

# SUPPORTING CONCEPTUAL PRODUCT DESIGN BY HYBRID SIMULATION OF USE PROCESSES WITH SCENARIO STRUCTURES

Wilhelm F. (Wilfred) van der Vegte and Imre Horváth <sup>1</sup>

<sup>1</sup> Delft University of Technology, Faculty of Industrial Design Engineering

## ABSTRACT

The approach described in this paper aims to offer designers a new way to investigate use processes of products by integrating scenarios of expected user behaviour with simulations of physical artefact behaviour. Use is considered a key process in the life cycle of a product, being the phase in which an instance of a product is put into service or applied for its purpose. Our approach aims at resolving three issues: (i) dealing with the diversity of use processes in behavioural simulations, (ii) the integration of simulation and modelling approaches and (iii) extending the deployability of behavioural simulations towards conceptual design.

Currently, options for behavioural simulation of the use process of a product are limited. Performing complete-picture behavioural simulation in which the product and the human user react on each other's behaviours is not practicable. To make this possible, a hybrid approach is proposed. Behaviour that is commonly modelled based on the laws of physics is simulated as continuous behaviour, while information-processing behaviour is simulated as discrete behaviour. This paper elaborates on modelling and simulation of discrete behaviour and linking it to continuous-behaviour simulation. Scenario structures are introduced to represent knowledge about different human decision-making patterns that influence the courses of a use process. Depending on what is available to the designer, these can be based on observations from real users or on conjecture. The objective is to make what-if studies possible to compare different scenarios of product use. This is demonstrated with a pilot study of a basic use process.

*Keywords: Designing for use, scenarios, hybrid simulation, state machines, conceptual design, diversity of use processes, human decision-making, human-artefact interaction.*

## 1 INTRODUCTION

Simulations have been defined as experiments performed on models [1]. They make it possible to investigate life-cycle processes, such as use (or manufacturing, maintenance, etc.) before a product is available in its final form. This paper focuses on behavioural simulation of the use process of a product during its conceptual design. In that early stage of design, virtual prototypes (non-real, digital prototypes modelled and visualized using computers [2]) are often used as simulation models rather than real, physical prototypes. In product design, a prototype is typically a model of the product. However, a use process does not only involve a product. In use, typically three key actors come together: the human user, the product and the surrounding environment [3]. To perform comprehensive simulations of use, prototypes representing this system, for short human-product-surroundings system or HPS system, are needed rather than just product prototypes. This paper introduces a method and system framework for the creation and simulation of *virtual HPS system models*. It aims at resolving two knowledge-related issues in use-process simulation: (i) the diversity of use processes and (ii) the integration of distinct simulation and modelling approaches. These two issues are elaborated in the next two subsections. In section 2, related work is discussed. Section 3 introduces a reasoning model and a concept for simulating HPS systems. Section 4 describes the method of implementing this concept into a workflow and its embedding in the design process. In section 5 the concept of *scenario structures* is elaborated, which is used to model and simulate human decision-making. Section 6 describes the proposed system architecture for modelling and simulation. Section 7 describes a pilot implementation to test the concept of scenario structures and finally sections 8 and 9 present the conclusions and plans for future work.

## 1.1 The diversity of use processes

When designers want to anticipate usage of a not-yet existing product, one of the key issues they are facing is the fact that ‘future usage’ is not just one process but a multitude of possible processes. Yet these different processes share certain elements. Our assumption is that based on common elements the diversity can be structured and made manageable. The variety in use processes can be considered from three aspects: (i) variety in artefacts, (ii) variety in humans and (iii) variety in the courses of action. The variety in artefacts appears primarily due to the diversity of products and surroundings in which products can be used. The variety of humans manifests itself in size and dimensions [4], but there is also a wide variety in the way humans cognitively process information. Human information processing is interlinked with the third aspect, i.e., the different possible courses of action [5]. Researchers in this area have characterized use processes as problem-solving processes. Newell and Simon [6] introduced a theory of human problem solving that has been adopted in cognitive psychology [7]. Its application to the use of products can be explained as follows [8]: the goal of use is to reach a solution to a problem. Performing the problem-solving task is similar to moving through a maze, from the initial state to the goal state. Each junction has various paths representing state-transforming operations. From each junction the user selects one operation, its execution causing a change of state. This maze has also been called ‘scenario tree’ (Figure 1) because it represents multiple *scenarios* for a user with a given product in given surroundings, a scenario being one possible way to use the product, i.e., one course through the maze [9].

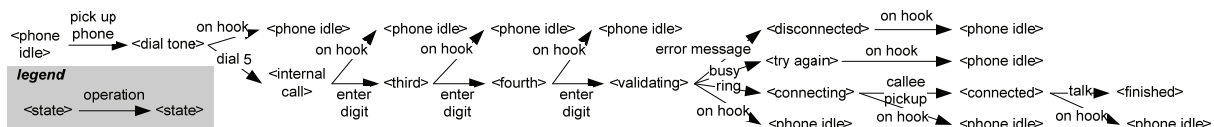


Figure 1. Use of a telephone: partial scenario tree, adapted from [9]

Selection between available operations is based on perception and exploration. Since various paths are possible from each junction, multiple use processes are possible from a given initial state, some of which may not lead to a solution of the problem. Such unwanted result of a use process is sometimes attributed to ‘human error’. According to Reason [10] however, the design should not permit erroneous operations in the first place. Comprehensive use-process simulation can help to identify such operations and anticipate them in the product’s design. For that purpose, the work in this paper aims at enabling designers to explore possible use scenarios and play with different human decision-making patterns in simulation-based what-if studies.

## 1.2 Integration issues in simulation and modelling

The majority of current systems for simulating use focus either on artefact behaviour without taking into account the role of the human user, or on certain ergonomical human aspects without considering artefact behaviour. Artefact-simulation packages are commonly used in engineering. Covering particular areas of physics, they allow investigating behaviour of products and surroundings. Examples are simulations of rigid-body mechanics [11] and finite-element simulations to investigate various physical phenomena [12]. In a review of simulation approaches, we found that the integration of artefact simulation approaches is impeded by differences between the modelling techniques that underlie the various simulation approaches. Key challenges are the integration of physics phenomena into multiphysics simulation, and the integration of discrete and continuous simulation into hybrid simulation [13]. Human behaviour is usually investigated using ergonomics software, which typically focuses on aspects of kinematics, field of view, and static loads [14]. Simulations that include other aspects of human behaviour, such as cognition and motion control, can be found in areas other than product design or in specialized areas of product design [15]. The integration of simulation approaches described in this paper concerns three aspects: (i) artefact/engineering aspects and human/ergonomics aspects, (ii) the various different types of artefact behaviour, and (iii) the various different types of human behaviour in simulation.

## 2 RELATED WORK

Several approaches for combined modelling and simulation of human and artefact behaviour have been proposed and put into practice over the last decades. Approaches to modelling and simulation of human information processing together with physical behaviour are typically tailored to a specific application. For instance, Šuc and Bratko [16] presented a simulation approach for human steering control during bicycle-riding. However, the model of human control behaviour is only qualitative and dedicated to one product: a bicycle. A more versatile approach is offered by the commercial software package Endorphin by NaturalMotion Ltd., for simulating rigid-body mechanical behaviour of human manikins and artefact models that can be imported from CAD. Simulation of human information processing is restricted to low-level motion control based on genetic algorithms [16]. There are various approaches that can be considered related because they use finite state machines to represent human and/or artefactual information processing. The most common application field for these approaches is man machine interaction, where finite state machines are used to model and simulate information-processing behaviour of machines (e.g., [18], [19]) and/or human cognition and control (e.g., [20], [21]). However, these approaches do not include investigation of continuous physical processes. Another approach based on finite state-machine simulation of human decision-making by Martins et al. [22] includes continuous behaviour, but only in the form of predefined sequences of animation frames rather than in a form based on knowledge of the laws of physics.

## 3 CONCEPTUAL ELABORATION OF THE HUMAN-PRODUCT-SURROUNDINGS SYSTEM AS A HYBRID SYSTEM

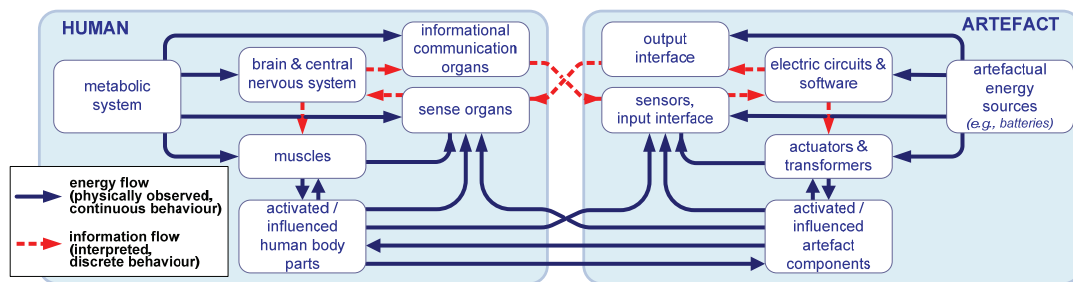


Figure 2. Reasoning model: a simplified representation of a human-artefact system and the interaction between its components.

Figure 2 shows a reasoning model of use as a process in which humans and artefacts (i.e., products and surroundings) interact. The model is similar to a diagram of human-machine interaction in [2]. It is based on the assumption that a HPS system can be considered a hybrid system, i.e., a system in which the various behaviours are investigated either as they can be *physically observed* as energy flows, or as they can be *interpreted* as information flows, disregarding the physical phenomena on which they are based. Observed physical behaviour and interpreted physical behaviour are also known as continuous behaviour and discrete behaviour [23]. The concept underpinning the method and system framework presented in this paper is the concept of a resource-integrated human-product-surroundings scheme. A resource-integrated model is a combination of (i) a use state machine (USM) which is a behavioural model that is used for the simulation of discrete behaviour and (ii) a nucleus-based object model (NBOM) of the HPS system that is used for the simulation of continuous behaviour. The USM represents interpreted physical behaviours (‘information processing’) performed by the HPS system. It is a *finite state machine*, i.e., a mathematical construct to describe behaviour of discrete systems. In use processes we distinguish two forms of interpreted behaviour that can be simulated as discrete systems. The first one is decision-making performed by the human brain. Based on what the sense organs perceive around the human, the brain decides to activate muscles as well as communication organs. Actually, modelling brain functions has been the original purpose of state machines in the 1940s [24]. The second common form of interpreted physical behaviour in use processes is the operation of software and digital circuits in artefacts that control actuators and output interfaces. This paper focuses on integrating the first type of interpreted behaviour into simulation, namely human decision-

making. The part of the USM that represents human decision-making is called *scenario structure*. It is further explained in section 5.

The NBOM that is used for continuous simulation is a composition of nucleus-based models of humans, products, and surroundings. Nucleus-based modelling [25] allows geometric and physically-based representation of humans and artefacts in the same manner so that their observed physical behaviour can be simulated integrally. To provide algorithms for calculating observed physical behaviour from input values which the NBOM receives during simulation, physics knowledge is built into the HPS system model during its creation. Input values for two types of parameters are distinguished: (i) initial parameters, which define the situation in which the HPS system is at the start of the simulation and (ii) actuator parameters, which are imposed upon the NBOM during the course of the simulated use process by the USM. In a typical use process the actuator parameters refer to displacements, angles, and forces exerted by human actuators (muscles controlled by the human brain) and by artefactual actuators (controlled by digital circuits and software).

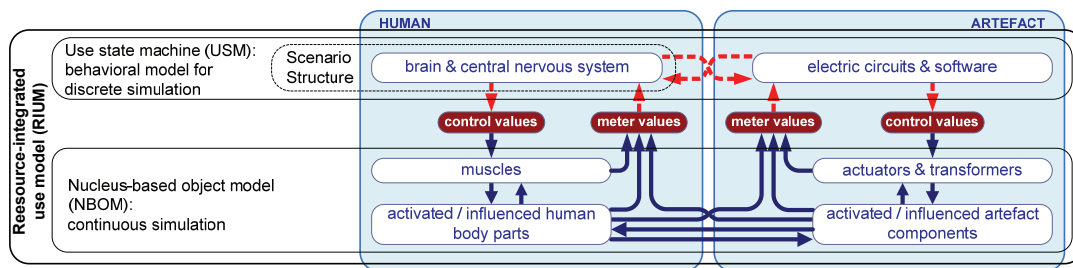


Figure 3. Exchange of simulation variables between the USM and the nucleus-based virtual model of the HPS system shown in Figure 2.

By combining simulation of interpreted and observed physical behaviour, hybrid simulation of the HPS system is accomplished. Discrete and continuous simulations are running in parallel on separate simulation engines that are mutually exchanging values. In accordance with the terminology used in commercial simulation software, parameters in the virtual continuous system of which the values are ‘read’ by the USM are called *meters*. Based on logic defined in the USM, its simulation engine reacts on specified changes in meter values received from the continuous simulation, and it transfers updated actuator parameter values to the continuous simulation engine as shown in Figure 3. Together, the NBOM and the USM are called *resource-integrated use model*. Comparing the simulation model with the reasoning model in Figure 2, the following further simplifications were made:

- The behaviour of energy sources (human metabolic system, batteries, etc.) is not simulated. The assumption is that there is sufficient energy to make the other components operate.
- Sensory input to electronics/software and to the brain is bypassed. It is assumed that electronics/software and the human brain receive all the information needed to make decisions. The process of translating physical variables to interpreted variables, i.e., to information, is not simulated.
- Processes performed by output interfaces of electronics/software and by human communication organs are not simulated. Translation processes between interpreted variables and physical ones are bypassed.

#### 4 EMBEDDING IN A DESIGN-PROCESS CONTEXT

The basic activities in use-process prediction are (i) USM modelling, (ii) NBOM modelling, (iii) linking the NBOM and the USM and (iv) hybrid simulation. After simulation, the USM and/or the NBOM can be revised for a next simulation run. Typical reasons to revise the USM are (i) to redefine the state machine to include different human decisions, or (ii) to revise the state machine that represents the software or electronics in the product. Typical reasons for revising the NBOM are (i) to improve the product concept, (ii) to select virtual humans with different characteristics or (iii) to include other surroundings. This paper focuses on the USM; for an elaboration of nucleus modelling the reader is referred to [26].

Modelling and simulating USMs enriches the conventional workflow in two ways. Firstly, it is an extension of the functional specification of the product, which describes intended operation of the prod-

uct. The USM specifies the expected interpreted physical behaviour of the user and the surroundings (including foreseen erroneous usage) and enables simulation, for checking these behaviours with the operation of the product concept. The second enhancement is that it allows modelling and simulation of interpreted physical behaviours of the product. The NBOM offers three improvements to the conventional workflow. First, it offers the possibility to model the product with the human and the surroundings as one system, and to simulate its observed physical behaviour integrally. Secondly, simulation of nucleus-based models has the advantage over conventional simulation methods that it can handle multiple different physical phenomena simultaneously. Thirdly, nucleus-based modelling allows the designer to work with imprecise and incomplete models in conceptual design. Integrating simulation of the USM together with the NBOM, finally, gives the opportunity to investigate observed and interpreted behaviour simultaneously.

## 5 HYBRID SIMULATION WITH SCENARIO STRUCTURES AND USE STATE MACHINES: ELABORATION OF THE KEY CONCEPTS

In this section we will explain how scenario structures and USMs are composed, and how they are used by algorithms for hybrid simulation. This is done based on key notions such as ‘situation’, ‘setting’ and ‘event’ which will be explained first.

Like any other process the use process is a sequence of changes in a system. Simulation predicts the changes based on a model of the system, in our case the HPS system. The continuous simulation of the NBOM predicts continuous changes based on the laws of physics. Such changes are confined to a *situation*. A situation can be defined as a sub-process completely controlled by a particular subset of the physics laws. Additionally, muscles or actuators may be active during the situation according to a behaviour pattern prescribed from the beginning of that situation. In a use process, situations can end in two ways: (i) through a natural transition to a different set of physics laws and (ii) through a change in the behaviour pattern of muscles or actuators, which are controlled by the human brain or by electronics/software. An example of the first case is the series of transitions a falling object undergoes when it bounces. Such a succession of situations can be predicted through continuous simulation. The second type of transitions needs input from decision-making process simulation – in our case USM simulation.

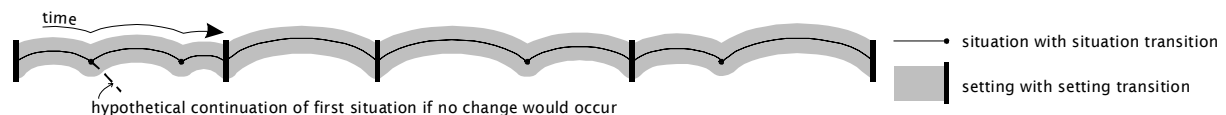


Figure 4. Transitions in a use process

Figure 4 is a graphical interpretation of a use process with transitions depicted as ‘bumps’ in the timeline. A sub-process that takes place from one decision-based intervention until the next intervention is called a *setting*. It consists of one or more *situations*. A situation is an interval during which the operation of the system can be simulated based on the laws of physics. Decisions, and thus transitions between settings are triggered by *events* that can be observed in the continuous behaviour.

As a simple example of the use of a product involving (human) decision-making, consider the pedal bin shown in Figure 5a. Table 1 gives a qualitative description of the events, settings, decisions, and situations.

### 5.1 The scenario structure: a use state machine for human decision-making

When we try to depict the example use process in the same fashion as Figure 4, it appears that some situations and settings apply to a part of the HPS system, while other sub-processes apply to other parts – e.g., the bouncing of the litter inside the bin may continue while the lid is being closed. Since multiple situations and/or settings can be active in parallel, a branched arrangement of settings and situations is needed rather than the linear arrangement of Figure 4. This is shown in Figure 5b.

In the above description it is assumed that the litter lands inside the bin. However, for certain starting positions and motion patterns of the human hand, the litter may fall outside the bin, and thus the event to trigger setting 6 does not take place. For this special case an alternative intervention can be defined. Figure 6a shows an extension of Figure 5, introducing exclusive branching to include an alternative course of human-decision making. If at setting 6 the garbage object crosses the top plane of the bin

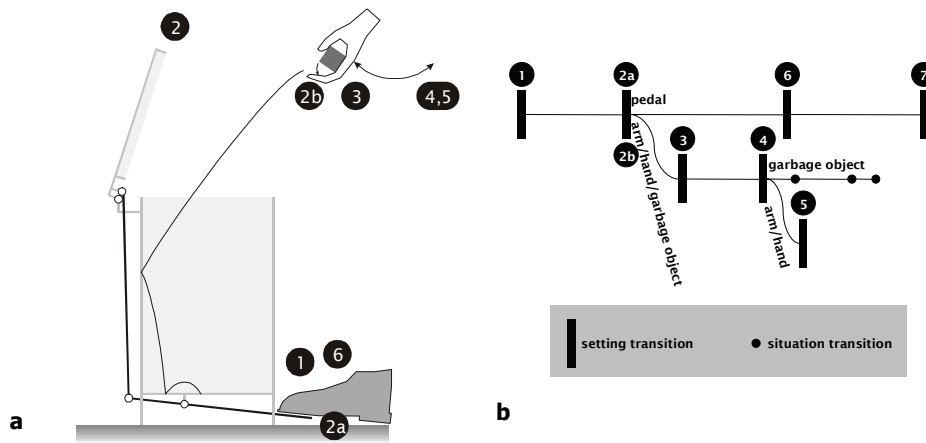


Figure 5. a. Pedal bin.  
b. Transitions between settings and situations in its intended use process

Table 1. Settings and situations in the use of a pedal bin

event	setting transition	decision-based intervention	situations included in setting after transition
t=t0	1	start pressing foot down	various subsets of physics laws involving mechanical behaviour of lid-opening mechanism
lid open	2a	stop pressing foot down	mechanical equilibrium in lid-opening mechanism
	2b	start opening hand	litter sliding down hand
fingers stretched	3	initiate throwing motion	litter sliding down hand
			litter receives horizontal velocity
litter leaves hand	4	reverse throwing motion	litter in freefall
hand back in start position	5	stop reverse throwing motion	hand in rest
litter enters bin	6	release foot	various subsets of physics laws involving mechanical behaviour of lid-opening mechanism
lid closed	7	stop moving foot	sequential succession of subsets of physics laws determining bouncing of litter inside bin and coming to a standstill
		end use process	

inside the perimeter of the bin, the use process finishes as in Figure 5. If it crosses the same plane outside the bin, the alternative path is taken. This is similar to the concept of alternative paths in the scenario tree of Figure 1. The alternative path loops back to setting 1 and includes bringing the hand to the litter, picking it up, and retrying. The diagram that connects the transitions is no longer a line or a tree that is read left-to-right, but a *network* that, if the situation transitions are omitted, can be considered a *finite state machine*.

A finite state machine formally describes a system that is always in at least one of a finite set of states. It defines transitions between states as a reaction on inputs. In our network, the *states* correspond to *settings*. The choice for one of the existing state-machine representations, such as state transition diagrams [27], Petri nets [28], and statecharts [29] is arbitrary. For instance, Figure 6b illustrates how the network in Figure 6a resembles a Petri net. Also note the resemblance to the scenario tree in Figure 1, which is actually another finite state machine representation. Extending the scenario-tree concept, we call the diagrams in Figure 5 and 6 *scenario structures*, which we use to describe human-decision-

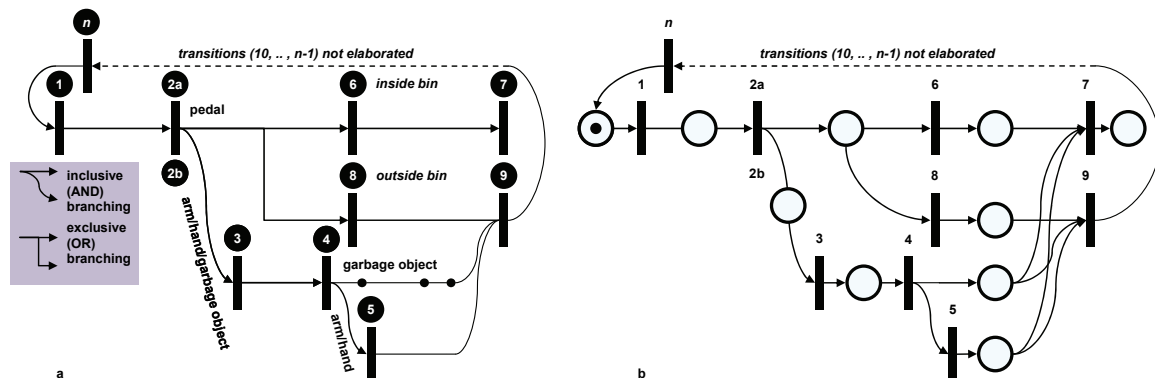


Figure 6. a). Transitions between settings for use of the pedal bin in Figure 5 including an alternative course of the process in case the garbage object misses the bin.  
b). Petri-net representation of the same network of transitions.

making in use processes. A complete use state machine USM can also incorporate decision-making by programmed artefacts, modelled with the same technique. For a USM to be used in computer-based simulation of interpreted physical behaviour, a formal definition has been presented in [30].

## 5.2 Algorithms for hybrid simulation of use state machines

In resource-integrated simulation we distinguish simulation time and calculation time. Simulation time corresponds to happenings in the virtual HPS system that is being simulated, while calculation time corresponds to happenings in the system that performs the simulation. When, in calculation time, the algorithm that simulates the USM receives an event, it evaluates if the event is associated with any outgoing transitions of states that are active at that point in time. For all combinations of outgoing transitions and events for which this is true, the concerned transitions are taken, and the USM assumes the states for which these are the ingoing transitions. Consequently, the actuator control instructions associated to these states are transferred to the NBOM simulation. If an incoming event is not associated to any outgoing transition of current states, the USM does not react. From the moment the USM receives an incoming event until it outputs the actuator control instructions that follow from it, simulation time does not advance. From the moment a new set of states is entered until a next event comes in, simulation time advances as long as the physical processes simulated outside the USM need until they trigger the occurrence of the next event. Thus, the simulation time is determined only by the continuous simulation. The first event in use-process simulation is the start event which is generated by the system user. It triggers the system to start the simulation of the USM. Simulation stops if a final state is reached that is defined in the USM. Events other than the start event are derived from meter values produced by the continuous NBOM simulation. To interlink the continuous simulation and the discrete simulation, an interfacing algorithm generates events based on meter values. Linking the discrete simulation and the continuous simulation also involves converting control instructions from the USM (i.e., time-dependent function descriptions) to control values for the continuous simulation (i.e., the actual values of these functions for the current simulation time step). Figure 7 shows a flowchart of hybrid simulation and an allocation of information-processing functions among the three key algorithms involved in simulation. In the next section, the main software components of the overall system which incorporates these algorithms are discussed.

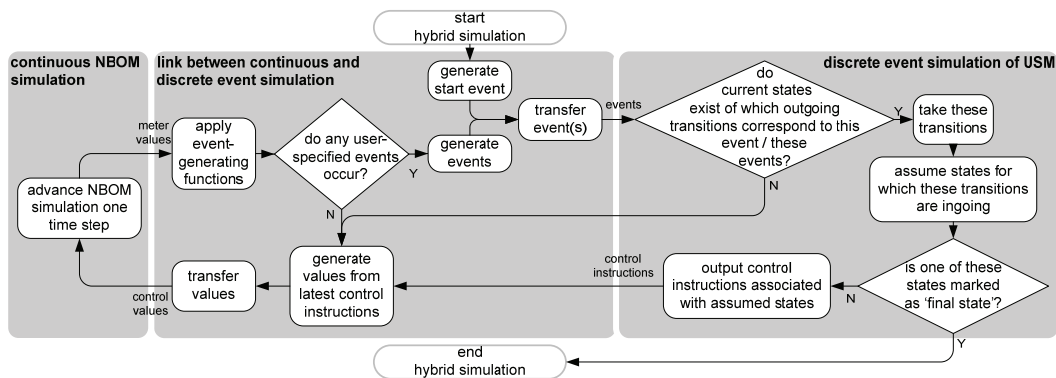


Figure 7. Flowchart of hybrid simulation involving algorithms for continuous simulation, linking, and discrete-event simulation

## 6 SYSTEM ARCHITECTURE AND SOFTWARE COMPONENTS

A system that supports creation and simulation of resource-integrated use models should offer modelling functionality, simulation functionality, and user-interface functionality. Modelling involves NBOM modelling, USM modelling, and defining the connections between the NBOM and the USM so that their behaviours can be simulated as the behaviour of one hybrid system. Simulation involves continuous simulation, discrete-event simulation, and connecting the two simulations by generating events and controlling actuator parameters. The user interface functionality enables the user to provide input for modelling and to initiate simulation runs, and provides output to the user in the form of modelling feedback and simulation results.

The proposed system architecture is subdivided into three modules, (i) the nucleus module, (ii) the hybrid resource integration module, and (iii) the use state machine module. Each of these modules deals with different aspects of the modelling, simulation, and user-interface functionality of the system as shown in the matrix in Figure 8.

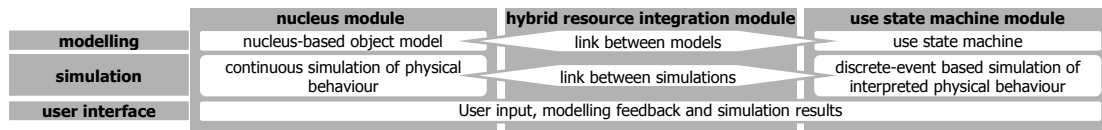


Figure 8. Overview of the system architecture: main software components and functions.

## 7 PILOT IMPLEMENTATION: SIMULATING A SCENARIO STRUCTURE

Based on existing commercial software packages a tentative system was implemented to test the concepts of modelling and simulating scenario structures, and linking them to continuous simulation. As a provisional substitute for nucleus-based object modelling and continuous simulation MSC VisualNastran4D (vN4D) was used, a commercial software package for modelling and simulation of kinetic mechanical behaviour in systems consisting of rigid-body components. Two components of the commercial MATLAB software package were used to model and simulate the USM and the hybrid resource integration: (i) Simulink Stateflow and (ii) Simulink elements for building conventional block diagrams, respectively. Figure 9a shows the basic HPS system that was simulated to test the tentative pilot system: a human arm attempting to throw a piece of litter into a litterbin. The only interpreted physical behaviour investigated is human decision-making; therefore the complete USM is to be considered a scenario structure. The model shown in the picture is further discussed in section 7.2.

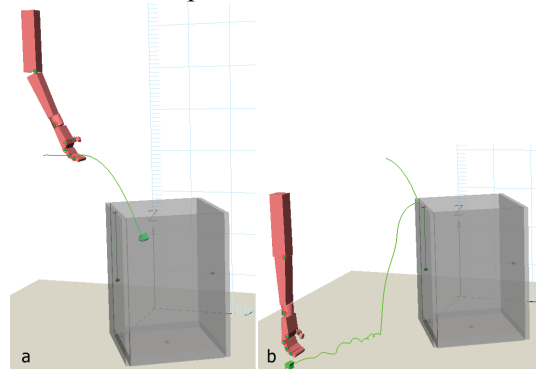


Figure 9 a). Human-product-surroundings system that was used to test the tentative pilot system, modelled and simulated with MSC VisualNastran4D. b). Failed use of the litterbin

### 7.1 Modelling the scenario structure

Figure 10 shows the scenario structure (or USM) that was created with Simulink Stateflow to describe how human decision-making controls the motions of the arm. Stateflow uses a variant of the statechart notation established by Harel [29], [31].

At its highest level the scenario structure in Figure 10 contains seven sub-processes that can run in parallel. These are represented by numbered containers inside the outermost bounding box. The sub-process named ‘MainControl’ describes the main decision-making process that starts with throwing (state: ‘Throw’). If the litter has been released from the hand and has reached a certain distance from the fingertips, the arm goes back (state: ‘GoBack’) to its original position. If the litter lands inside the bin, the end state ‘InsideBin’ is reached. If the litter has landed outside the bin and has almost stopped moving, the hand moves to the position of the litter as a preparation for picking it up and trying again with a different throwing speed (states for moving along three axes: ‘xDisplacement’, ‘yDisplacement’, ‘zDisplacement’). Modelling and simulating the process of picking up turned out difficult as vN4D’s continuous simulation algorithms do not cover flexible hand deformations. Therefore, the USM ends here: ‘StopForNow’. A retry loop for throwing at a different speed can be included once



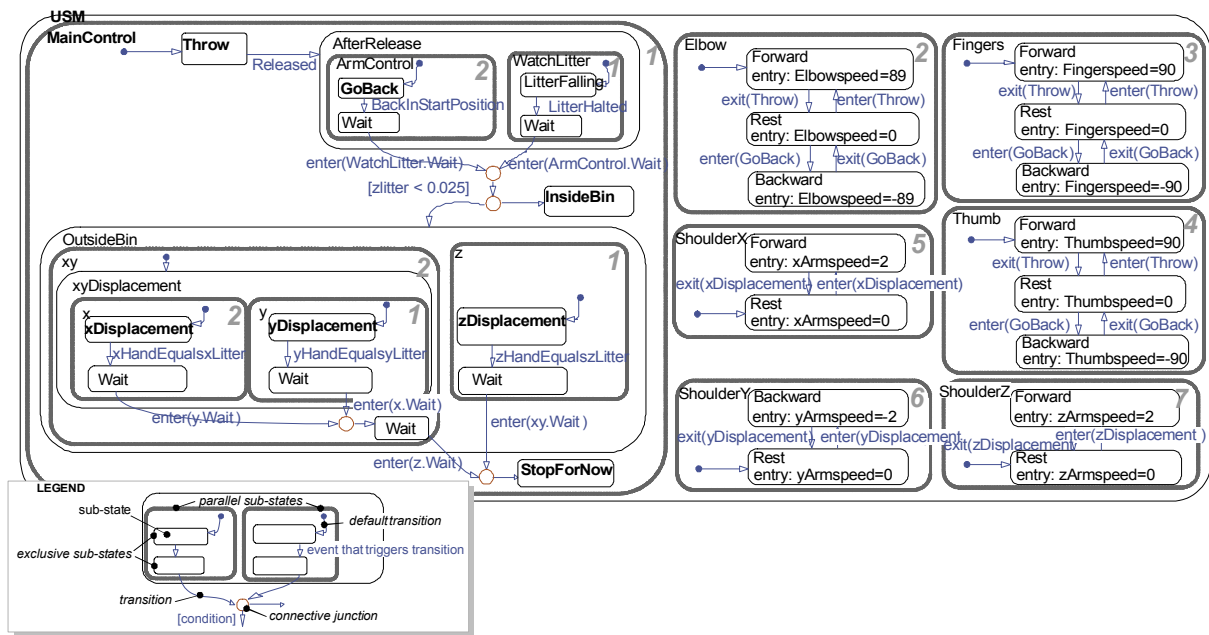


Figure 10 Scenario structure modelled with Simulink Stateflow

nucleus-based simulation of continuous behaviour has been implemented, and gripping can be modelled and simulated more adequately. There are six sub-processes that activate rotations of the lower arm, the fingers, and the thumb as well as translations of the shoulder. They are linked to the ‘Main-Control’ sub-process through events that are internally defined in the statechart.

## 7.2 Continuous simulation modelling

The object model for simulation of the continuous mechanical behaviour of the HPS system shown in Figure 9 was created using the basic solid-modelling functionality of vN4D. Apart from rigid bodies representing the geometry of the HPS system, other elements that have been defined in vN4D are joints connecting human limbs, actuators representing muscles and controls, and meters for linking to the discrete simulation. Controls are inputs to be defined for each actuator, and meters are outputs that can be defined as distances between specific points on rigid bodies, angles between bodies, velocities, etc. A so-called *vNPlant* is defined to link a Simulink model to a vN4D model through a user-specified selection of its controls and meters.

In simulating the use of the litterbin, it was not the goal to give an accurate representation of human geometry or to achieve realistic simulation of human muscle behaviour and human kinetics. It is assumed that knowledge from other research can be used at a later stage to build more realistic models.

## 7.3 Linking the models

Figure 11 shows how the link between the models is realized with block-diagram elements (depicted against shaded background). They connect the ‘vNPlant’ to a block representing the scenario structure (or USM). The linking block-diagram elements convert scenario-structure output as control values for the continuous simulation. They convert meter values from the continuous simulation so that these can be used as input to the scenario-structure simulation. Events are generated based on detection of specified threshold values for (combinations of) continuous output variables. Events defined based on a combination of multiple meter values (e.g., subtraction of coordinates to express a distance) are pre-processed by block-diagram elements (e.g., subtraction of values). The resulting value is evaluated by a ‘hit crossing’ block sending occurring events with their specified names to Stateflow. Additionally, values of meters that are used to evaluate logical conditions are passed on to the scenario structure directly.

## 7.4 Running hybrid simulations

Once the object model and the scenario structure have been modelled and connected, hybrid simulations can be initiated from the Simulink user interface. During the simulation, vN4D animates the behaviour of the HPS system, while Stateflow animates the scenario structure by highlighting the active states and transitions. By varying the throwing speeds in the scenario structure, different scenarios can

be simulated. Some end with the litter at the bottom of the bin and the arm back in its original position (Figure 9a shows an intermediate snapshot of such a scenario), while at the end of other scenarios the piece of litter is on the floor outside the bin with the arm just above it, ready to pick it up (Figure 9b).

## 8 DISCUSSION

In this paper the concept of use state machines and hybrid resource integration has been elaborated to a level that made it possible to create a hybrid model and test it in a basic application example. In the example a scenario structure is linked to a continuous simulation and tested on a use process of a basic human-product-surroundings system. Of course it would be more interesting to test the concept with more complex products that allow for a large diversity of use scenarios with different possible outcomes. It is expected that hybrid simulation of the use of such products will be possible once a full implementation of nucleus-based object modelling and simulation is available, and a wider range of observed physical behaviours can be included in continuous simulation. For now, the tentative pilot implementation could confirm that it is possible (i) to apply a scenario structure as a networked model of human-decision-making, which controls human motion patterns in a continuous simulation, and (ii) to link the results of the continuous simulation to the scenario structure so that they influence the decision making. By using exclusive branching in the scenario structure, different scenarios could be simulated with different outcomes.

It can be argued that, even for the simple processes that have been modelled here, the scenario structure shown in Figure 10 is already quite complex, and that it is questionable whether designers are willing to go through the necessary modelling effort during conceptual product design. However, since it is a formal model, it is also an unambiguous model. This means that once a model has been created, it can be used to preserve knowledge about product usage and it can relatively easily be reused in similar projects. Therefore, in a design environment where similar products are frequently (re)designed, most of the work (with the exception of possible refinements) is a one-time effort. The complexity of the scenario structure also depends on the chosen representation form, and it has to be noted that the Stateflow representation was chosen based on its availability and on its direct connectivity to other simulations. Perhaps using another finite state machine representation makes simpler models for the same scenario structure possible. For instance, part of the complexity in Figure 10 is caused by the need to synchronize outgoing transitions from multiple states. With Petri nets, synchronization can be modelled in a more straightforward way [32]. Finally, it must be mentioned that probably not all the possibilities of modelling with Stateflow have been used to their full potential. One idea that seems promising is to prepare modular sub-routines of human decision-making and muscle control that can be reused in different simulations (regardless whether the products involved are similar or not), so that

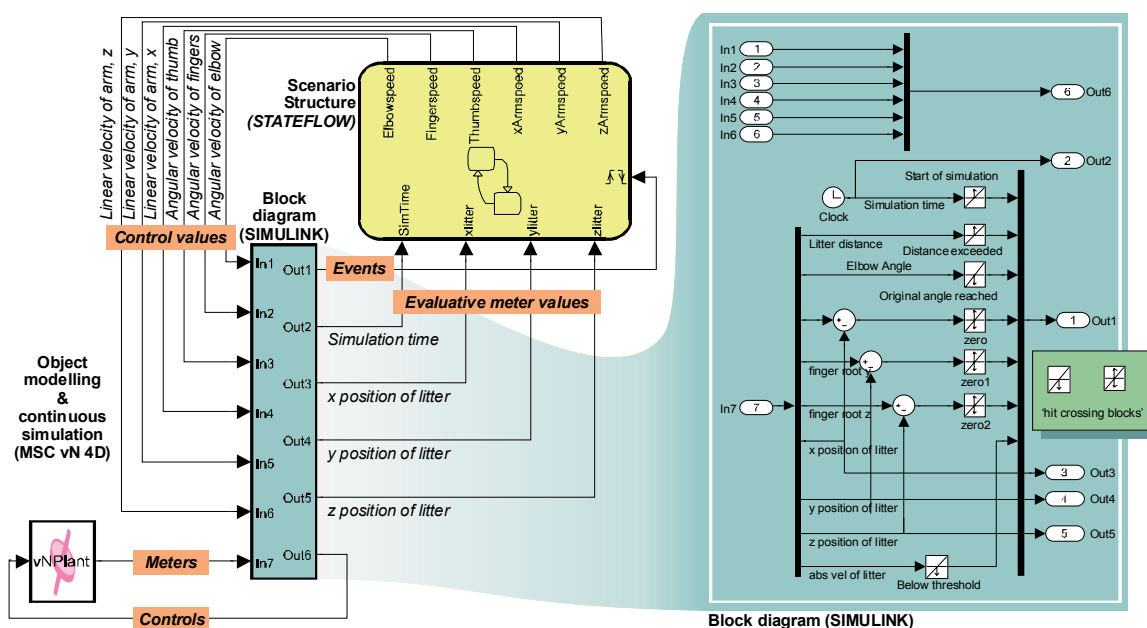


Figure 11 Hybrid simulation model

the modelling effort can be further reduced. Candidate subroutines for such modules would be common motion patterns, such as throwing something in a given direction, arranging the human body into a specified posture, or gripping an object. For the preparation of such modules, results from additional research into human motion patterns are needed.

## 9 CONCLUSIONS AND FUTURE WORK

Modelling scenarios of human decision-making during the use of products as scenario structures based on the theory of finite state machines makes it possible to represent use not only as a linear sequence of interactions, but also as parallel sub-processes, including alternative paths. Thus, multiple options for the process can be represented. Scenario structures offer a formal means to capture knowledge about anticipated use. The representation offers two key benefits to designers, namely (i) it can directly be used to control integral simulations of use and to evaluate different courses of the use process, as was demonstrated in the example, and (ii) it describes anticipated usage in an unambiguous form, which makes application in other, similar design projects easier.

Among the further development activities towards resource-integrated modelling and simulation of use processes, the development of system components to enable nucleus-based object modelling and simulation has priority. Since the integration between a commercial rigid-body simulation package and Simulink/Stateflow turned out to be successful, the idea is to write a front-end application to simulate nucleus-based models as rigid bodies and obtain a first proof of ideas. A major challenge is that nucleus-based object models are built up from particles rather than rigid volumetric objects. However, in most commercial multibody simulation packages, objects can be connected with discrete flexible components. If the front-end application discretizes rigid volumetric objects into particle clouds in which the particles are connected by springs and dampers, it is expected that particle systems can be represented and simulated. The result would be a pilot system to simulate all mechanical behaviours in use processes, including large deformations, together with information processing and decision-making. One key application of simulating large deformations is to investigate physical interaction between humans and artefacts. As was demonstrated by the example in this paper, modelling and simulating interacting human body parts as rigid elements is often an inadequate workaround. A dedicated implementation of nucleus-based modelling that will also support observed physical behaviour outside the mechanical domain (thermodynamics, acoustics, etc.) is planned as a later development effort. Other future work involves the inclusion of generalized knowledge about human motion patterns, to achieve more realistic control of human body parts in simulations, and including discrete behaviour of programmed artefacts (e.g. electronic products) in use state machines.

## REFERENCES

- [1] Korn G.A. and Wait J.V. *Digital continuous system simulation*, 1978 (Prentice-Hall, Inc., Englewood Cliffs).
- [2] Eggert R.J. *Engineering Design*, 2005 (Pearson Prentice-Hall, Upper Saddle River).
- [3] Roozenburg N.F.M. and Eekels J. *Product design: fundamentals and methods*, 1995 (John Wiley & Sons, Chichester).
- [4] Pheasant S. and Haslegrave C.M. *Bodyspace - Anthropometry, Ergonomics and the Design of Work*, 1996 (Taylor & Francis Ltd., London).
- [5] Robertson I.T. Human information-processing strategies and style, *Behaviour and Information technology*, Vol. 4, 1985, No.1, pp. 19-29.
- [6] Newell A. and Simon H.A. Simulation of human thought. In *Computer simulation of human behavior*, 1971 (John Wiley and Sons, Inc. New York).
- [7] Eysenck M.W. and Keane M.T. *Cognitive Psychology*, 2000 (Psychology Press, Hove).
- [8] Stanton N.A. and Baber C. A systems analysis of consumer products, In *Human factors in consumer products*, 1998, pp. 75-90 (Taylor & Francis Ltd, London).
- [9] Hsia P. Samuel J. Gao J. Kung D. Toyoshima Y. and Chen C. Formal approach to scenario analysis, *IEEE Software*, March 1994, pp. 33-41.
- [10] Reason J. *Human error*, 1990 (Cambridge University Press, Cambridge).
- [11] Von Schwerin R. *Multibody system simulation - numerical methods, algorithms, and software*, 1999 (Berlin, Springer Verlag).
- [12] Zienkiewicz O.C and Taylor R.L. *The finite element method*, Vol. 1-3, 2000 (Butterworth-

- Heinemann, Oxford).
- [13] Van der Vegte W.F. A survey of artifact-simulation approaches from the perspective of application to use processes of consumer durables, In *Proceedings of the TMCE 2006*, Ljubljana, pp. 617-632.
  - [14] Chaffin D.B. *Digital human modeling for vehicle and workplace design*, 2001 (Society of Automotive Engineers, Inc., Warrendale).
  - [15] Van der Vegte W.F. and Horváth I. Including human behavior in product simulations for the investigation of use processes in conceptual design: a survey. In *Proceedings of ASME-IDETC 2006*, Philadelphia.
  - [16] Šuc D. and Bratko I. Qualitative trees applied to bicycle riding, *Linköping Electronic Transactions on Artificial Intelligence*, Vol. 4, 2000, Section B, pp. 125-140
  - [17] Reil T. and Husbands P. Evolution of central pattern generators for bipedal walking in a real-time physics environment. *IEEE Transactions on Evolutionary Computation*, Vol. 6, 2002, No. 2, pp. 159-168.
  - [18] Parnas D.L. On the use of transition diagrams in the design of a user interface for an interactive computer system. In *Proceedings of the ACM 24th National Conference*, 1969, pp. 379-385.
  - [19] Glinz M. Systematically combining specifications of internal and external system behavior using statecharts. In *Proceedings of the 3rd ICSE Workshop on Scenarios and State Machines: Models, Algorithms and Tools*, 2004.
  - [20] Rauterberg M. Fjeld M. and Schlupe S. Goal setting mechanism in Petri net models of human decision making. In *Proceedings of the IEEE Int. Conference on Systems, Man and Cybernetics*, Orlando, 1997, pp. 2696-2701.
  - [21] Liu A. and Salvucci D. Modeling and prediction of human driver behavior. In *Proceedings of the 9th Int. Conference on Human-Computer Interaction*, Los Angeles, 2001, pp. 1479-1483
  - [22] Martins J.D.F. Nedel L.P. Menezes P.F.B. and Accorsi F. An automata-based animation engine to control articulated characters. In *Proceedings of SBC Symposium on Virtual Reality*, São Paulo, 2004, pp. 39-50.
  - [23] Bobrow D.G. Qualitative reasoning about physical systems: an introduction, *Artificial Intelligence*, Vol. 24, 1984, pp.1-5.
  - [24] Hopcroft J.E. Motwani R. and Ullman J.D. *Introduction to automata theory, languages, and computation*, 2001 (Addison Wesley, Reading).
  - [25] Horváth, I. and Van der Vegte, W.F. Nucleus-based product conceptualization – Principles and formalization In *Proceedings of the 14th International Conference on Engineering Design*, 2003, Stockholm.
  - [26] Rusák Z. Horváth I. and Van der Vegte W.F. First steps towards an all embracing relations-based modelling. In *Proceedings of ASME-CIE 2004*, Salt Lake City
  - [27] Gill A. *Introduction to the theory of finite-state machines*, 1961 (McGraw-Hill book company, Inc., New York).
  - [28] Petri C.A. Fundamentals of a theory of asynchronous information flow. In *Proceedings of the 1962 IFIP Congress*, Amsterdam, pp. 386-390.
  - [29] Harel D. Statecharts: a visual formalism for complex systems, *Science of Computer Programming*, Vol. 8, 1987, pp. 231-274.
  - [30] Van der Vegte W.F. Hybrid simulation of the use of products by controlling continuous behaviour with state machines. In: *Proceedings of EDIProD' 2006*, Gronów, pp. 147-161.
  - [31] The MathWorks, Inc. *Stateflow and stateflow coder for use with Simulink - modeling, simulation, implementation*. 2004 (The MathWorks, Inc., Natick).
  - [32] David R. and Alla H. *Discrete, continuous and hybrid Petri nets*. 2005 (Springer Verlag, Berlin).

Contact: W.F. van der Vegte  
Delft University of Technology  
Faculty of Industrial Design Engineering  
Landbergstraat 15  
2628 CE Delft  
The Netherlands  
phone (+31)152781061  
e-mail w.f.vandervegte@tudelft.nl