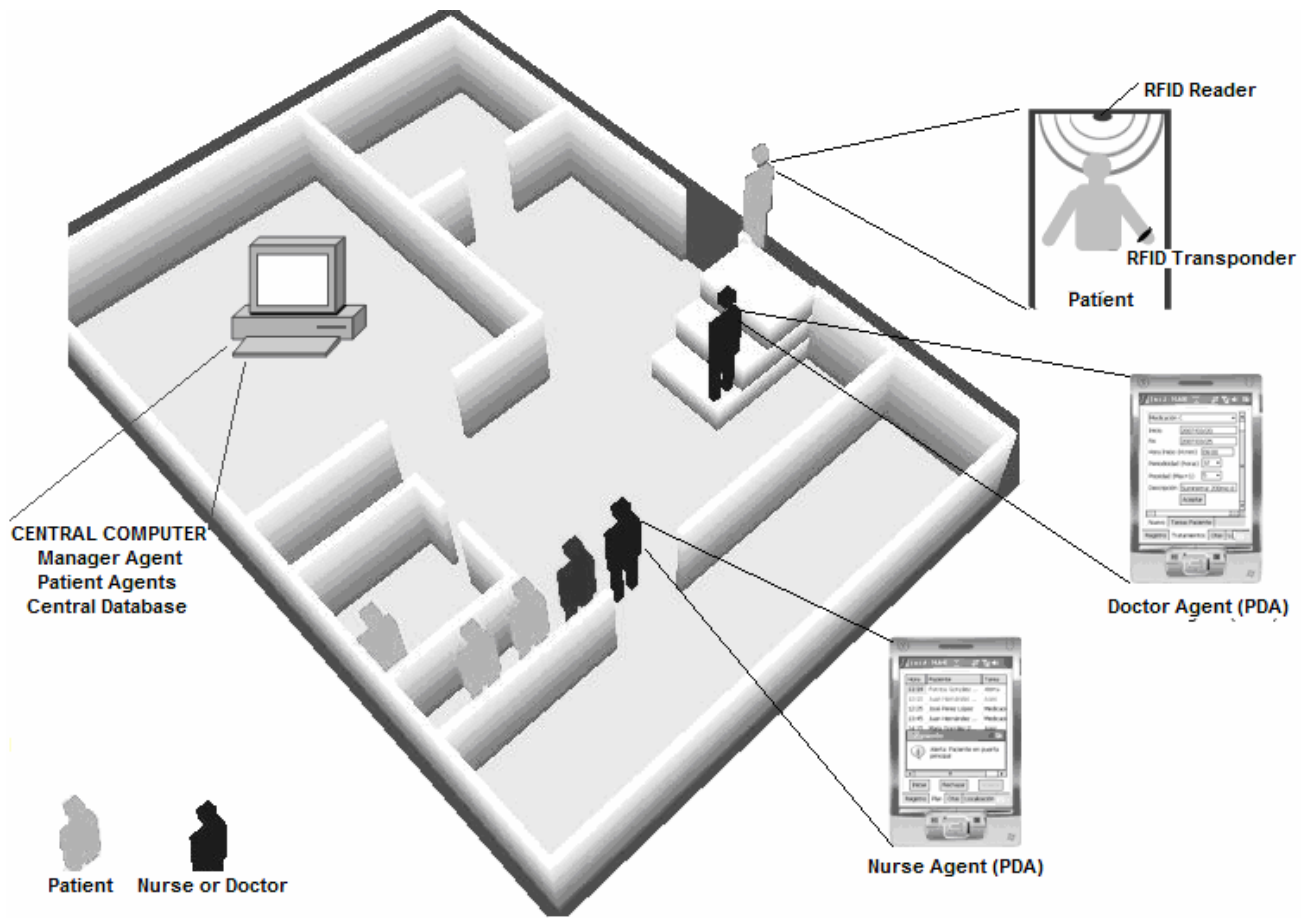
**Figure 1.** Diagram including a CBR-BDI agent reasoning cycle.



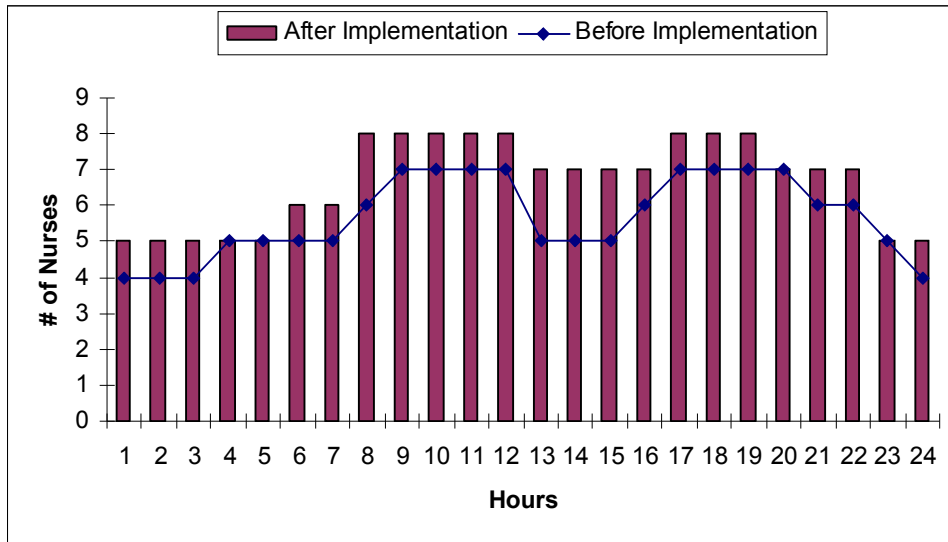
**Figure 2.** Case-based planning mechanism for dynamic environments.



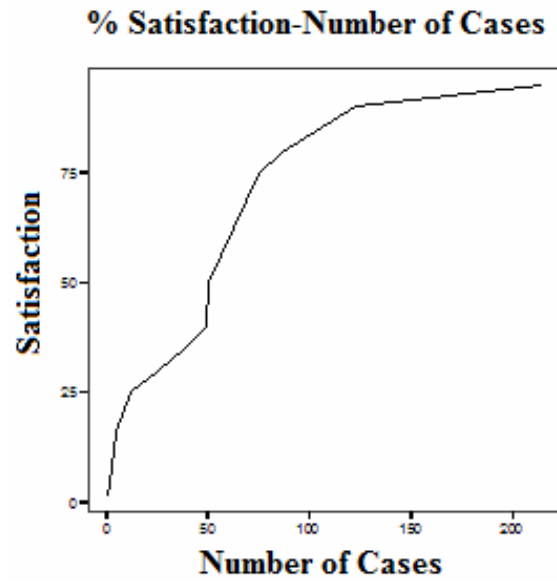
**Figure 3.** Sensor positioning in the first floor of the Santísima Trinidad hospital of Salamanca



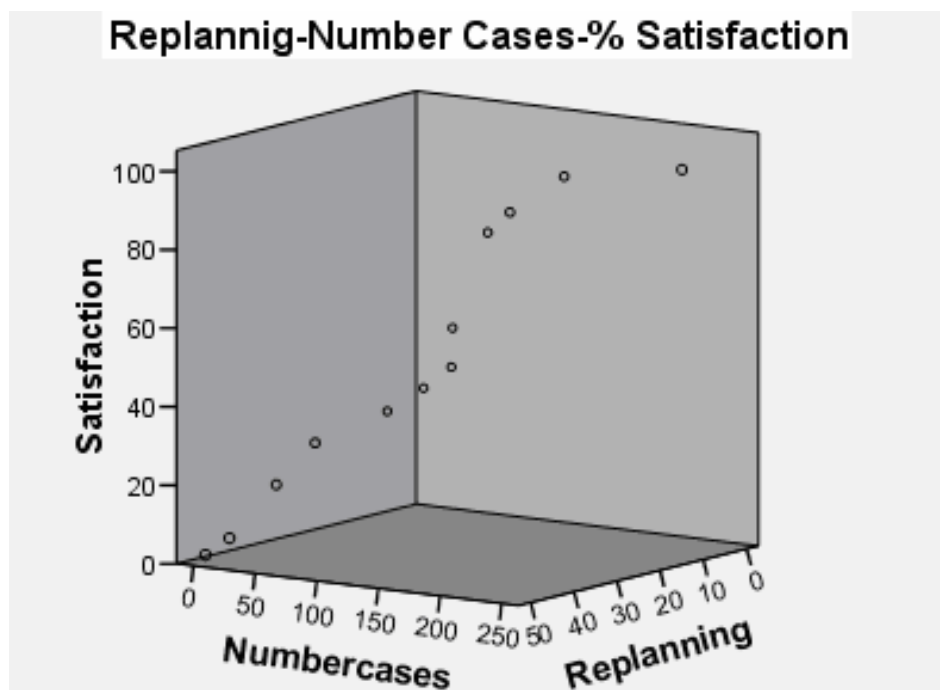
**Figure 4.** Wireless technology organization schema.



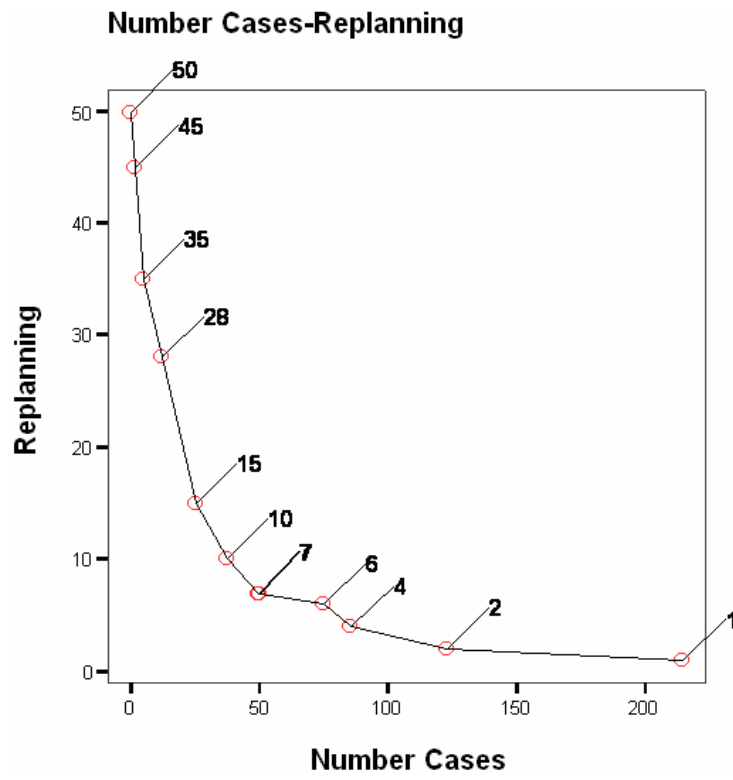
**Figure 5.** Number of nurses working simultaneously



**Figure 6.** % Average satisfaction degree related to the average number of retrieved cases

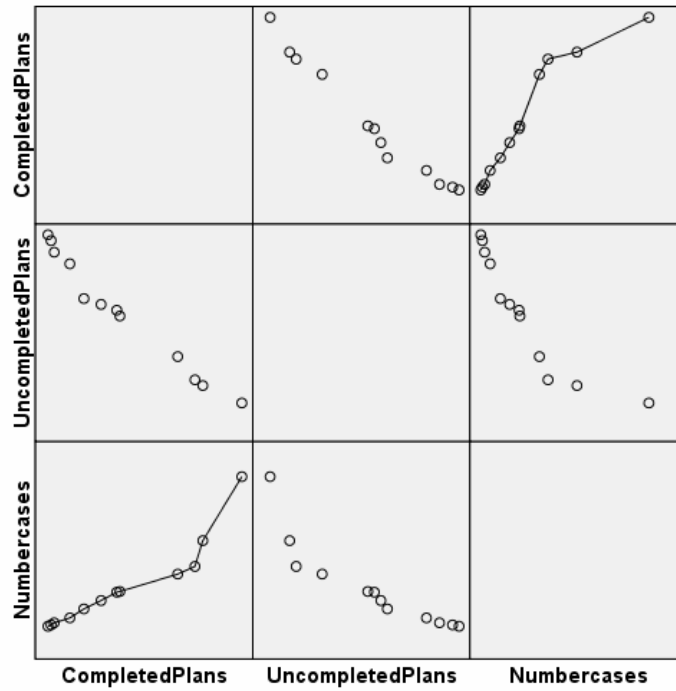


**Figure 7.** Interaction between average number of cases retrieved, average number of replannings required per plan and the nurse satisfaction degree

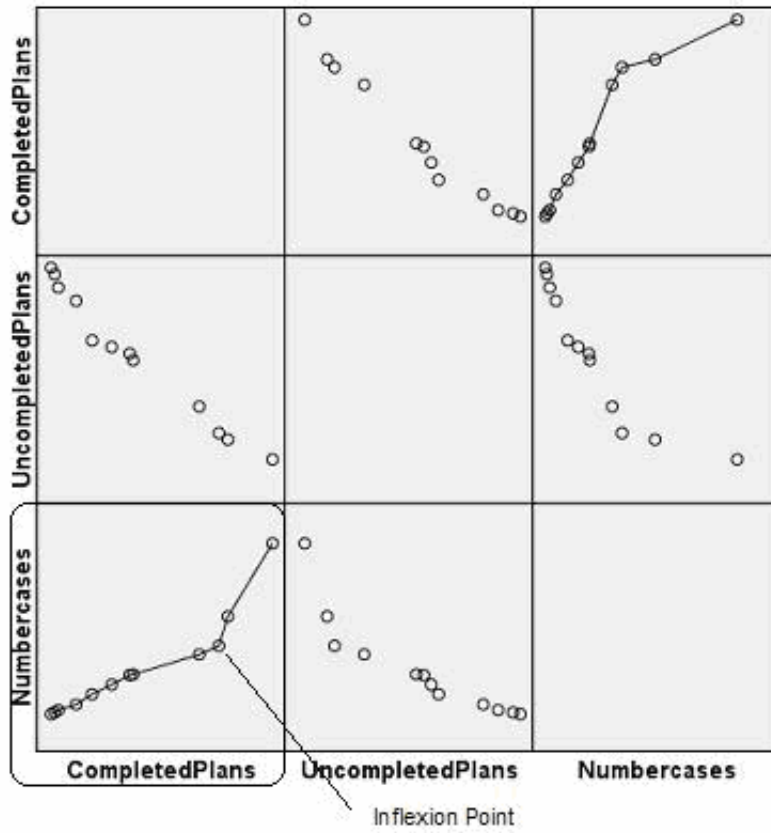


**Fig. 8.** Number of replannings related to cases retrieved.

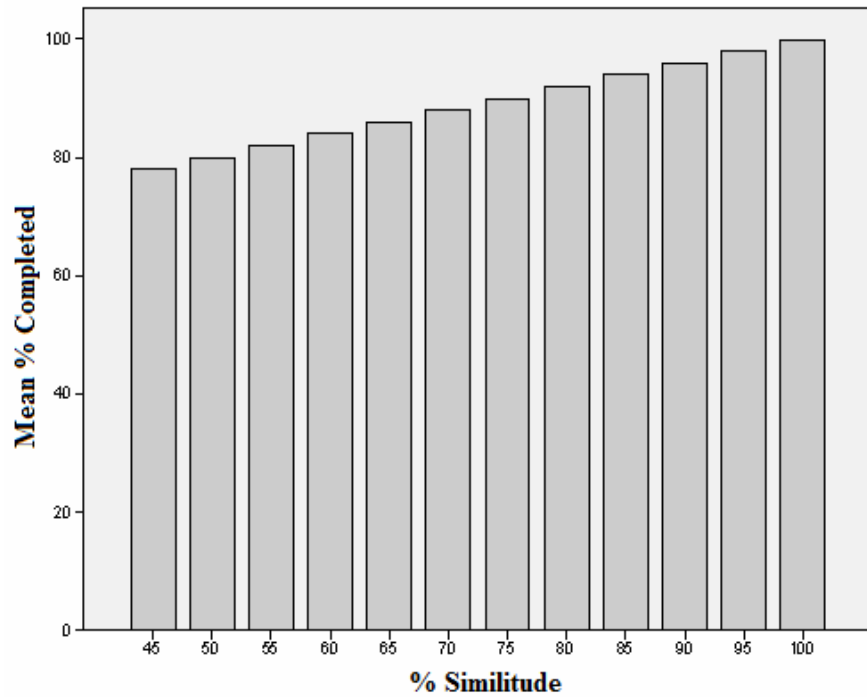




**Figure 9.** Plans completed and plans failed related to the average number of cases retrieved



**Figure 10.** Inflexion point in the interaction between the average number of completed plans and the average number of retrieved cases



**Figure 11.** % Plans completed related to the % of similitude for retrieved cases

**Table 1.** Task example

Task	Data
TaskId	36
TaskType	32
TaskDescript	Feeding (lunch)
TaskPriority	3
TaskObjective	0
TaskIncidents	0
PatientId	7
PatientDependence	2
MinTime	12:30
MaxTime	15:00
TaskResources	Food 1

**Table 2.** Problem description for a planning a nurse working day

Problem Description Field	Field Type
CaseId	Integer
Tasks	ArrayList of Tasks
Nurse Profile	NurseProfile
Nurse Preferences	NursePreferences
Restrictions	ArrayList of Restriction

**Table 3.** Solution (Plan) proposed to the nurse

Solution Field	Field Type
Case d	Integer
TimeTables	ArrayList of TimeTable
Route	Route

**Table 4.** Timetable for a task

TimeTable	Field Type
Priority	Float
StartTime	Time
FinishTime	Time
Task	Float
Room	Float

**Table 5.** Route describing the activities executed by a nurse in a given room

Route Fields	Field Type
room	Room
ArrivalTime	Time
ServiceTime	Time
patient	Patient
Tasks	ArrayList of Task
NextRoom	Route



**Table 6.** NurseProfile

NurseProfile Fields	Field Type
Skills	ArrayList of Skill
ServiceTime	Time
Timetable preferences	Date
Timetable restrictions	Date

**Table 7.** Task structure

Task Field	Field Type
taskData	TaskData
ServiceTime	Time
Date	Date
Room	Room
Resources	ArrayList of Resource

**Table 8.** Tasks assigned to a given nurse

TaskId	Description	Time Restrictions
1	Change clothes	Arrival time 8
2	Medication and blood analysis patient 2	28 minutes before 9:00
3	Clean patient 2	between 9:00 and 9:15, 37 minutes
4	Breakfast patient 2	before 10:00, 23 minutes
5	Rehabilitation patient 4	30 minutes
6	Rehabilitation patient 5	30 minutes
7	Rehabilitation patient 6	30 minutes
8	Monitoring and Medication patient 5	after 12 and before 13
9	Monitoring patient 6	after 12 and before 13, 17 minutes
10	Clean patient 7	38 minutes after 10:00 and before 13:30
11	Rehabilitation patient 7	after 14:00, 90 minutes
12	Change clothes	
13	End of the shift	

**Table 9.** Results obtained for the proposed plan

Number in the map	Room Coordinates	Distance	Arrival Time	Finish Time
0		0	8:00	
1	(93, 60)	183	8:03	8:17
2	(32, 30)	377	8:24	8:52
3	(70, 30)	73	8:54	9:37
4	(93, 43)	145	9:39	10:08
5	(27, 37)	23	10:09	10:39
6	(32, 30)	47	10:40	11:10
7	(41, 30)	47	11:11	11:41
8	(51, 30)	65	11:43	12:27
9	(32, 30)	26	12:28	12:41
10	(32, 43)	149	12:44	13:22
11	(93, 43)	132	13:25	15:30
12	(32, 43)	452	15:37	15:51
13	(32, 30)	183	15:51	15:55

**Table 10.** Time (minutes) spent on indirect tasks

	Monitoring	Reports	Visits	Other	TOTAL
Before	167	48	73	82	370
Geodesic Planner	105	40	45	60	250
RPTW Planner	95	37	45	52	229

**Table 11.** Planner comparison

Number of Task	Brute Force		A* Algorithm		Classical Planning		CBPMP		Genetic	
	Time (s)	Distance	Time	Distance	Time	Distance	Time	Distance	Time	Distance
10	237.63	878	0.01572	878	0.02827287	878	4.32	878	3.27	878
15	85652671.1	...	3.3012	1573	5.89119644	1573	6.12	1590	6.13	1605
20	1.5936E+14	...	237.6864	1857	415.257322	1857	8.35	1897	9.31	1952
25	1.016E+21	...	313746.048	...	...	...	9.79	2551	14.12	2510
30	1.7374E+28	...	57101780.7	...	...	...	12.11	2856	21.37	2803