Representation of Prior Autonomous Virtual Agent's Experience by Using Plan and Situation Cases

Haris Supic, *Member, IAENG*

***Abstract*— In this paper we describe an approach to representation of prior autonomous virtual agent's experience. The autonomous virtual agent (AVA) and virtual environment interact at each of a sequence of interaction loops. The AVA selects the appropriate actions based on a previous interaction experience represented in the form of plan and situation cases. Cases include observations of the virtual environment as well as behavior routines that represent the solution component of cases. The AVA that continuously interacts with a virtual environment must be able to autonomously create new situation cases based on its perception of the local virtual environment in order to select the appropriate actions to achieve the goal state.**

***Index Terms*— autonomous virtual agent, case-based reasoning, three-layered cycles, virtual environment.**

I. Introduction

The development of Autonomous Virtual Agents (AVAs) that use prior experience in virtual environments is very helpful in many areas. A number of researchers have studied virtual agents with internal sensory and storage mechanisms [1,2]. In this paper we present an approach of experience-based virtual agent that specifies how an agent represent and store its experiences. The AVA's interaction experience is represented in the form of three different types of cases.

A. Case-based Reasoning

Case-based reasoning (CBR) is a type of reasoning based on the reused past experiences called cases. In general, a case consists of a problem, its solution and an outcome. *Solving a problem by CBR involves obtaining a problem description, measuring the similarity of the current problem to previous problems stored in a casebase with their known solutions, retrieving one or more similar cases, and attempting to reuse the solution of one of the retrieved cases, possibly after adapting it to account for differences in problem descriptions* [3]. The solution proposed by the system is then evaluated (e.g.., by being applied to the initial problem or assessed by a domain expert). The new problem description and its solution can then be retained as a new case, and the system has learned to solve a new problem [3]-[5].

Manuscript received December 10, 2008. This work was supported in part by the Canton Sarajevo, Ministry of education and science under Grant 11-14-31575-11, and Grant 11-14-20308.1/07.

Haris Supic is with the Department of Computer Science, Faculty of Electrical Engineering, University of Sarajevo, Bosnia and Herzegovina, Zmaja od Bosne bb, phone: +(387) 33 250 736; fax: +(387) 33 250 725; e-mail: haris.supic@ etf.unsa.ba.

II. Model of AVA-Virtual Environment Interaction

The interaction model involves the exchange of information between the AVA and its *environment*, where the AVA's response to earlier perception inputs can affect the contents of later perception inputs.

Now, we will briefly describe *the AVA-environment* interaction. The AVA and environment interact at each of a sequence of interaction loops. Each interaction loop includes the following phases:

1. perceive the environment,
2. select a step, and
3. step execution.

Throughout each interaction loop, the AVA receives a perception stimulus, $p_i \in P$, where P is a set of all possible perception stimuli, and on that basis selects an action $a \in A$, where A is the set of all possible actins. The AVA selects an appropriate action by using case-based reasoning and after that the AVA executes the action. One interaction loop later, as a consequence of its previous action, the AVA finds itself in a new situation (see Fig. 1).

A. Steps

A step is any action selected and/or taken by the AVA. According to the interaction model, there are three types of steps as follows:

- *the act of shifting the attention is called the focus of attention,*
- *the act of changing the state of the AVA is called an action step, and*
- *the act of asking for help or defining the new mission is called a step for user intervention.*

Step of focusing of attention. To act in a continuously changing environment, the AVA must be able to react appropriately to changes and unexpected events in the environment. To overcome this problem, we have introduced the notion of the step of focusing of attention. The step of focusing of attention represents the act of moving the attention to the currently relevant attributes of the local environment. By applying this step, the AVA can autonomously create a new situation case that reflects changes in the environment. Formally, a step of focusing of attention f is an n-tuple $f=(A_1, A_2, \ldots A_n)$ where A_i is a perception attribute, $i=1,2,\ldots n$. A perception attribute is a relevant feature of the virtual environment that is important for the AVA's next step selection. Each perception attribute

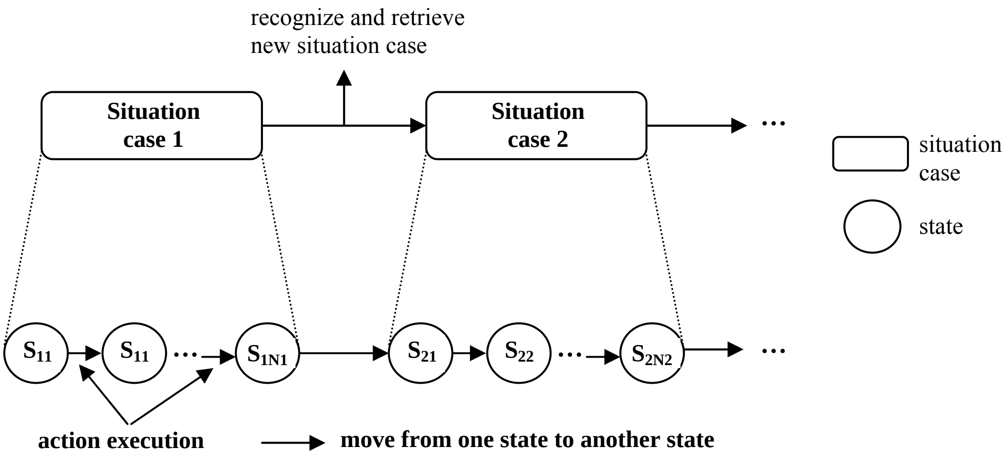

Fig. 1. As a consequence of its previous action, the AVA finds itself in a new situation

A_i takes values from exactly one domain D_i, $i=1, 2, ...n$. The step of focusing of attention is an abstract representation of the AVA's perceptual action. We have used the formalization for describing the representational scheme of situation cases. As an illustration, Fig. 2 shows an example of relevant attributes for autonomous navigation tasks in indoor virtual environments. Illustrated perception attributes constitute the formal representation of the step of focus of attention $f=(R,L,\theta R,\theta L,D)$.

Action step. An action step a is an n-tuple $a = (v_1, v_2,...v_p) \in \boldsymbol{R}^p$ that represents actions that change state of the AVA. For example, the AVA may have the defined an action step a as three-tuple $a=(\Delta x, \Delta y, \Delta\theta)$ where Δx denotes the shift of the AVA in X direction, Δy denotes the shift of the AVA in Y direction and $\Delta\theta$ denotes the shift of the AVA's orientation.

Step of user intervention. A step of user intervention h is represented by string, $h \in H$, where H is a set of all possible steps of user intervention. The domain dependent set of all strings H is predefined in the design phase of the AVA.

B. Perception stimuli

Our interaction model uses the three types of perception stimuli: *directions*, *synthetic perception stimuli*, and *perception stimuli that represent a state of the* AVA.

Directions. Directions represent suggestions to the AVA how to achieve a specified final goal of the AVA. Directions d are represented as n-tuples $d = (d_1, d_2, ...d_i, ...d_n)$, $d_i \in I$, where I denotes the set of all possible intentions. The meaning of the term intention is explained in the next section.

Synthetic perception stimuli. Objects are the building blocks of the 3D virtual environment (see Fig. 3). Examples of objects include walls, doors, floors, tables, chairs etc. The surface is assumed to be flat in the implementation described here. The AVA acquires image data from a scene, and the scene is graphically rendered from the AVA's point of view. We have developed an approximate model for synthetic vision that is suitable for moving through the 3D virtual environment. Synthetic vision generally involves determining which object surfaces in the environment are currently visible to a virtual agent. This problem can be viewed as calculating all visible surfaces from a particular viewpoint given a collection of objects in 3D. We adopt an approach to synthetic vision similar to the one described in [6], and [7].

Synthetic vision stimuli from the virtual environment are generated as a consequence of a previously selected action for focusing of attention. Synthetic perception stimuli are synthesized input n-tuples that represent an integrated description of relevant aspects of virtual environments in the

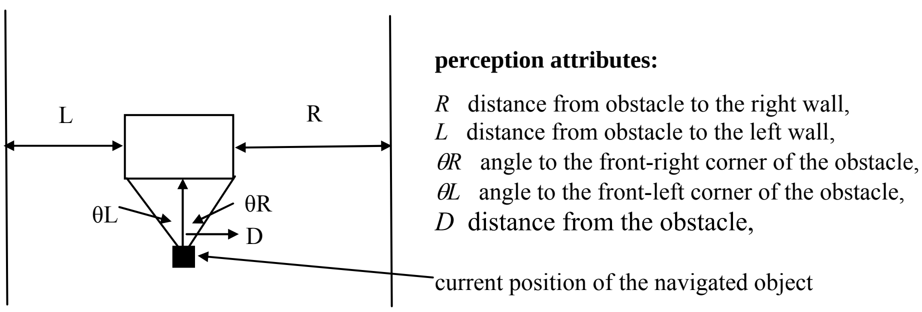

Fig. 2. An 2D illustration of the perception attributes for autonomous navigation tasks in indoor environments.

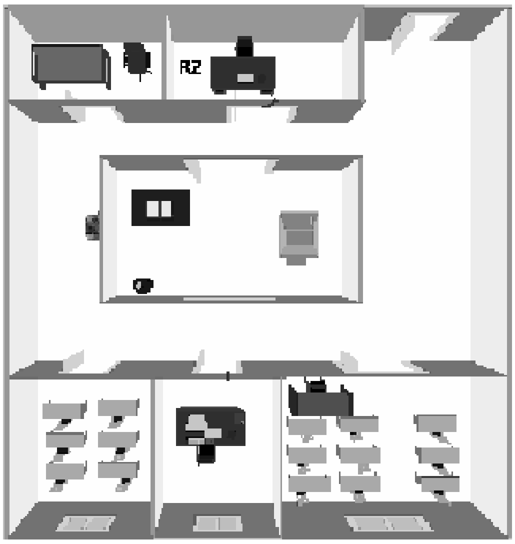

Fig. 3. Virtual Environment

current situation. For example, if the AVA selects an action for focusing of attention $f=(A_1, A_2,..., A_n)$, then a perception generator generates a synthetic perception stimulus $sp_f=(v_1, v_2 ...,v_n)$, $v_i \in D_i$, $i=1,2,...n,$ as a consequence of the action denoted as f and where D_i is a domain of a perception attribute A_i.

Perception stimuli that represent the state of the AVA. This type of perception stimuli is also generated as a consequence of a previously selected action step. For example, in autonomous navigation tasks, the perception stimulus p that represent the state of the AVA can be described by three-tuple $p=(x,y,\theta)$, where x, y and θ denote values that represents x-coordinate, y-coordinate and the orientation of the AVA relative to the X-axes, respectively. Hence, in our interaction model, all information from the virtual environment is defined as perception stimuli.

III. Intentions and Behavior Routines

Intentions. In our model the term intention denotes a determination to act in a certain way or to do a certain thing. Each intention starts a selection of a step of focusing of attention. The AVA needs to reason and decide how to select the next intention and what actions it should take in order to successfully achieve its intention. In our model, there are two types of intentions: *plan intentions* and *contextual intentions*.

Plan intentions and contextual intentions are in hierarchical relationships. Plan intentions are planned in advance at the beginning of the current mission due to the fact that the environment contains certain static structures that do not change over time. This is an opposite of the contextual intentions. A contextual intention represents the AVA's desire that is the most appropriate for the given contextual conditions. Furthermore, a contextual intention tends to promote the fulfillment of a plan intention. The selection of contextual intention depends on the current contextual conditions, but plan intentions can be planned at the beginning of the current mission. Contextual intentions will have to be left unspecified until the situation in which they are required arises and relevant perception attributes of the local environment can be determined by selecting an appropriate step of focusing of attention. For example, a plan intention for autonomous navigation tasks is "*exit-from-room*". Following are some examples to illustrate contextual intentions: "*move-right-to-avoid-obstacle*", "*left-wall-following*" etc. Plan intention "*exit-from-room*" can be planned in advance, but the contextual intention "*move-right-to-avoid-obstacle*" can not be planned in advance. Hence, the intention "*move-right-to-avoid-obstacle*" will be selected when the relevant perception attributes of the local environment can be determined by selecting an appropriate step of focusing of attention.

Behavior Routines. As mentioned earlier in this paper, the AVA does not use case-based reasoning throughout each interaction loop. Instead, throughout certain interaction loops, the AVA routinely selects steps based on behavior routines that are generated by case-based reasoning throughout certain previous interaction loop. We will now introduce the three types of behavior routines: *plan behavior routines*, *contextual behavior routines, and action behavior routines*.

Behavior routines are defined as n-tuples $b=(b_1, b_2, ...b_j, ...b_n)$, $1 \le j \le n$, where the three types of n-tuples are possible. If all elements b_i in n-tuple denoted as b are *plan intentions*, then b is called *a plan behavior routine*. Furthermore, if all elements b_i in n-tuple b are *contextual intentions*, then b is called *a contextual behavior routine*. And finally, if all elements b_i in n-tuple b are *action steps*, in this case b is called *an action behavior routine*.

Notice that plan and contextual behavior routines are composed of intentions, but the action behavior routine is composed of action steps. Different types of behavior routines represent a behavior of the AVA at various levels of abstraction. Plan behavior routines and contextual behavior routines direct the attention to the relevant aspects of the local virtual environment. Action behavior routines provide a selection of the appropriate action steps.

IV. Case-based Representation of the AVA's Interaction Experience

Formally, the AVA interaction experience ex_i until i-th interaction step can be represented by n-tuple $ex_i=(p_1, a_1, p_2, a_2, ...p_i,a_i, ...p_i, a_i)$. The n-tuple ex_i can be defined inductively as:

- $ex_1=(p_1,s_1)$,
- $ex_i=(ex_{i-1}, p_i, s_i)$, $i=2,3,....$

The AVA selects the appropriate actions based on a previous interaction experience. Of course, it is not appropriate to store all perception experience. The approach to representing interaction experience is to extract cases from continuous the AVA-environment interaction. Cases include observations of the virtual environment as well as behavior routines that represent the solution component of cases. In our representation scheme, cases are classified as plan and situation cases [8]. Furthermore, situation cases are classified as contextual and action cases (Fig. 4).

Plan cases are used to support reasoning processes at planning abstraction level. As a result of reasoning processes at plan level the AVA select an appropriate *plan behavior routine* (Fig. 6). On the other side, situation cases are used to support reasoning processes at contextual and action abstraction levels. As a result of reasoning processes at contextual and action levels, the AVA selects appropriate contextual and action behavior routines.

Contextual classes and instances. Most environments contain structures that remain static during the lifetime of the AVA. These static structures represent local environments and each local environment represents a particular context. The AVA that performs case-based reasoning is adequate for the environments that contain identifiable configurations (structures) of the local virtual environment. In this paper, types of identifiable local environments are called *contextual classes*. Furthermore, concrete examples of local environments are called *contextual instances*. This means that such structured environments can be described as a set of contextual instances. To help in understanding the term contextual class used throughout the paper, we show in Fig. 5 a virtual indoor environment. This environment is created using the following contextual classes: *Hallway, Room, L-shaped junction, and T-shaped junction*. Each contextual class is characterized by its specific configuration of relevant perceivable objects in the local environment.

A. Plan Cases

Definition of a plan cases. A plan case c_p is a *three*-tuple $c_p=(d_p, b_p, q_p)$ where

– d_p is a description component;
– b_p is a solution component, and
– q_p is an outcome component.

A description component d_p is an *n*-tuple $d_p = (c_1,c_2 \ldots, c_n)$, $1 \le i \le n$, $c_i \in C$, where C denotes the set of all contextual instances.

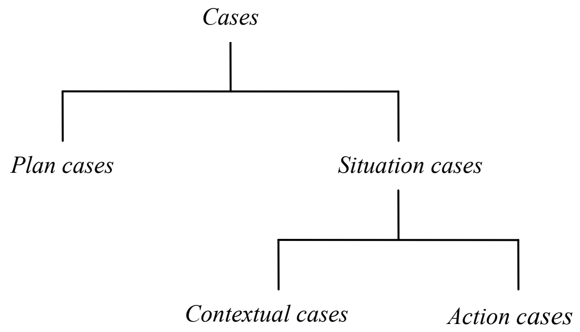

Fig. 4: Classification of cases

A solution component is a plan behavior routine previously described in section II.

An outcome component is an *n*-tuple $q_p=(q_{p1}, q_{p2}, \ldots q_{pi}, \ldots q_{pn})$ where q_{pi} denotes a perception stimulus $q_{pi}=(v_1, v_2, \ldots v_j, \ldots v_k)$ that represents the distinctive states of the AVA. An agent receives this perception stimulus as a consequence of a specific previously selected step of focusing of attention $f_{qp}=(A_{p1}, A_{p2}, \ldots A_{pi}, \ldots A_{pk})$, $i=1,2,\ldots k$.

An example of a plan case is $c_1=(d_p, b_p, q_p)$ where:

- $d_p=(R_1, H_1, L_1, H_2, L_2, R_2)$
- b_p=((*exit-from-room, go-to-end-of-hallway, go-right, go-to-end-of-hallway, enter-to-room, go-straight)*
- $q_p= (q_{p1}, q_{p2}, q_{p3}, q_{p4}, q_{p5}, q_{p6})$, where
- $q_{p1}=((8, 5.5, 1.57, 5.5)$, $q_{p2}=(4,6,3.14,4.5)$, $q_{p3}=(3,8,1.57,2.8)$, $q_{p4}=(3,12,1.57,6)$, $q_{p5}=(4,14,1.05,2)$, $q_{p6}=(5,17,1.05,2)$

Here we assume that the step f_{qp} is defined as $f_{qp}=(X, Y, \theta, \Delta S)$ where

- X, Y and θ denotes perception attributes that represents *x*-coordinate, *y*-coordinate and the orientation of an agent relative to the *X*-axes, respectively;

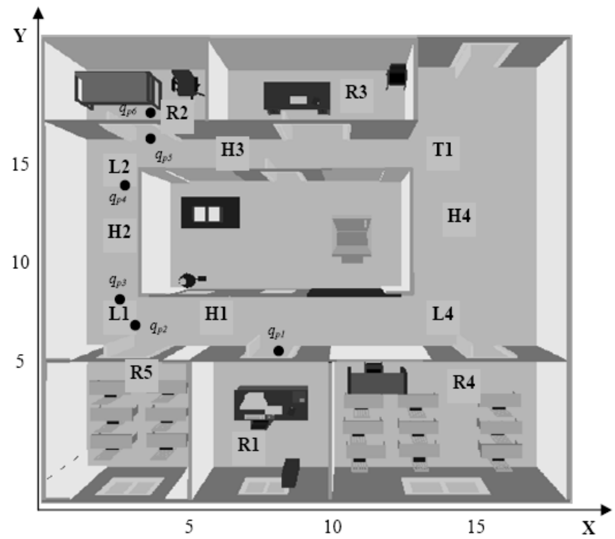

Fig. 5. Examples of virtual indoor environment that contains instances of contextual classes C={Hallway, Room, L-shaped junction, T-shaped junction}.

- ΔS denotes the perception attributes that represents the distance the AVA traversed from one contextual instance to other.

For example, the interpretation of the perception stimulus $q_{p1}=((8, 5.5, 1.57, 5.5)$ *is:* X=8, Y=5.5, θ=1.57, ΔS=5.5.

B. Situation Cases

As stated earlier in this paper, we propose two types of situation cases: *contextual cases* and *action cases*. Both types of situation cases have the same representation structure of the description components.

Description component of situation cases. A description component of situation cases *d* is a *two*-tuple $d=(f, sp_f)$ where

- *f is a step* of focusing of attention*, and*
- sp_f *is a synthetic perception stimulus over the step of focusing of attention f.*

When the AVA wants to create a new situation case, it must select a step *f* that moves the attention to the perception attributes that describes the relevant aspect of the local environments. As a consequence of the previously selected step *f*, the AVA produces selective synthetic perception stimulus. So, the two elements *f* and sp_f form the description component of situation cases. The above definition and definitions presented in Section III allow for a formal definition of situation cases.

Definition of Situation Cases. Let *F* denote a set of all steps of focusing of attention. A situation case is a *three*-tuple $c_s=(d_s, b_s, q_s)$ where

- d_s *is a description component of a situation case ds=(f,* sp_f*), where f is a contextual step of focusing of attention,* $f \in F$*, and* sp_f *is a perception stimulus over the step f,*
- b_c *is a contextual (for contextual cases) or action behavior routine (for action cases), and*
- q_c *is an outcome component of a situation case.*

An outcome component of a situation case is a perception stimulus that the AVA receives as a consequence of a specific previously selected step of focusing of attention $f_q=(A_1, A_2, \ldots A_n)$. Formally, an outcome component of a situation case is $q_s=(v_1,v_2,\ldots v_i \ldots v_n)$, $v_i \in D_i$, $i=1,2,\ldots,n$, where D_i is a domain of a perception attribute A_i.

Therefore, when the AVA wants to complete a current situation case, it must select a step that moves the attention to the perception attributes that describe an outcome of execution of behavior routines. A selection of step of focusing of attention will cause the AVA to produce a perception stimulus that represents the outcome component of a situation case.

Notice that a contextual behavior routine represents a solution component of a contextual case. This component describes how to choose the appropriate contextual intentions in situation described by a description component d_c. Contextual cases include not only a description of the relevant aspects of the local environment, but also information about how to select the appropriate contextual intentions in given situation.

On the other side, an action behavior routine represents a solution component of an action case. This component describes how to choose the appropriate action steps in situation described by a description component d_s. Action cases include not only a description of the relevant aspects of a local environment for next action step selection, but also information on how to select the appropriate action steps.

C. Examples of Situation Cases

Here we show the two situation cases: the contextual case c_c and the action case c_a. We use the following perception attributes: distance from obstacle to the left wall *(L)*, distance from obstacle to right wall *(R)*, distance from obstacle *(D)*, angle to front-right corner of obstacle *(θR)* and distance from current position to right wall *(W)*. Furthermore, let action steps be represented as a three-tuple $a=(\Delta x, \Delta y, \Delta\theta)$ where Δx denotes the shift of the agent in *X* direction, Δy denotes the shift of the agent in *Y* direction and $\Delta\theta$ denotes the shift of the agent orientation. Table I gives the formal specifications for the contextual case $c_c=(d_c,b_c,q_c)$. Table II gives the formal specifications for the action case $c_a=(d_a,b_a,q_a)$. The meaning of the outcome components is determined by steps of focusing of attention to the outcome of behavior routines. We will assume the following steps: $f_{qc}=(\Delta S, \Delta T)$ and $f_{qa}=(\Delta T)$, where ΔS denotes the perception attribute that represents a distance, and ΔT denotes the perception attribute that represents a time interval.

Table I. An example of a contextual case for autonomous navigation in indoor environments

Description, d_c	$d_c = (f, sp_f)$, $f = (L, R)$, $sp_f = (1, 2)$
Solution, b_c	$b_c = (mrao, md)$
Outcome, q_c	$q_c = (5, 1)$

Table II. An example of an action case for autonomous navigation in indoor environments

Description, d_a	$d_a = (f, sp_f)$, $f = (D, \theta, W)$, $sp_f = (1, 1.5, 3)$
Solution, b_a	$b_a = (a_1, a_2)$, $a_1 = (1, 2, 1.2)$, $a_2 = (3, 0, 0)$
Outcome, q_a	$q_a = (4.5)$

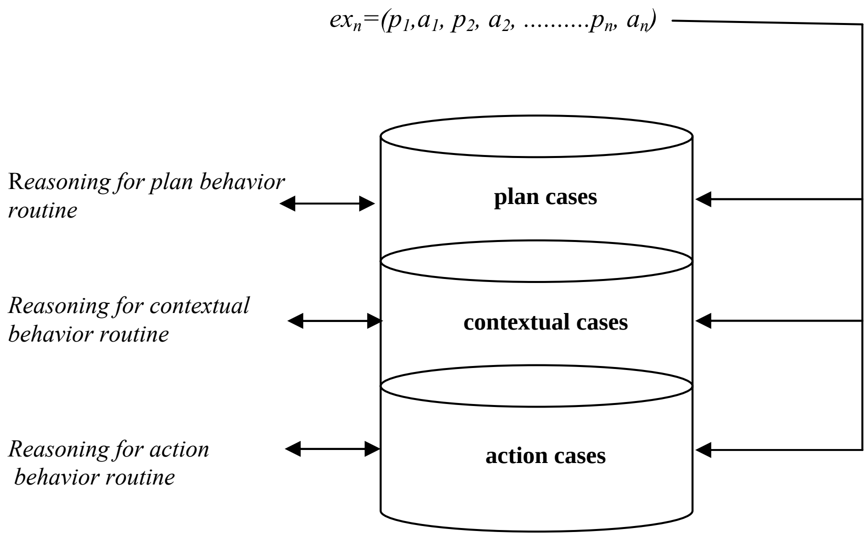

Fig. 6: As a result of reasoning processes at plan, contextual and action levels, an AVA selects an appropriate *plan, contextual and action behavior routine, respectively.* All cases are extracted from interaction experience ex_i.

The outcome component $q_c=(5, 1)$ of the contextual case c_c indicates the traveled distance $\Delta S=5$ by applying behavior routine b_c, and time interval $\Delta T=1$ it takes the agent to travel the distance ΔS. The outcome component $q_a=(4.5)$ of the action case c_a indicates the time interval $\Delta T=4.5$ that agent took to execute the behavior routine b_a. Therefore, the meaning of outcome components of situation cases is determined by the meanings of the perception attributes that constitute steps of focusing of attention to the outcome of behavior routines.

V. THREE-LAYERED CASE-BASED REASONING CYCLES

In this section, we describe three-layered case-based reasoning cycles for AVAs moving through 3D virtual environment. First, we introduce a connectivity function *cf* that is used throughout case-based reasoning cycles. This function connects intentions and actions for focusing of attention. A connectivity function *cf* defines the mapping from a set of all intentions *I* to a set of all actions for focusing of attention F, $cf\colon I \rightarrow F$.

The phases of three-layered case-based reasoning cycles are now described [9, 10].

Create a new plan case: A user defines a new mission of the AVA**.** A new mission is defined as a goal state of the AVA.

Retrieve a plan case: The plan case similar to the new plan case is retrieved from the casebase. Plan behavior routines $b_p=(i_{p1}, i_{p2}, \ldots i_{pn})$ is obtained from the retrieved case and adapted to the new goal state.

Create a new contextual case: From a plan behavior routine the AVA selects a current plan intention i_p. Using a connectivity function, a contextual action for focusing of attention is selected, $f=cf(i_p)$, and the AVA forms a synthetic perception sp_f based on synthetic vision stimuli. In this way, the description component of the new contextual case is created, $d_c=(f, sp_f)$.

Retrieve a contextual case: The contextual case similar to the new contextual case is retrieved from the casebase. Contextual behavior routines $b_c=(i_{c1}, i_{c2}, \ldots i_{cn})$ is obtained from the retrieved case. The behavior routine is adapted to the new conditions.

Create a new action case: Using a connectivity function *cf* an action for focusing of attention *f* is selected and the AVA forms a synthetic perception sp_f based on synthetic vision stimuli. Thus, the description component of the new action case is created, $d_a=(f, sp_f)$.

Retrieve an action case: An action case similar to the new action case is retrieved from the casebase. Action behavior routine $b_a=(a_1, a_2, \ldots a_n)$ is obtained from the retrieved case. The behavior routine is adapted to the new conditions. An action behavior routine b_a is an ordered sequence of actions a_i, $i=1,2,\ldots n$.

Reuse of an action case: In this phase, the action behavior routine $b_a=(a_1,a_2,\ldots.a_n)$ is used by a step by step approach. In this phase, after each selected action a_i, the action behavior routine b_a is evaluated. It must be checked if the behavior routine as a plan to achieve a current contextual intention is adequate. When all actions are selected and executed, the action for focusing of attention to the outcome of the action behavior routine is selected. The AVA receives a perception stimulus that represents the outcome component of the new action case.

Retain a new action case: The new action case is stored in the casebase.

Revise a contextual behavior routine: In this phase, before selection of a new contextual intention i_c from the contextual behavior routine b_c, this routine is revised and eventually adapted. If the contextual behavior routine $b_c=(i_{c1}, i_{c2}, \ldots i_{cn})$ contains unrealized contextual intentions, then the reasoning goes to the phase <*Create new action case*>, else the AVA selects an action for focusing of attention to the outcome of the current contextual routine. As a result of the selected action, the AVA receives a perception stimulus that represents the outcome component of the new contextual case. The AVA's reasoning cycles continue with the phase denoted as <*Retain contextual case*>.

Retain a new contextual case: The new contextual case is stored in the casebase. The AVA selects a next plan intention from the current plan behavior routine b_p, and reasoning cycles continue with the phase denoted *as* <*Create new contextual case*>.

Revise a plan behavior routine: The AVA selects an action for focusing of attention to the outcome of the current plan behavior routine b_p. As a result of the selected action, the AVA receives a perception stimulus q_{pi} that represents the element of outcome component of the new plan case. In this phase, before selection of a new plan intention i_p from the plan behavior routine b_p, this routine is revised and eventually adapted. If the plan behavior routine $b_p=(i_{p1}, i_{p2}, \ldots i_{pn})$ contains unrealized plan intentions, then the reasoning goes to the phase <*Create new contextual case*>, else the AVA selects an action for focusing of attention to the outcome of the current plan behavior routine b_p. As a result of the selected action, the AVA receives a perception stimulus that represents the element of outcome component of the new plan case. The AVA's reasoning cycles continue with the phase denoted as <*Retain plan case*>.

Retain a new plan case: The new plan case is stored in the casebase. The AVA asks a definition of the new mission from a user and reasoning cycles continue with the phase denoted *as* <*Create a new plan case*>.

VI. An Illustration and Experimental Results

A. An Illustration

To help in understanding how three-layered case-based reasoning cycles work, we show one situation from autonomous navigation task. The AVA's planning module generates a current plan intention, based on plan cases. The solution component of plan cases is an ordered sequence of plan intentions. The AVA selects a current plan intention i_p. Assume that the current plan intention is "*exit-from-room"*. This intention directs the AVA's attention to the relevant perception attributes: distance from obstacle to the left wall (L) and distance from obstacle to the right wall (R) (see Fig. 7). Formally, the AVA selects the contextual action for focusing of attention $f=(L, R)$ and forms a synthetic perception $sp_f=(1, 2)$ based on synthetic vision stimuli. Thus, the new contextual case is created $c_c=(d_p, ?, ?)$, $d_p=(f, sp_f)$, where *?* denotes temporarily undefined components. Then, the most similar contextual case is retrieved from the casebase $rc_c=(d_c, b_c, q_c)$, and the solution component b_c is adapted to the new conditions. The adapted solution component of the retrieved case is an ordered sequence of contextual intentions. Assume, that this component is $b_c=(mrao, md)$ where

– *mrao* denotes the contextual intention "*move – right – to – avoid - obstacle"*, and
– *md* denotes the contextual intention "*move- to-door"*.

The AVA selects the intention *mrao* that directs the attention to the relevant perception attributes: distance from obstacle *(D)*, angle to front-right obstacle's corner (θ), and distance from right wall *(W)* (see Fig. 7). Formally, the AVA selects the action for focusing of attention $f=(D, \theta, W)$. Then, the AVA forms a synthetic perception $sp_f=(1.5, 1, 3)$. The action for focusing of attention f and the synthetic perception sp_f are elements of the new action case's description component, $d_a=(f, sp_f)$. Then, the most similar action case is retrieved, $c_a=(d_a, b_a, q_a)$ and the solution component b_a is adapted to the new conditions. The adapted

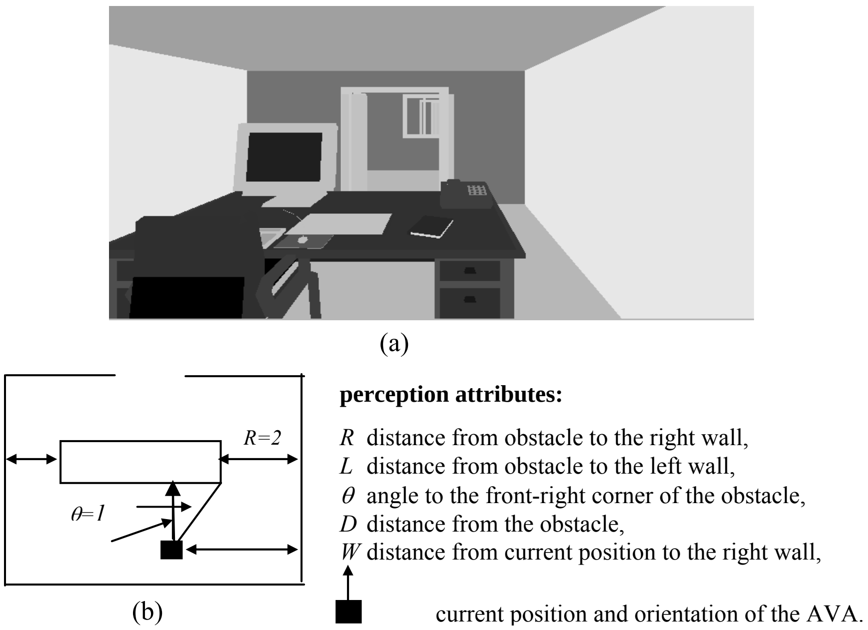

Fig. 7: An illustration of the perception attributes for autonomous navigation in indoor environments. (a) A simulated vision from the current position of the robot (b) 2D representation of the relevant perception attributes for the current situation

solution component b_a is an ordered sequence of actions. The AVA selects the actions from b_a and moves through the virtual environment.

When all actions are selected and executed, the action for focusing of attention to the outcome of the action behavior routine is selected. As a result of the selected action, the AVA receives a perception stimulus that represents the outcome component of the new action case. The new action case is stored in the casebase. Before selection of the intention *md ("moving-to-door")* from the contextual behavior routine b_c, this routine is revised and eventually adapted. When all contextual intentions from the contextual behavior routine b_c are achieved, the AVA selects the action for focusing of attention to the outcome of the current contextual behavior routine. As a result of the selected action, the AVA receives a perception stimulus that represents the outcome component of the new contextual case. The new contextual case is stored in the casebase. Furthermore, the AVA selects a next plan intention i_p from the plan behavior routine b_p, and case-based reasoning cycles are repeated throughout the new contextual instance similarly as previously described for achieving the plan intention "*exit-from-room*".

B. Experimental Results

In order to evaluate the described three-layered case-based reasoning cycles, we have experimented and analyzed mission success rate. The experiments performed with the AVA were done to verify that it could meet certain requirements needed to autonomously carry out a navigational mission in virtual environment. These requirements include: being able to navigate to a designated room; being able to autonomously create new situation cases; being able to avoid obstacles by going around the obstacles, etc.

All of the experiments had the same basic format, though the details of the AVA's starting position and orientation, the positions of obstacles, and the AVA's goal position and orientation. The AVA's positions are given in *X-Y* coordinates. Three different casebase configurations were tried in order to investigate the mission success rate:

1. $|CB_p|=20$, $|CB_c|=30$, and $|CB_a|=30$
2. $|CB_p|=35$, $|CB_c|=50$, and $|CB_a|=50$
3. $|CB_p|=50$, $|CB_c|=80$, and $|CB_a|=70$

where $|CB_p|$, $|CB_c|$, and $|CB_a|$ denote the number of plan, contextual, and action cases, respectively. Each casebase configuration has been tested by using the 30 AVA's navigation missions. Table III shows a summary of experimental results.

Table III. Experimental results

Casebase size	Unsuccesseful missions		Succesful missions	
	Number	Rate	Number	Rate
$\|CB_p\|=20$, $\|CB_c\|=30$, $\|CB_a\|=30$	9	30%	21	70%
$\|CB_p\|=35$, $\|CB_c\|=50$, $\|CB_a\|=50$	2	6,7%	28	93,3%
$\|CB_p\|=50$, $\|CB_c\|=80$, $\|CB_a\|=70$	1	3,3%	29	96,7%

We can see that the experimental results obtained using the three-layered approaches increases mission success rate with increasing number of previously stored cases.

VII. Conclusions and Future Work

The development of Autonomous Virtual Agents (AVAs) that use prior experience in virtual environments is very helpful in many areas. A number of challenges are raised in developing a system incorporaiting AVAs. One of the most important challenges facing today's AVA's development is the appropriate representation and usage of prior interaction experience. In this paper, we have described three-layered case-based reasoning cycles for autonomous virtual agent moving through 3D virtual environment. The AVA selects the appropriate actions based on a previous interaction experience represented in the form of cases. Experimental results obtained using the presented approach increases mission success rate with increasing number of previously stored cases. Much future work remains. We are planning to extend this research and involve different types of virtual environments.

Acknowledgment

The author is grateful for the support by the Ministry of education and science, Canton Sarajevo, B&H.

References

[1] J. Kuffner, and J.C.Latombe, "Perception-based navigation for animated characters in real-time virtual environments", The Visual Computer: Real-Time Virtual Worlds, 1999.
[2] N. Noser, and D. Thalmann, "Synthetic vision and audition for digital actors", In Proc. Eurographics '95, Maastricht, 1995, pp. 325-336.
[3] L. Mantaras, et al. "Retrieval, reuse, revision, and retention in CBR". *Knowledge Engineering Review*, 20(3), 2005, pp. 215-240.
[4] A. Aamodt, and E. Plaza, "Case-based reasoning: Foundational issues, methodological variations and system approaches", in *AICOM*, vol 7(1), 1994, pp. 39-59.
[5] J.L. Kolodner, *Case–based reasoning*. Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1993.
[6] X. Tu, and D. Terzopouls, "Artificial fishes: Physics, locomotion, perception, behavior", In A. Glassner, editor, Proc. SIGGRAPH 1994, Computer Graphics Proceedings, Annual Conferences Series, 1994, pp. 43-50.
[7] D. Terzopoulos, and T. Rabie," Animat vision: Active vision in artificial animals", In Proc. Fifth Int. Conf. on Computer Vision , Cambridge, MA, 1995, pp. 801-808.
[8] H. Supic, "An agent's action selection strategy by using case-based reasoning and simulated vision". In Proc. of World Congress on Engineering, The 2007 International Conference of Computational Intelligence and Intelligent Systems, London, 2007, pp. 138-141.
[9] H. Supic, and S. Ribaric, "Autonomous creation of new situation cases in structured continuous domains", LNCS (LNAI), vol. 3620, 2005, pp. 537-551
[10] H. Supic, "Three-layered Case-based Reasoning Cycles for Autonomous Virtual Agent Moving Through 3D Virtual Environment". In Proc. of World Congress on Engineering, The 2008 International Conference of Computational Intelligence and Intelligent Systems, London, 2008, pp. 1-5.