Ambient Support by a Personal Coach for Exercising and Rehabilitation

Maarten F. Bobbert[1], Mark Hoogendoorn[2], Arthur J. van Soest[1],
Vera Stebletsova[2], and Jan Treur[2]
[1] VU University, Faculty of Human Movement Sciences
[2] VU University, Faculty of Exact Sciences, Agent Systems Research Group

Abstract In this paper an agent-based ambient system is presented to support persons in learning specific movement patterns. The ambient system serves as a personal coach that observes a person's movement pattern, and analyses this based on comparison with an ideal pattern generated by optimisation using a computational musculoskeletal model for this type of pattern minimizing knee joint loading. Based on this analysis the Personal Coach generates advices to adapt the person's pattern in order to better approximate the ideal pattern.
The Personal Coach has been designed using the agent design method DESIRE, thereby reusing an available generic agent model. The system was evaluated (a proof of principle) by setting up an environment in which sensoring of body part positions was incorporated. In evaluations with a few subjects substantial improvement of the movement pattern compared to the ideal movement pattern was achieved.

1 Introduction

Within the area of exercising and rehabilitation, personal coaching can be a crucial factor for success. This holds in particular when patterns of movements have to be learned which are quite complex, or which are hard to perform due to limitations of the person, as in rehabilitation. Only trying the same type of movement pattern over and over again, thereby hoping to learn from experience, may be a very long way to a desired situation or may not even lead to a desired situation at all. Personal feedback from a coach may be essential to make substantial progress. However, giving such feedback is far from trivial, as it has to be well-informed. In the first place a coach has to monitor very well the exact movements over time from the person, where very small differences in timing may already be crucial. In the second place a coach needs knowledge about how the movement pattern can be optimal. Finally, a coach has to be able to point out how the monitored pattern can be changed towards an optimal pattern, by giving important suggestions without overloading the person.

As pointed out above a coach needs to fulfill rather demanding requirements: in observation capabilities, in having knowledge about movement patterns, and in the interaction with the person. Ambient Intelligence is an area in which such types of capabilities needed for personal support are incorporated in the environment in an automated manner; see, e.g., (Aarts, Harwig, and Schuurmans, 2001; Aarts and Grotenhuis, 2011; Riva, Vatalaro, Davide, and Alcañiz, 2005). For example, observation capabilities can be realised using sensor systems, and domain knowledge may be made available in the form of computational models of the human processes considered (e.g., Treur, 2008).

This paper presents an architecture of a personal coach for exercising and rehabilitation and its application to teaching persons a specific target movement pattern to stand up from a chair (henceforth referred to as sit-to-stand, STS, movement (e.g., Doorenbosch, Harlaar, Roebroeck, Lankhorst, 1994; Janssen, Bussmann, Stam, 2002; Yoshioka, Nagano, Himeno and Fukashiro, 2007). The architecture has been designed as a specialisation of the personal assistant agent model described in (Bosse, Hoogendoorn, Klein, and Treur, 2011), using the agent design method DESIRE (Brazier, Jonker, and Treur, 2002). The specific target STS movement was obtained using movement simulation (Casius, Bobbert, Soest, 2004): the motion of a musculoskeletal model was optimized to minimize the peak knee joint moment reached during the movement. Finally, persons were coached to approximate this optimal STS movement using monitored kinematics and ground reaction forces.In the end it was checked whether the peak knee joint moment of the persons had indeed become smaller after coaching.)

2 Overview of the Overall Method Used

The method used involves monitoring a movement pattern by observation and an ideal pattern. The movement pattern is described by values for a number of relevant variables considered, such as the positions of hip, knee, ankle, toe, that can be observed, forces that are exerted, and information that can be derived from this such as angles between different parts of the body, and movement speeds. Let the vectors $\underline{op}(t)$ and $\underline{ip}(t)$ for each time point t be defined by

$\underline{op}(t)$ the person's observed movement pattern
$\underline{ip}(t)$ the ideal movement pattern

The components of these vectors are the values at t of the relevant variables considered.
The values of these vectors are determined as follows. For $\underline{op}(t)$ at a number of points on the body LED markers are attached (e.g., on knee, ankle, hip, toe) that can easily be tracked over time by sensors. Moreover, sensors are used to measure forces exerted on the ground. This sensor information is acquired by the ambient system and stored. The values of the ideal movement pattern are determined by optimising the pattern based on a computational musculoskeletal model. The *deviation pattern* $\underline{d}(t)$ is the difference vector

$$\underline{d}(t) = \underline{op}(t) - \underline{ip}(t)$$

This indicates the deviation of the observed pattern from the ideal pattern. For each time point t in principle a possible advice is to make at time point t the difference between observed and ideal pattern smaller by $\underline{d}(t)$, i.e., by making the new movement pattern $\underline{op}'(t)$ as follows:

$$\underline{op}'(t) = \underline{op}(t) - \underline{d}(t) = \underline{op}(t) - (\underline{op}(t) - \underline{ip}(t)) = \underline{ip}(t)$$

Such an advice could be formulated as: at t change all the positions by $\underline{d}(t)$. However, giving all these advices for all time points t and for all components of the vector would not be realistic. It simply would be too much for the person to follow all these advices. Therefore an important capability of a personal coach is to determine in an intelligent manner a *focus advice set*, which is a limited subset of the set of all possible advices. The idea is that such a focus advice set can be determined based on one or a number of criteria values $c_i(t)$ for the possible advices. Examples of such criteria are:

- *single vs multiple correction*
  indicated by a number between 0 (single) and 1 (all)
  'single' means per time point only one position (e.g., knee position) is corrected
  'multiple' means per time point more positions (e.g., knee, ankle and toe position) are corrected
- *wide vs narrow*
  indicated by a number between 0 (narrow) and 1 (wide)
  'narrow' concentrates on the points in time with highest deviations
  'wide' addresses all deviations
- *early vs late*
  indicated by a number between 0 (early) and 1 (late)
  'early' concentrates on the early part of the time axis
  'late' concentrates on the later part of the time axis
  for example, 0.5 concentrates on the middle area
- *causes vs consequences*
  'causes' may show up earlier than 'consequences' and may preferable as points of correction
- *threshold*
  indicated by a number between 0 (low threshold) and 1 (high threshold)
  only the possible advices are chosen for which the deviation divided by the maximal deviation is above the threshold.

To define the focus advice set, one of these criteria $c_i(t)$ can be chosen, or a subset of them. In the latter case it can be useful to use weight factors and determine an aggregated criterion value by a weighted average $aggc(t) = \Sigma\ w_i c_i(t)$, and only choosing the possible advices with this value above a certain threshold τ: $\mathrm{aggc(t)} \geq \tau$.

3 Finding the Ideal Movement Pattern for the STS Task

In this section it is briefly described how the ideal movement pattern $\underline{ip}(t)$ was determined for the STS task by using a numerical musculoskeletal model and optimisation techniques.

Musculoskeletal model
For simulation of the STS task a two-dimensional forward dynamic model of the human musculoskeletal system was used (van Soest, Schwab, Bobbert, van Ingen Schenau, 1993). The model (see Fig. 1) , which had the time-dependent muscle stimulation *STIM* as its only independent input, consisted of four rigid segments

representing feet, shanks, thighs and HAT (head, arms and trunk). These segments were interconnected by hinges representing hip, knee, and ankle joints, and the distal part of the foot was connected to the ground by a hinge joint. Nine major muscle-tendon complexes (MTC) of the lower extremity were embedded in the skeletal model: m. gluteus maximus, biarticular heads of the hamstrings, short head of m. biceps femoris, m. iliopsoas, m. rectus femoris, mm. vasti, m. gastrocnemius, m. soleus and m. tibialis anterior (Bobbert and Richard Casius, 2011). Each MTC was represented using a Hill-type unit. The MTC model, which has also been described in full detail elsewhere (van Soest and Bobbert, 1993), consisted of a contractile element (CE), a series elastic element (SEE) and a parallel elastic element (PEE).

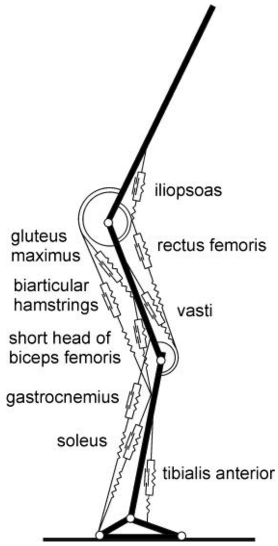

Fig. 1. Model of the musculoskeletal system used for forward dynamic simulations. The model consisted of four interconnected rigid segments and nine muscle–tendon complexes of the lower extremity, all represented by Hill type muscle models. The only input of the model was muscle stimulation as a function of time.

Briefly, behavior of SEE and PEE was determined by a simple quadratic force-length relationship, while behavior of CE was complex: CE velocity depended on CE length, force and active state, with the latter being defined as the relative amount of calcium bound to troponin (Ebashi and Endo, 1968). Following Hatze (1977) the relationship between active state and *STIM* was modeled as a first order process. *STIM*, ranging between *0* and *1*, was a one-dimensional representation of the effects of recruitment and firing frequency of α-motoneurons.

Optimization

The model was put in a standard static initial posture to stand up from a chair, and had to achieve a postureclose to a fully extended standing position, while joint angular velocities were close to zero. . For the application in this paper, the ideal motion was defined as the motion for which the peak knee extension moment was minimal while satisfying the constraints mentioned above. To achieve this, an objective function as described elsewhere was defined (Bobbert and Casius, 2011), incorporating penalties on deviations from the desired final configuration, and the peak knee extension moment. Then the objective function was minimized by optimizing for each of the muscles 4 instants at which *STIM* changed and for each of these the piecewise constant *STIM*-level to which the change occurred. For the optimization, a parallel genetic algorithm was used (van Soest and Casius, 2003). The motion pattern corresponding to the optimal *STIM(t)* solutions was used as the ideal motion pattern *ip(t)*.

4 The Agent Architecture for the Personal Coach

In this section an overview is given of a dedicated generic agent model that can be used as a Personal Coach for performing and training for physical exercising and rehabilitation. It will be illustrated for the process of standing up from a chair as part of a rehabilitation process. The agent uses a computational model of the supported physical process to obtain an ideal way of performing the exercise, as described in Section 2. Moreover, it uses monitoring information of the human actually performing the exercise obtained by sensoring (for more details, see Section 7), in order to analyse what still has to be improved, and to determine which aspect is brought under the attention of the human as an intervention. The model was specified using the component-based agent system design method DESIRE (DEsign and Specification of Interacting REasoning components; see Brazier, Jonker and Treur, 2002) and automatically implemented using the DESIRE software environment, and in a dedicated Matlab version. Below the design of the model is described by the interacting components at different levels of process abstraction. Moreover, for each of the components the generic information types are described that define their input and output.

The process of standing up from a chair, used as an illustration, is monitored by the locations over time of different points of the body, and by forces exerted on the ground. Learning to stand up in the right way can be an important aspect in rehabilitation, for example, to avoid pain and prevent old injuries from coming back. At the top level the agent system consists of two interacting agents: the human, and the Personal Coach, also called agent; see Fig. 2.

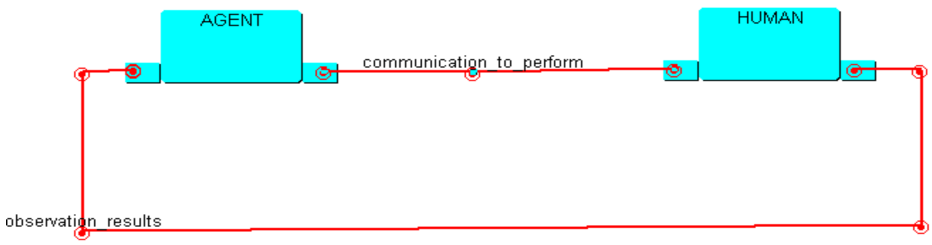

Fig. 2. The agent system: interaction between Personal Coach agent and human

Note that this picture was generated by the DESIRE software environment; the same holds for Figures 3 to 6. The interaction between the two is modelled in the sense that that communication takes place from Personal Coach to human, but not in the opposite direction, and the agent observes the human but not the other way around. Extensions of the model could be made involving also communication from human to agent and observation of the agent by the human. However, the model is kept simple for the purpose at hand. The following describes the generic input and output information types for both components, by showing the generic template for basic statements (atoms) that are used. Examples of instances for the case addressed are the following. It is observed that at time 1 the knee position of person1 is 0:

```
observation_result(personal_info(1, person1, has_x_position, knee, 0.0), pos)
```

The focus advice for person1 to change at time 1 the knee x-position by 1 is communicated by the Personal Coach agent:

```
communicated_by(fa(personal_info(1, person1, has_x_position, knee, 1.0), pos, agent)
```

When looking inside the agent (the box at the right hand side in Fig. 2), a further structure is found as shown in Fig. 3. For the internal design of the Personal Assistant the Generic Agent Model GAM is reused (Brazier, Jonker, and Treur, 2000), from which for the moment the following three the components are adopted (see Fig. 3):

- *World Interaction Management (WIM)*
  handling incoming observation information (about the human performance)
- *Agent Interaction Management (AIM)*
  handling outgoing communication (advices)
- *Agent Specific Task (AST)*
  to determine the advices to be given

Observations and communications as transferred internally are represented as shown above. Beliefs are transferred from World Interaction Management to AST and from AST to Agent Interaction Management. They are represented as follows. It is (positively) believed that at time t person1 has the knee at x-position 0:

```
belief(personal_info(1, person1, x-position, knee, 0), pos)
```

It is believed that a focus advice is to change person1's knee x-position by 1:

```
belief(fa(personal_info(1, person1, x-position, knee, 1)), pos)
```

The components Agent Interaction Management and World Interaction Management can be kept simple. Within World Interaction Management information is extracted from incoming observation results, and incorporated in beliefs. To achieve this in a general manner the following generic knowledge base element can be used:

```
if    observation_result(personal_info(T:TIME, A:AGENT, A2:ATTRIBUTE,
      B:BODY_PART, V:VALUE), S:SIGN)
then  belief(personal_info(T:TIME, A:AGENT, A2:ATTRIBUTE, B:BODY_PART, V:VALUE),
      S:SIGN)
```

Within Agent Interaction Management communications are generated based on beliefs, using the following generic knowledge base element:

```
if    belief(fa(personal_info(T:TIME, A:AGENT, A2:ATTRIBUTE, B:BODY_PART, V:VALUE)),
      pos)
then  to_be_communicated_to(fa(personal_info(T:TIME, A:AGENT, A2:ATTRIBUTE,
      B:BODY_PART, V:VALUE), pos, human)
```

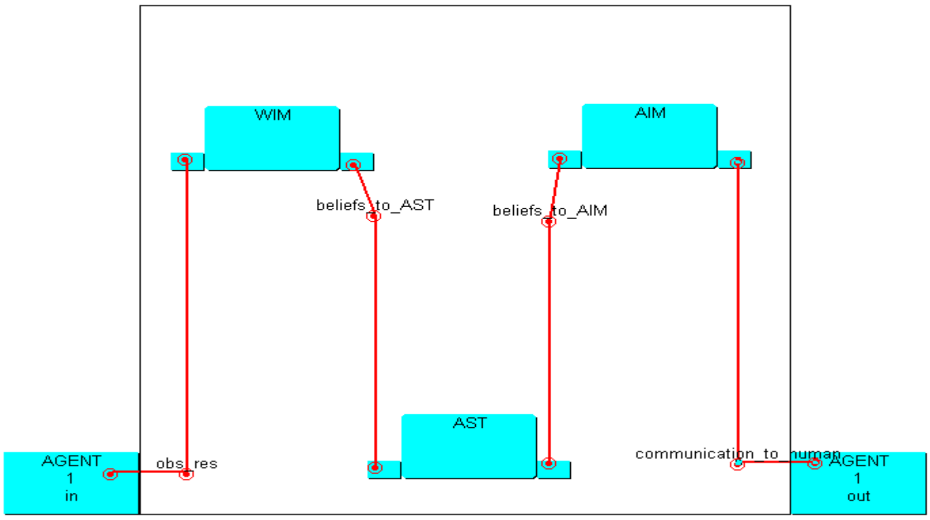

Fig. 3. The internal agent model used for the Personal Coach agent

5 Analysis and Support within the Agent Specific Task

Adopting elements of the reusable model presented in (Bosse, Hoogendoorn, Klein, and Treur, 2011), within the Agent Specific Task two subtasks were modelled (see Fig. 4):

- *Analysis*
  to analyse a performance by the human
- *Support*
  to determine the support to be provided

The information transferred as output from Analysis to input for Support is of the following form.

belief(deviation(T:TIME, A:AGENT, A:ATTRIBUTE, B:BODY_PART, V:VALUE),pos)

An example instance is

belief(deviation(0, human, has_x_position, toe, 1.0), pos)

The analysis component is composed of three components:

- *Descriptive Information Maintenance*
  Here beliefs on the human's observed current performance are maintained (the vector representing the observed movement pattern *op(t)*)
- *Prescriptive Information Determination*
  Here beliefs on the ideal performance are determined (the vector representing the ideal movement pattern *ip(t)*)

- *Assessment*
  Here assessments are done by comparing input from the two other components (determining the deviation vector $\underline{d}(t)$)

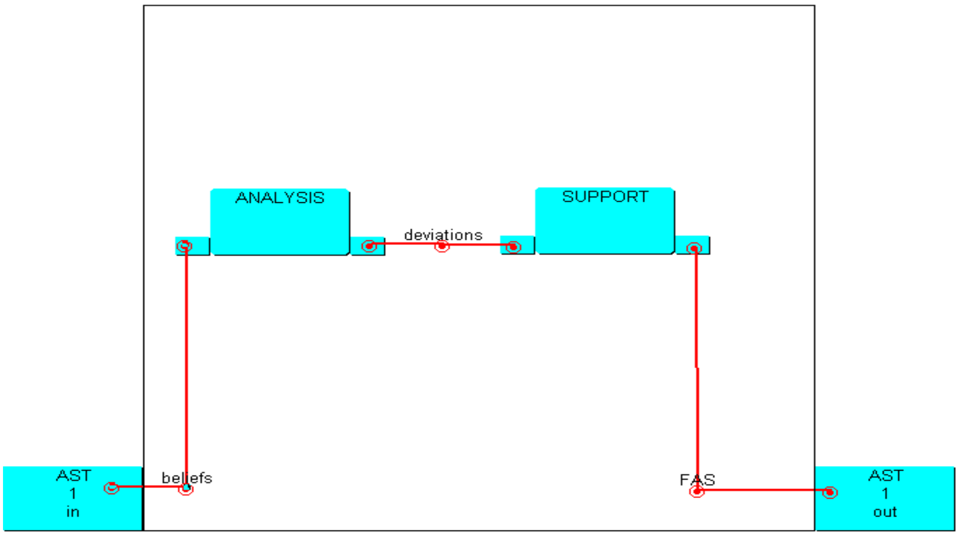

Fig. 4. The Agent Specific Task as composed of an Analysis and Support component

The latter component makes assessments of the performance of the human by comparing input from two other components providing, respectively *descriptive* information (beliefs on the observed current performance) and *prescriptive* information (beliefs on the performance considered as ideal); see Fig. 5.

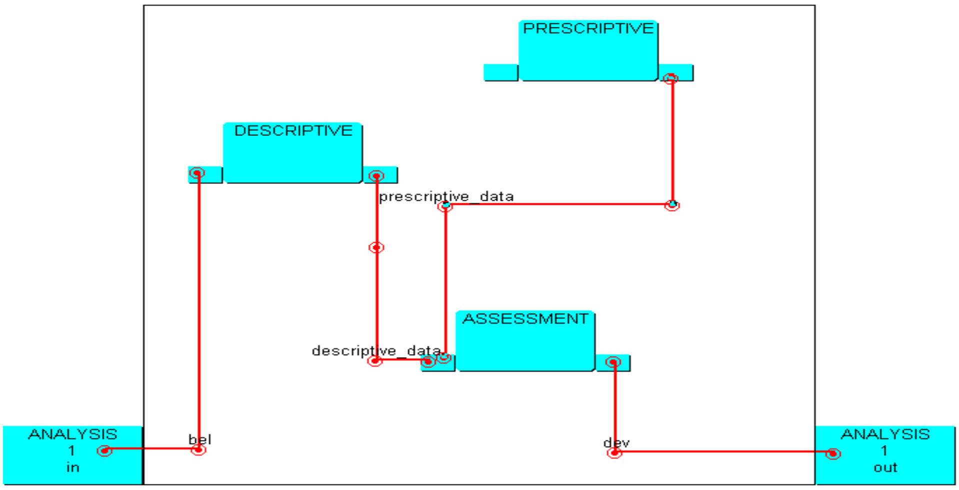

Fig. 5. Analysis component

An example instance of output of Assessment is

```
belief(deviation(0, person1, has_x_position, toe, 1.0),pos);
```

which describes that at time *0* the toe is deviating from the ideal horizontal position by *1.0*. Such output can be generated in Assessment by using the following simple generic knowledge base element:

```
if     belief(personal_info(t:integers,idealperson,A:ATTRIBUTE,B:BODYPART,V1:reals),pos)
 and   belief(personal_info(t:integers,person1,A:ATTRIBUTE,B:BODYPART,V2:reals),pos)
 and   V3:reals = V1:reals - V2:reals
then   belief(deviation(t:integers,person1,A:ATTRIBUTE,B:BODYPART,V3:reals),pos)
```

The component Support receives information about deviations for different body parts at different time points, and determines what advice should be given. Here some strategic choices have to be made, as it will not be very helpful to provide the human with an overwhelming amount of information. The process to determine the advices is composed of two subcomponents (see Fig. 6):

- *Advice Generation*
  providing possible advices based on deviation information
- *Advice Selection*
  providing focus advices as a limited subset of the set of possible advices

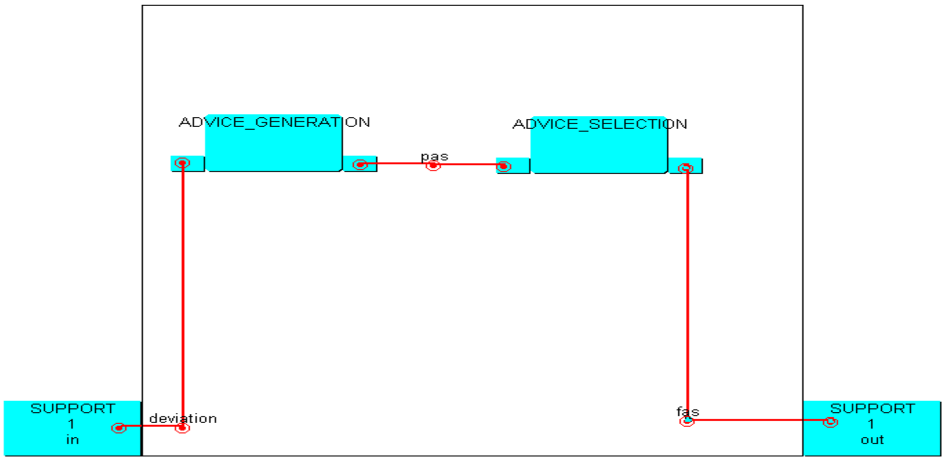

Fig. 6. Support component

For the moment the choice has been made to keep the component Advice Generation simple: for every deviation identified a possible advice (to compensate the deviation) is generated. This was specified by the following generic knowledge base element in Advice Generation:

```
if     belief(deviation(t:integers,person1,A:ATTRIBUTE,B:BODYPART,V1:reals),pos)
then   belief(pa(t:integers,person1,A:ATTRIBUTE,B:BODYPART,V1:reals),pos);
```

The component Advice Selection models a more complex process.

Within Advice Selection a form of filtering of the many possible advices is performed. To this end it is composed of two components (see Fig. 7):

- *Possible Advice Evaluation*
  where each possible advice is valuated (rated between *0* and *1*) for a number of criteria
- *Focus Advice Generation*
  where based on the rates of the possible advices a selection of advices is made

Examples of criteria for which ratings can be determined within Possible Advice Evaluation are:

- *early or late time points of the advices*
  Early provides high ratings for possible advices for the early part of the time axis, late concentrates on the later part of the time axis.
- *higher deviations*
  Provides high ratings for possible advices with highest deviations, and low ratings for those with low deviations
- *longer times of deviations*
  Provides higher ratings for possible advices with deviations above a certain value that last long.

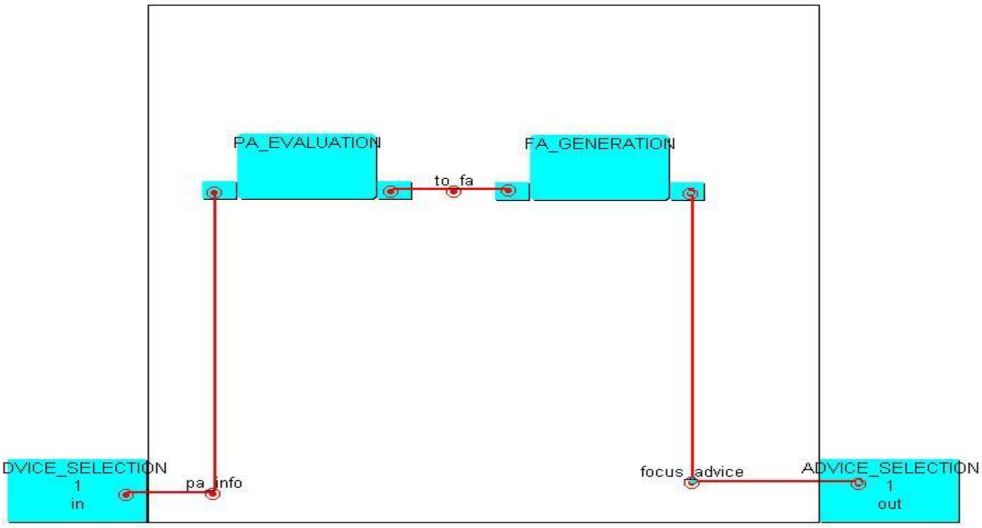

Fig. 7. Advice Selection component

Rating for such criteria can be specified as part of the knowledge base of Possible Advice Evaluation, for example, as follows (here c0 is the first example criterion indicated above, and c1 the second):

```
if      belief(pa(t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals),pos)
  and   R:reals=1/(t:integers+1)
then    valuation(pa(t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals),c0,R:reals);

if      belief(deviation(t:integers, person1, A:ATTRIBUTE, B:BODYPART,V:reals), pos)
  and   belief(totalmax(A:ATTRIBUTE, B:BODYPART,Y:reals), pos)
  and   R1:reals=V:reals/Y:reals
then    valuation(pa(t:integers, person1,A:ATTRIBUTE,B:BODYPART, V:reals), c1, R1:reals);
```

Note that totalmax(A:ATTRIBUTE,B:BODYPART,Y:reals) defines the maximal value that occurs for A:ATTRIBUTE and B:BODYPART, which gets rating *1*, and all other deviations are normalised using this maximal value.

For the third criterion mentioned, first it has to be determined for how long a possible advice lasts. This can be done by introducing the representation

```
pa_duration(D:integers, t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals)
```

expressing that for a duration Dfontsize? starting at time t the possible advice is to reduce the deviation by at least V, and using the following knowledge to generate beliefs about this:

```
if      belief(pa(t:integers,person1, A:ATTRIBUTE, B:BODYPART, V:reals), pos)
  and   belief(minumum_deviation, A:ATTRIBUTE, B:BODYPART, S1:reals), pos)
  and   V:reals ≥ S:reals
then    belief(pa_duration(1, t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals), pos)

if      belief(pa_duration(D:integer, t:integers, person1, A:ATTRIBUTE, B:BODYPART, W:reals),
        pos)
  and   belief(pa(t:integers+D:integer, person1, A:ATTRIBUTE, B:BODYPART, V:reals),pos)
  and   V:reals ≥ S:reals
  and   E:integer = D:integer+1
then    belief(pa_duration(E:integer, t:integers, person1, A:ATTRIBUTE, B:BODYPART, W:reals),
        pos)
```

Given this, a valuation of a possible assumption for the third criterion c2 can be determined as follows:

```
if      belief(pa_duration(D:integer, t:integers, person1, A:ATTRIBUTE, B:BODYPART, W:reals),
        pos)
  and   belief(maxduration(M:integer), pos)
  and   V:reals = D:integer/M:integer
then    valuation(pa(t:integers, person1, A:ATTRIBUTE, B:BODYPART, W:reals), c2, V:reals)
```

Within Focus Advice Generation for any subset of the set of criteria used, the obtained ratings for possible advices can be aggregated to obtain one rating for this subset. In this aggregation process weights are applied for the different criteria used. This is specified for two criteria c0 and c1 in the following knowledge base elements for this component:

```
if    valuation(pa(t:integers,person1,A:ATTRIBUTE,B:BODYPART, V:reals),c0,R0:reals)
and   valuation(pa(t:integers,person1,A:ATTRIBUTE,B:BODYPART, V:reals),c1,R1:reals)
and   belief(weight(c0, W1:reals), pos)
and   belief(weight(c1, W2:reals), pos)
and   W1:reals * R0:reals + W2:reals * R1:reals = R2:reals
then  aggregated_valuation(pa(t:integers,person1,A:ATTRIBUTE,B:BODYPART, V:reals), c0, c1,
      R2:reals);
```

One possibility is to apply such an aggregation to the set of *all* criteria considered. From the overall aggregated ratings, those above a certain threshold can be selected, using the following:

```
if    aggregated_valuation(pa(t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals), c0, c1,
      …, R2:reals)
and   belief(threshold(c0, c1, …, R1:reals)
and   R2:reals > R1:reals
then  belief(fa(t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals),pos);
```

But it is also possible to not aggregate any of the ratings, or only for some proper subsets, and use thresholds in a more differentiated form. For example, for two separate criteria c0 and c1, different thresholds can be used:

```
if    valuation(pa(t:integers,person1, A:ATTRIBUTE, B:BODYPART, V:reals), c0, R0:reals)
and   valuation(pa(t:integers,person1, A:ATTRIBUTE, B:BODYPART, V:reals), c1, R1:reals)
and   belief(threshold(c0, W0:reals), pos)
and   belief(threshold(c1, W1:reals), pos)
and   R0:reals > W0:reals
and   R1:reals > W1:reals
then  belief(fa(t:integers, person1, A:ATTRIBUTE, B:BODYPART, V:reals), pos);
```

6 Evaluation

This section briefly describes the experiments to evaluate the approach and summarizes some of the results.

Experiments

One male subject participated in this study, who had infrared light emitting diodes applied at fifth metatarsophalangeal joint, calcaneus, lateral malleolus, lateral epicondyle of the femur, greater trochanter and acromion. The subject performed the STS task various times, while sagittal-plane positional data of these anatomical landmarks were collected at 200 Hz using an Optotrak (Northern Digital, Waterloo, Ontario) system, and ground reaction forces were measured using a force platform (Kistler 9281B, Kistler Instruments Corp., Amherst, New York). The positional data were used to calculate segment angle time histories and, by numerical differentiation, segment angular velocities. Also, kinematic information and ground reaction forces were combined in an inverse-dynamics analysis (Elftman, 1939) to obtain net joint moments.

The actual coaching with a focus advice set

For the proof of principle that the automatic coach could get the subject to make his motion <u>op</u>*(t)* more similar to the ideal motion <u>ip</u>*(t)*, an advice set had to be specified. According to the literature, the variables in the STS task that have the strongest effect on the peak knee extension moment are initial foot position (Kawagoe, Tajima and Chosa, 2000), speed of movement execution (Pai and Rogers, 1991) and hip flexion (Doorenbosch, Harlaar, Roebroeck, Lankhorst, 1994). Therefore the following focus advice set was used: (1) difference between <u>op</u>*(t)* and <u>ip</u>*(t)* in initial angle of the lower legs, normalized for the peak value in <u>ip</u>*(t)* and weighted by *0.9*, (2) difference between <u>op</u>*(t)* and <u>ip</u>*(t)* in peak angular velocity of the upper legs, normalized for the peak value in <u>ip</u>*(t)* and weighted by *0.5*, and (3) difference between <u>op</u>*(t)* and <u>ip</u>*(t)* in minimal angle of HAT reached during the motion, normalized for the minimum angle of HAT reached in <u>ip</u>*(t)* and weighted by *0.7*. After each trial, the largest of the three elements in the focus advice set was used to give an advice to the subject; this feedback was provided within 15 seconds after completion of the trial. During postprocessing of the experimental data,In the end, the maximum knee joint moment was calculated for each of the STS movements, and in particular it was determined if in the last <u>op</u>*(t)* this maximum knee joint moment was reduced relative to that in the first <u>op</u>*(t)*.

Results

Fig. 8, taken from (Aarts et al., 2011) shows an example of how, over a series of six trials, the subject was able to reduce the values in the advice set. The solid black markers indicate the variables for which the focus advice was given by the coach after the different trials. For example, after trial 1 the advice was focused on the peak thigh angular velocity, and after trial 2 on the minimal HAT angle.

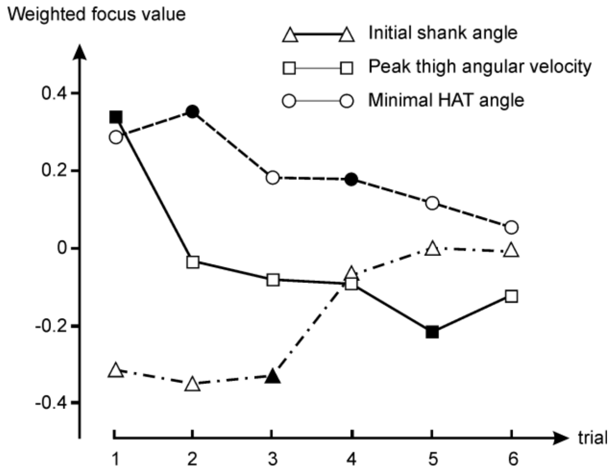

Fig. 8. Focus variables as a function of trial number during an example learning process consisting of 6 STS movements. The solid black markers indicate advices given.

Fig. 9, also taken from (Aarts et al. 2011) shows that the behavioral change shown in Fig. 8 was in fact accompanied by a reduction in peak knee extension moment from almost *200*Nm to roughly between *50* and *100*Nm. Note that the peak knee extension moment did not decrease monotonically. One reason for this may be that the ideal pattern and its initial situation were generated using a standard set of parameter values, i.e. without using a subject-specific set of parameter values. Another reason for this may be that focus variables used in this application do not fully determine the peak knee extension moment.

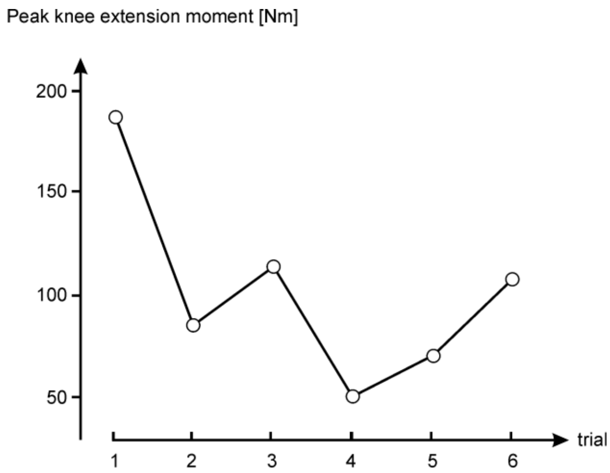

Fig. 9. Peak knee extension moment as a function of trial number during an example learning process consisting of 6 STS movements; see also Fig. 8.

7 Discussion

The agent-based ambient system presented in this paper supports persons in learning specific movement patterns. It serves as a personal coach that observes and analyses a person's movement pattern. The analysis is done by comparing it with an ideal pattern which is generated by optimisation using a computational model for this type of pattern. The analysis is used by the Personal Coach to generate advices to adapt the person's pattern in order to better approximate the ideal pattern. The Personal Coach has been designed using the agent design method DESIRE (Brazier, Jonker and Treur, 2002), thereby reusing available generic agent models (Brazier, Jonker and Treur, 2000; Bosse, Hoogendoorn, Klein, and Treur, 2011).

Based on the agent model designed in the DESIRE environment, an experimental setup was developed. In addition to the DESIRE design environment, this setup makes use of Matlab and Optotrak and the Kistler force plate for the sensoring of body part positions and forces, respectively. Different strategies for focusing were incorporated in different experiments that were conducted, one of which was described in Section 6. Alternatives for presentation of advices by the Personal Coach were offered in text, in speech, or in visualised pictural form. In these experiments a substantial improvement of the movement pattern in the direction of the ideal pattern was found, which was accompanied by a decrease in the criterion variable, i.e. peak knee extension moment. Further experiments with a larger number of subjects will be needed to draw statistically sound conclusions about the effectiveness of the coach and the way in which this effectiveness depends on specific choices made in focusing and presentation.

Acknowledgements

The authors are grateful to Lars Aarts, Kaj Emanuel, and Ingmar de Vries for providing the empirical data discussed in Section 6.

References

1. Aarts, L. Emanuel, K., and Vries, I. de (2011) Een hulp bij het opstaan: Project geautomatiseerde coaching (in Dutch). Report, VU University Amsterdam, Faculty of Human Movement Sciences.
2. Aarts, E., Harwig, R. , and Schuurmans, M. (2001). Ambient Intelligence. In: P. Denning (ed.), The Invisible Future, McGraw Hill, New York, pp. 235-250.
3. Aarts, E., Grotenhuis, F.: Ambient Intelligence 2.0: Towards Synergetic Prosperity. Journal of Ambient Intelligence and Smart Environments 3, 3–11 (2011)
4. Bobbert, M.F., Casius, L.J. (2011) Spring-like leg behaviour, musculoskeletal mechanics and control in maximum and submaximum height human hopping. Philos Trans R Soc Lond B Biol Sci. 2011; 366(1570):1516-29
5. Bosse, T., Both, F., Gerritsen, C., Hoogendoorn, M., Treur, J., (2012) Methods for Model-Based Reasoning within Agent-Based Ambient Intelligence Applications. Knowledge-Based Systems 27, 190–210.
6. Bosse, T., Hoogendoorn, M., Klein, M.C.A., and Treur, J. (2011) An Ambient Agent Model for Monitoring and Analysing Dynamics of Complex Human Behaviour. Journal of Ambient Intelligence and Smart Environments 3, 283-303
7. Brazier, F.M.T., Jonker, C.M., and Treur, J., (2000). Compositional Design and Reuse of a Generic Agent Model. Applied Artificial Intelligence 14, 491-538.
8. Brazier, F.M.T., Jonker, C.M., and Treur, J. (2002). Principles of Component-Based Design of Intelligent Agents. Data and Knowledge Engineering 41, 1-28.
9. Casius, L.J., Bobbert, M., van Soest, A.J. (2004). Forward Dynamics of Two-Dimensional Skeletal Models. A Newton-Euler Approach. Journal of applied biomechanics 20, 421-449.
10. Doorenbosch, C.A., Harlaar, J., Roebroeck, M.E., Lankhorst, G.J. (1994). Two strategies of transferring from sit-to-stand: the activation of monoarticular and biarticular muscles. J Biomech. 27, 1299–1307.
11. De Morree et al (2006), Inspannigsfysiologie, oefentherapie en training, 158, Bohn Stafleu van Lochum, Houten.
12. Ebashi, S., Endo, M. (1968). Calcium ion and muscle contraction. Prog Biophys Mol Biol. 1968;18:123-83.
13. Elftman, H. (1939). Forces and energy changes in the leg during walking. Am J Physiol. 1939;125(2):339-56.
14. Hatze, H. (1977). A myocybernetic control model of skeletal muscle. Biol Cybern. 1977 Jan 20;25(2):103-19.
15. Janssen, W.G.M., Bussmann, H.B.J., and Stam, H.J. (2002). Determinants of a sit-to-stand movement: a review. Physical Therapy 82(9), 866---879.
16. Kawagoe, S., Tajima, N., Chosa, E., et al. (2000). Biomechanical analysis of effects of foot placement with varying chair height on the motion of standing up. J Orthop Sci. 5, 124–133.

17. Pai, Y.C., Rogers, M.W. (1991). Speed variation and resultant joint torques during sit-to-stand. Arch Phys Med Rehabil, 72, 881–885.
18. Riva, G., Vatalaro, F., Davide, F., Alcañiz, M. (eds.) (2005). Ambient Intelligence. IOS Press.
19. Sadri, F. (2011) Ambient intelligence, a survey. ACM Computing Surveys 43, 1–66.
20. Treur, J., (2008). On Human Aspects in Ambient Intelligence. In: Proc of the First Intern. Workshop on Human Aspects in Ambient Intelligence. Published in: M. Mühlhäuser, et al. (eds.), Constructing Ambient Intelligence: AmI-07 Workshops Proceedings. Communications in Computer and Information Science (CCIS), vol. 11, Springer Verlag, 2008, pp. 262-267.
21. Winter, D.A. (1990). Biomechanics and motor control of human movement, Wiley & Sons edition, New York.
22. Yoshioka, S., Nagano, A., Himeno, R., and Fukashiro, S. (2007). Computation of the kinematics and the minimum peak joint moments of sit-to-stand movements. BioMedical Engineering OnLine, 6(26).
23. van Soest, A.J., Schwab, A.L., Bobbert, M.F., van Ingen Schenau, G.J. (1993). The influence of the biarticularity of the gastrocnemius muscle on vertical jumping achievement. J Biomech. 1993 Jan;26(1):1-8.
24. van Soest, A.J., Bobbert, M.F. (1993). The contribution of muscle properties in the control of explosive movements. Biol Cybern. 1993;69(3):195-204.
25. van Soest, A.J., Casius, L.J. (2003). The merits of a parallel genetic algorithm in solving hard optimization problems. J Biomech Eng. 2003 Feb;125(1):141-6.