Self-modeling Networks Using Adaptive Internal Mental Models for Cognitive Analysis and Support Processes

Jan Treur(✉)

Social AI Group, Vrije Universiteit Amsterdam, Amsterdam, Netherlands
j.treur@vu.nl

Abstract. In this paper, an adaptive network model is presented for cognitive analysis and support processes for the situation and states of a human, illustrated for a car driver. The adaptive network model makes use of first-order self-models for the internal mental models used for the cognitive analysis and support processes. To obtain adaptation of these first-order self-models, second-order self-models are included. The adaptive network model is illustrated for realistic scenarios for a car driver.

1 Introduction

Complex cognitive processes often make use of internal mental models for the processes addressed; e.g., [9, 10, 12, 15]. In the current paper, the focus is on cognitive processes involved in monitoring and assessing the situation of a human (in the addressed case study, a car driver), and generating support actions in order to keep the situation (the driving) safe. For example, when we observed that the driver took alcohol, we will assess that it will not be safe to start driving and therefore as a support action we will (try to) block starting the car. As another example, if while driving we observe that the driver's steering is unstable, we will assess this as unsafe driving and therefore as a support action we will propose to slow down the car (and perhaps completely stop the driving). Internal mental models involved in such processes are an analysis model to determine assessments based on the monitoring information and a support model to determine the proper support action based on the generated assessment. In principle, such internal mental models are adaptive so that they can improve over time.

Artificial variants of such cognitive processes implemented as AI, are more and more built in as automatic safety systems in new generations of cars. The advantage of this is that no passenger is needed to monitor the driver and, moreover, without further ado the car can more adequately execute support actions, for example, by refusing to start or refusing to drive fast if the driver condition is assessed as not safe.

In this paper, it is shown how the complex cognitive processes described above can be modeled using a network-oriented modeling approach. There are three crucial elements that play a role here. The first is that within the overall network model, internal mental models need to be modeled and executed, the second is that these mental models are adaptive so that they can improve over time, and the third is that control over

R. M. Benito et al. (Eds.): COMPLEX NETWORKS 2020, SCI 944, pp. 260–274, 2021.
https://doi.org/10.1007/978-3-030-65351-4_21

these processes is useful. All three elements can be addressed well by the approach based on self-modeling networks described in [20]. According to this approach through what is called network reification or self-modeling, any network can be extended to a self-modeling network by adding a self-model of part of its own network structure characteristics. Moreover, this can be done iteratively, so that multi-order self-models can be included in a self-modeling network, where any included self-model (of some order) can have its own (next-order) self-model.

In the paper, in Sect. 2 the modeling approach based on self-modeling networks is briefly described. In Sect. 3 the application domain is described in some detail. Section 4 presents the design of the introduced self-modeling network model and Sect. 5 presents outcomes of example simulations of it. Finally, Sect. 6 is a discussion.

2 Network Models Using Self-models

In this section, the network-oriented modeling approach used from [19, 20] is briefly introduced.

Distinction Between Network Characteristics and Network States
The following is a crucial distinction for network models:

- Network *characteristics* (such as connection weights and excitability thresholds) have values (their strengths) and determine (e.g., cognitive) processes and behaviour in an implicit, automatic manner. They can be considered to provide an *embodiment view* on the network. In principle, these characteristics by themselves may not be directly accessible nor observable for network states (or a person: usually you don't see or feel a specific connection in your brain).
- Network *states* (such as sensor states, sensory representation states, preparation states, emotion states) have values (their activation levels) and are explicit representations that may be accesible for network states or a person and can be handled or manipulated explicitly. They can be considered to provide an *informational view* on the network; usually the states are assumed to have a certain informational content. In principle, for the case of a mental network, states may be accessible or observable for a person: you may see (mental image), feel (emotion) or note in some other way a specific state in your brain.

Following [18, 20], a temporal-causal network model is characterised by (here X and Y denote nodes of the network, also called states):

- *Connectivity characteristics*
- Connections from a state X to a state Y and their weights $\boldsymbol{\omega}_{X,Y}$
- *Aggregation characteristics*
- For any state Y, some combination function $\mathbf{c}_{\boldsymbol{Y}}(..)$ defines the aggregation that is applied to the impacts $\boldsymbol{\omega}_{X,Y}X(t)$ on Y from its incoming connections from states X.
- *Timing characteristics*

Each state Y has a speed factor $\boldsymbol{\eta}_Y$ defining how fast it changes for given impact. The following difference (or differential) equations that are used for simulation purposes and also for analysis of temporal-causal networks incorporate these network characteristics $\boldsymbol{\omega}_{X,Y}$, $\mathbf{c}_Y(..)$, $\boldsymbol{\eta}_Y$ in a standard numerical format:

$$Y(t + \Delta t) = Y(t) + \boldsymbol{\eta}_Y[\mathbf{c}_Y(\boldsymbol{\omega}_{X_1,Y}X_1(t), \ldots \boldsymbol{\omega}_{X_k,Y}X_k(t)) - Y(t)]\Delta t \tag{1}$$

for any state Y and where X_1 to X_k are the states from which Y gets its incoming connections. Here the overall combination function $\mathbf{c}_Y(..)$ for state Y is the weighted average of available basic combination functions $\mathbf{c}_j(..)$ by specified weights $\boldsymbol{\gamma}_{j,Y}$ (and parameters $\boldsymbol{\pi}_{1,j,Y}$, $\boldsymbol{\pi}_{2,j,Y}$ of $\mathbf{c}_j(..)$) for Y:

$$\mathbf{c}_Y(V_1, \ldots, V_k) = \frac{\boldsymbol{\gamma}_{1,Y}\mathbf{c}_1(V_1, \ldots, V_k) + \ldots + \boldsymbol{\gamma}_{m,Y}\mathbf{c}_m(V_1, \ldots, V_k)}{\boldsymbol{\gamma}_{1,Y} + \ldots + \boldsymbol{\gamma}_{m,Y}} \tag{2}$$

Such Eqs. (1), (2) and the ones in Table 1 are hidden in the dedicated software environment; see [20], Ch 9. Within the software environment described there, a large number of around 40 useful basic combination functions are included in a combination function library; see Table 1 for the first two of them: these are the ones used in this paper. The above concepts enable to design network models and their dynamics in a declarative manner, based on mathematically defined functions and relations. How it works is that the network characteristics $\boldsymbol{\omega}_{X,Y}$, $\boldsymbol{\gamma}_{j,Y}$, $\boldsymbol{\pi}_{1,j,Y}$, $\boldsymbol{\pi}_{2,j,Y}$, $\boldsymbol{\eta}_Y$ that define the design of the network model, are given as input to the dedicated software environment, and hidden within this environment the difference Eqs. (1) are executed for all states, thus generating simulation graphs as output.

Table 1. Basic combination functions from the library used in the model presented here

	Notation	Formula	Parameters
Euclidean	$\mathbf{eucl}_{\boldsymbol{n},\boldsymbol{\lambda}}(V_1, \ldots, V_k)$	$\sqrt[n]{\frac{V_1^{\boldsymbol{n}}+\cdots+V_k^{\boldsymbol{n}}}{\lambda}}$	Order $\boldsymbol{n} > 0$ Scaling factor $\boldsymbol{\lambda} > 0$
Advanced logistic sum	$\mathbf{alogistic}_{\boldsymbol{\sigma},\boldsymbol{\tau}}(V_1, \ldots, V_k)$	$[\frac{1}{1+\mathrm{e}^{-\boldsymbol{\sigma}(V_1+\cdots+V_k-\boldsymbol{\tau})}} - \frac{1}{1+\mathrm{e}^{\boldsymbol{\sigma}\boldsymbol{\tau}}}](1 + \mathrm{e}^{-\boldsymbol{\sigma}\boldsymbol{\tau}})$	Steepness $\boldsymbol{\sigma} > 0$ Excitability threshold $\boldsymbol{\tau}$

Self-models Representing Network Characteristics by Network States

As indicated above, 'network characteristics' and 'network states' are two distinct concepts for a network. Self-modeling is a way to relate these distinct concepts to each other in an interesting and useful way:

- A *self-model* is making the implicit network characteristics (such as connection weights and excitability thresholds) explicit by adding states for these characteristics; thus the network gets an internal self-model of part of the network structure itself.

- In this way, different self-modeling levels can be created where network characteristics from one level relate to explicit states at a next level. By iteration, an arbitrary number of self-modeling levels can be modeled, covering second-order or higher-order effects.

Self-modeling causal networks can be recognized both in physical and mental domains. For example:

- In the *physical domain*, in the brain, information about the characteristics of the network of causal relations between activation states of neurons is, for example, represented in physical configurations for synapses (e.g., connection weights), neurons (e.g., excitability thresholds) and/or chemical substances (e.g., neurotransmitters).
- In the *mental domain*, a person can create mental states in the form of representations of his or her own (personal) characteristics, thus forming a subjective self-model (acquired by experiences); e.g., of being very sensitive for pain or for critical feedback or of having an anger issue.

Adding a self-model for a temporal-causal network is done in the way that for some of the states Y of the base network and some of the network structure characteristics for connectivity, aggregation and timing (in particular, some from $\boldsymbol{\omega}_{X,Y}$, $\boldsymbol{\gamma}_{i,Y}$, $\boldsymbol{\pi}_{i,j,Y}$, $\boldsymbol{\eta}_Y$), additional network states $\mathbf{W}_{X,Y}$, $\mathbf{C}_{i,Y}$, $\mathbf{P}_{i,j,Y}$, $\mathbf{H}_Y$ (self-model states) are introduced (see the blue upper plane in Fig. 2):

(a) **Connectivity self-model**

- Self-model states $\mathbf{W}_{X_i,Y}$ are added representing connectivity characteristics, in particular connection weights $\boldsymbol{\omega}_{X_i,Y}$

(b) **Aggregation self-model**

- Self-model states $\mathbf{C}_{j,Y}$ are added representing aggregation characteristics, in particular combination function weights $\boldsymbol{\gamma}_{i,Y}$
- Self-model states $\mathbf{P}_{i,j,Y}$are added representing aggregation characteristics, in particular combination function parameters $\boldsymbol{\pi}_{i,j,Y}$

(c) **Timing self-model**

- Self-model states $\mathbf{H}_Y$ are added representing timing characteristics, in particular speed factors $\boldsymbol{\eta}_Y$

The notations $\mathbf{W}_{X,Y}$, $\mathbf{C}_{i,Y}$, $\mathbf{P}_{i,j,Y}$, $\mathbf{H}_Y$ for the self-model states indicate the referencing relation with respect to the characteristics $\boldsymbol{\omega}_{X,Y}$, $\boldsymbol{\gamma}_{i,Y}$, $\boldsymbol{\pi}_{i,j,Y}$, $\boldsymbol{\eta}_Y$: here $\mathbf{W}$ refers to $\boldsymbol{\omega}$, $\mathbf{C}$ refers to $\boldsymbol{\gamma}$, $\mathbf{P}$ refers to $\boldsymbol{\pi}$, and $\mathbf{H}$ refers to $\boldsymbol{\eta}$, respectively. For the processing, these self-model states define the dynamics of state Y in a canonical manner according to Eqs. (1) whereby $\boldsymbol{\omega}_{X,Y}$, $\boldsymbol{\gamma}_{i,Y}$, $\boldsymbol{\pi}_{i,j,Y}$, $\boldsymbol{\eta}_Y$ are replaced by the state values of $\mathbf{W}_{X,Y}$, $\mathbf{C}_{i,Y}$, $\mathbf{P}_{i,j,Y}$, $\mathbf{H}_Y$ at time t, respectively.

An example of an aggregation self-model state $\mathbf{P}_{i,j,Y}$ for a combination function parameter $\boldsymbol{\pi}_{i,j,Y}$ is for the excitability threshold $\boldsymbol{\tau}_Y$ of state Y, which is the second parameter of the logistic sum combination function; then $\mathbf{P}_{i,j,Y}$ is usually indicated by $\mathbf{T}_Y$, where $\mathbf{T}$ refers to $\boldsymbol{\tau}$. Such aggregation self-model states $\mathbf{T}_Y$ will play an important role in the network model addressed below, as will connectivity self-model states $\mathbf{W}_{X,Y}$, referring to connection weights $\boldsymbol{\omega}_{X,Y}$. As the outcome of the addition of a self-model is also a temporal-causal network model itself, as has been proven in [21], Ch 10, this construction can easily be applied iteratively to obtain multiple levels of self-models.

3 Domain Description: Cognitive Analysis and Support Processes

In many cases, when humans perform complex or demanding tasks, it makes sense to keep an eye on them, to see how they are doing and to assess in how far their functioning is getting poor. If so, then some support actions may be needed or desirable. To determine such assessments and support actions requires complex and adaptive cognitive processes. For example, for a car driver, based on sensoring or observation data, it may involve judgements about the driver's alcohol usage, gaze and steering behaviour and whether for long trips (s)he takes enough rest. If the gaze is unfocused or the steering behaviour unstable, this may be assessed as a driving risk and if that occurs, a support action like slowing down the car may be adequate. The knowledge behind such assessments may be adaptive, so that the underlying cognitive processes can improve over time.

Within such complex and adaptive cognitive processes usually internal mental models are used. For example, in [3–6] internal mental models were used for the analysis process and for the support process (see also Fig. 1):

- *analysis model*
  This is used to assess the human's states and processes using observations (possibly using specific sensors) and domain knowledge. Examples of observations that are used in the car driver example are a long period of driving, a gaze that is not well-focused, unstable steering, and alcohol usage. Examples of assessments that come out of this process are that there is a risk for getting exhausted or there are other risks for driving.
- *support model*
  This is used to generate support for the human based on domain knowledge. This uses as input the assessments made by the analysis model. Examples of actions that come out of this process are advice to take some rest period, blocking the starting of the car (when it is not driving), and slowing down the car (when it is driving).

As such processes are in principle adaptive, a third internal mental model is needed [18], Ch. 16:

- *adaptation model*
  To make the analysis and support model better fit the specific characteristics of the driver, car and the further situation. This can be done by adapting certain characteristics of the internal mental models for analysis and support.

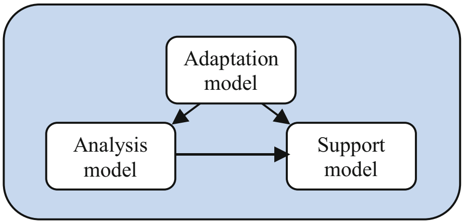

Fig. 1. Adaptive model-based architecture to analyse and support humans; adapted from [18], Ch 16, p. 469

Section 4 addresses the question how these internal mental models can be modeled by a self-modeling network, with as outcome a second-order self-modeling network model.

4 The Second-Order Adaptive Self-modeling Network Model

In this section it will be shown how the modeling approach briefly described in Sect. 2 can be and actually has been used to model within one self-modeling network the adaptive mental models for analysis and support sketched in Sect. 3.

A useful network architecture to handle internal mental models in general is a self-modeling network that covers at least two levels (see also [2]): a base level representing the mental model as a network so that it can be used to process it (based on within-network dynamics), and a first-order self-model explicitly representing the (network) characteristics of the mental model which can be used for formation and adaptation of the mental model. In addition, a third level with a second-order self-model can be used to control these processes. This general setup has been applied here.

First two useful adaptation principles for plasticity and metaplasticity from the Cognitive Neuroscience literature are discussed. When self-models are changing over time, this offers a useful method to model adaptive networks. This does not only apply to first-order adaptive networks, but also to higher-order adaptive networks, using higher-order self-models. For example, two types of (connectivity and aggregation) self-model states can be used to model adaptive connection weights and intrinsic neuronal excitability as described in [7]:

> 'Learning-related cellular changes can be divided into two general groups: modifications that occur at synapses and modifications in the intrinsic properties of the neurons. While it is commonly agreed that changes in strength of connections between neurons in the relevant networks underlie memory storage, ample evidence suggests that modifications in intrinsic neuronal properties may also account for learning related behavioral changes'. [7], p. 30.

More in particular, the following quote indicates that synaptic activity relates to long-lasting modifications in excitability of neurons:

> 'Long – lasting modifications in intrinsic excitability are manifested in changes

in the neuron's response to a given extrinsic current (generated by synaptic activity or applied via the recording electrode'.[7], p.30 (3)

The above refers to a form of *plasticity*, which can be described by a first-order adaptive network that is modelled using a dynamic first-order self-model for aggregation characteristics of the base network, in particular for the excitability threshold used in aggregation. Whether or not and to which extent such plasticity actually takes place is controlled by a form of *metaplasticity*; e.g., [1, 8, 13, 14, 16, 17]. For example, in [14] the following compact quote is found, summarizing that due to stimulus exposure, adaptation speed will increase:

'Adaptation accelerates with increasing stimulus exposure' [14], p.2 (4)

This indeed refers to a form of metaplasticity, which can be described by a second-order adaptive network that is modeled using a dynamic second-order self-model for timing characteristics of the first-order self-model for the first-order adaptation. In this way, both (first- and second-order) adaptation principles for plasticity and metaplasticity summarized in (3) and (4) will be applied in the network model presented below.

Because of its complexity, the model will be presented in two steps as depicted in Fig. 2 and Fig. 3. In Fig. 2 the connectivity of the first two levels of the proposed network model is depicted. This covers the base network within the base (pink) plane, and the first-order self-model in the upper (blue) plane. For an overview of all states of the network model, see Table 2; here the first 10 states describe the base level and the next 15 states (up to state 25) the network's first-order self-model.

The base network consists of two subnetworks, one that describes a mental model for the analysis to determine (by within-network dynamics) out of monitored information about the driver's situation (long drive, alcohol, unstable steering, unfocused gaze), an assessment of the situation of the driver (within the considered scenarios the two options are exhaustiveness risk and driving risk). The second one describes a mental model for the support process to determine (by within-network dynamics) out of the assessment a suitable support action for the driver (in the considered scenarios three options: rest advice, slow down, and block start).

For these mental models described at the base level, corresponding self-models have been added to be able to change them, for example by learning. The first-order self-model in the upper plane in Fig. 2 models some of the network characteristics of the two (sub)networks at the base level:

Analysis Self-Model: First-order self-model **W**-states and **T**-states X_{11} to X_{16}.
Support Self-Model: First-order self-model **W**-states and **T**-states X_{17} to X_{25}.

For each of the subnetworks for mental models at the base level, the first-order self-model has two submodels: a first-order *connectivity self-model* (based on **W**-states) and a first-order *aggregation self-model* (based on **T**-states). The connectivity self-model represents the connectivity characteristics of the particular mental model by self-model states $\mathbf{W}_{X,Y}$ and the aggregation self-model represents the excitability thresholds of the assessment options (for the analysis model) and the support action options (for the support model) by self-model states $\mathbf{T}_Y$. Each of these first-order self-model states $\mathbf{W}_{X,Y}$ and $\mathbf{T}_Y$ has a downward connection (in pink) to indicate the state Y of the mental model at

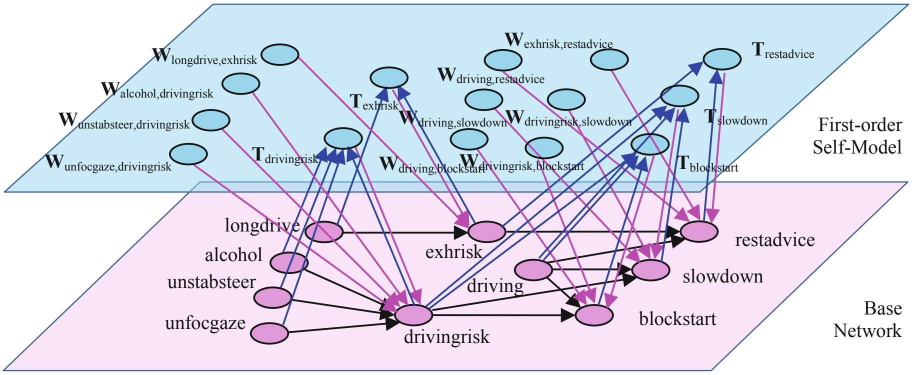

Fig. 2. Connectivity of the first-order self-modeling network model

the base level for which they have their special effect; so, based on these downward links, the value of $\mathbf{W}_{X,Y}$ plays the role of the indicated connection weight and the value of $\mathbf{T}_Y$ the role of excitability threshold for the state Y pointed at. For the sake of simplicity, the connectivity self-model states $\mathbf{W}_{X,Y}$ have no incoming connections from other states; for the scenarios considered here they are kept constant, in further extensions of the model they easily may made dynamic as well, for example, based on hebbian learning as modeled in [21], Ch. 3.

The aggregation self-model states for excitability thresholds do have incoming connections which make them dynamic, to model the abovementioned adaptation principle (3) for plasticity from [7]. This is modeled by specifying a negative weight of the connections from the states that causally preceed the indicated state Y.

In addition, to counterbalance an excess of this negative effect, a positive weight 1 is used for all upward connections from Y itself to $\mathbf{T}_Y$. For the simulation outcomes discussed in next section it is shown how these two opposite effects create some equilibrium value for each aggregation self-model state $\mathbf{T}_Y$, which illustrates one form of adaptivity in the model: in this way the aggregation self-model learns. However, by including a second-order self-model as shown (in the purple plane) in Fig. 3, this learning has been made adaptive itself (in particular the learning speed), which creates second-order adaptation used as a form of control over the first-order adaptation.

In Fig. 3, the connectivity of the entire second-order adaptive network model is depicted. Compared to Fig. 2 a third (upper, purple) plane was added consisting of a second-order self-model for the network. The second-order self-model states in this upper (purple) plane are explained in Table 2: the last 19 states from state 26 on:

Adaptation Self-Model: Second-order self-model **W**-states and $\mathbf{H_T}$-states X_{26} to X_{44}.

As in the first-order self-model, the second-order self-model includes a second-order *connectivity self-model* using states $\mathbf{W}_{X,\mathbf{T}_Y}$ for all incoming connections of the **T**-states of the first-order self-model. Again, like in the first-order self-model case, these states $\mathbf{W}_{X,\mathbf{T}_Y}$ are kept constant for now. In addition, the second-order self-model includes a second-order *timing self-model* for the first-order **T**-states based on states $\mathbf{H}_{\mathbf{T}_Y}$. This

Table 2. Explanation of the states of the second-order self-modeling network model

Name		Explanation
X_1	longdrive	The driver is driving for a long period of time
X_2	alcohol	Alcohol is detected
X_3	unstabsteer	The driver's steering is unstable
X_4	unfocgaze	The driver's gaze is not focused
X_5	exhrisk	Assessment of a risk that the driver will get exhausted
X_6	drivingrisk	Assessment of a safety risk for the driving
X_7	driving	The car is driving
X_8	restadvice	Supporting action to advice the driver to take some rest
X_9	slowdown	Supporting action to slow down the car
X_{10}	blockstart	Supporting action to block the starting of the car
X_{11}	$\mathbf{W}_{\text{longdrive,exhrisk}}$	First-order connectivity self-model state for weight of the connection from longdrive to exhrisk
X_{12}	$\mathbf{W}_{\text{alcohol,drivingrisk}}$	First-order connectivity self-model state for weight of the connection from alcohol to drivingrisk
X_{13}	$\mathbf{W}_{\text{unstabsteer,drivingrisk}}$	First-order connectivity self-model state for weight of the connection from unstablesteer to drivingrisk
X_{14}	$\mathbf{W}_{\text{unfocgaze,drivingrisk}}$	First-order connectivity self-model state for weight of the connection from longdrive to drivingrisk
X_{15}	$\mathbf{T}_{\text{exhrisk}}$	First-order aggregation self-model state for excitability threshold of exhrisk
X_{16}	$\mathbf{T}_{\text{drivingrisk}}$	First-order aggregation self-model state for excitability threshold of drivingrisk
X_{17}	$\mathbf{W}_{\text{exhrisk,restadvice}}$	First-order connectivity self-model state for weight of the connection from exhrisk to restadvice
X_{18}	$\mathbf{W}_{\text{driving,restadvice}}$	First-order connectivity self-model state for weight of the connection from driving to restadvice
X_{19}	$\mathbf{W}_{\text{drivingrisk,slowdown}}$	First-order connectivity self-model state for weight of the connection from drivingrisk to slowdown
X_{20}	$\mathbf{W}_{\text{driving,slowdown}}$	First-order connectivity self-model state for weight of the connection from driving to slowdown
X_{21}	$\mathbf{W}_{\text{drivingrisk,blockstart}}$	First-order connectivity self-model state for weight of the connection from drivingrisk to blockstart
X_{22}	$\mathbf{W}_{\text{driving,blockstart}}$	First-order connectivity self-model state for weight of the connection from driving to blockstart
X_{23}	$\mathbf{T}_{\text{restadvice}}$	First-order aggregation self-model state for excitability threshold of restadvice
X_{24}	$\mathbf{T}_{\text{slowdown}}$	First-order aggregation self-model state for excitability threshold of slowdown
X_{25}	$\mathbf{T}_{\text{blockstart}}$	First-order aggregation self-model state for excitability threshold of blockstart
X_{26}	$\mathbf{W}_{\text{longdrive},\mathbf{T}_{\text{exhrisk}}}$	Second-order connectivity self-model state for weight of the connection from longdrive to $\mathbf{T}_{\text{exhrisk}}$
X_{27}	$\mathbf{W}_{\text{exhrisk},\mathbf{T}_{\text{exhrisk}}}$	Second-order connectivity self-model state for weight of the connection from exhrisk to $\mathbf{T}_{\text{exhrisk}}$
X_{28}	$\mathbf{W}_{\text{alcohol},\mathbf{T}_{\text{drivingrisk}}}$	Second-order connectivity self-model state for weight of the connection from alcohol to $\mathbf{T}_{\text{drivingrisk}}$
X_{29}	$\mathbf{W}_{\text{unstabsteer},\mathbf{T}_{\text{drivingrisk}}}$	Second-order connectivity self-model state for weight of the connection from unstabsteer to $\mathbf{T}_{\text{drivingrisk}}$
X_{30}	$\mathbf{W}_{\text{unfocgaze},\mathbf{T}_{\text{drivingrisk}}}$	Second-order connectivity self-model state for weight of the connection from unfocgaze to $\mathbf{T}_{\text{drivingrisk}}$
X_{31}	$\mathbf{W}_{\text{drivingrisk},\mathbf{T}_{\text{drivingrisk}}}$	Second-order connectivity self-model state for weight of the connection from drivingrisk to $\mathbf{T}_{\text{drivingrisk}}$
X_{32}	$\mathbf{W}_{\text{exhrisk},\mathbf{T}_{\text{restadvice}}}$	Second-order connectivity self-model state for weight of the connection from exhrisk to $\mathbf{T}_{\text{restadvice}}$
X_{33}	$\mathbf{W}_{\text{restadvice},\mathbf{T}_{\text{restadvice}}}$	Second-order connectivity self-model state for weight of the connection from restadvice to $\mathbf{T}_{\text{restadvice}}$
X_{34}	$\mathbf{W}_{\text{drivingrisk},\mathbf{T}_{\text{slowdown}}}$	Second-order connectivity self-model state for weight of the connection from drivingrisk to $\mathbf{T}_{\text{slowdown}}$
X_{35}	$\mathbf{W}_{\text{driving},\mathbf{T}_{\text{slowdown}}}$	Second-order connectivity self-model state for weight of the connection from driving to $\mathbf{T}_{\text{slowdown}}$
X_{36}	$\mathbf{W}_{\text{slowdown},\mathbf{T}_{\text{slowdown}}}$	Second-order connectivity self-model state for weight of the connection from slowdown to $\mathbf{T}_{\text{slowdown}}$
X_{37}	$\mathbf{W}_{\text{drivingrisk},\mathbf{T}_{\text{blockstart}}}$	Second-order connectivity self-model state for weight of the connection from drivingrisk to $\mathbf{T}_{\text{blockstart}}$
X_{38}	$\mathbf{W}_{\text{driving},\mathbf{T}_{\text{blockstart}}}$	Second-order connectivity self-model state for weight of the connection from driving to $\mathbf{T}_{\text{blockstart}}$
X_{39}	$\mathbf{W}_{\text{blockstart},\mathbf{T}_{\text{blockstart}}}$	Second-order connectivity self-model state for weight of the connection from blockstart to $\mathbf{T}_{\text{blockstart}}$
X_{40}	$\mathbf{H}_{\mathbf{T}_{\text{exhrisk}}}$	Second-order timing self-model state for the speed of $\mathbf{T}_{\text{exhrisk}}$
X_{41}	$\mathbf{H}_{\mathbf{T}_{\text{drivingrisk}}}$	Second-order timing self-model state for the speed of $\mathbf{T}_{\text{drivingrisk}}$
X_{42}	$\mathbf{H}_{\mathbf{T}_{\text{restadvice}}}$	Second-order timing self-model state for the speed of $\mathbf{T}_{\text{restadvice}}$
X_{43}	$\mathbf{H}_{\mathbf{T}_{\text{slowdown}}}$	Second-order timing self-model state for the speed of $\mathbf{T}_{\text{slowdown}}$
X_{44}	$\mathbf{H}_{\mathbf{T}_{\text{blockstart}}}$	Second-order timing self-model state for the speed of $\mathbf{T}_{\text{blockstart}}$

second-order self-model is dynamic, which makes the whole network second-order adaptive. The special effect of each state $\mathbf{H}_{\mathbf{T}_Y}$ as speed factor for state $\mathbf{T}_Y$ is effectuated by the downward (pink) connection to the related state $\mathbf{T}_Y$.

To make them dynamic, the states $\mathbf{H}_{\mathbf{T}_Y}$ themselves are affected by upward connections from the base level network, in this case following the abovementioned adaptation principle (4) for metaplasticity 'Adaptation accelerates with increasing stimulus exposure' [14]. Therefore, there are (blue) upward links with positive weights to each state $\mathbf{H}_{\mathbf{T}_Y}$ from the base states causally preceeding base state Y. This makes that, as soon as

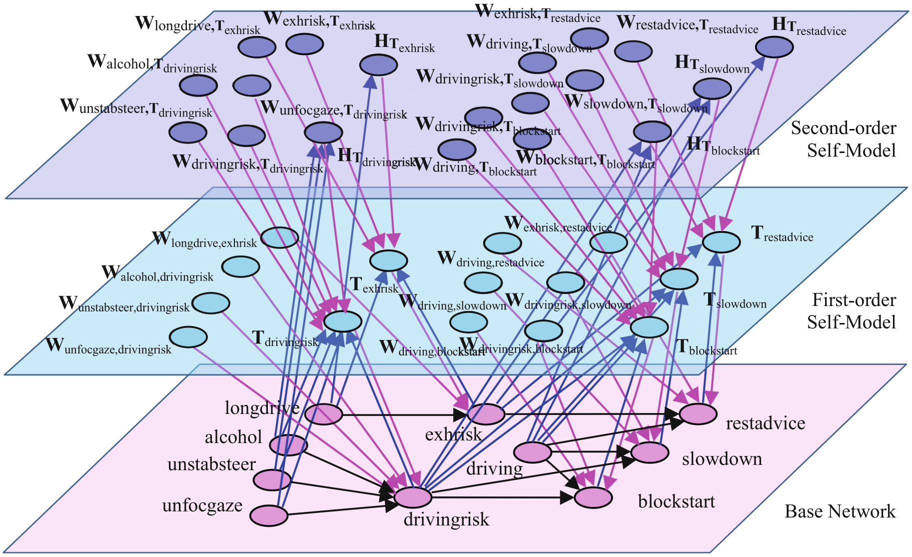

Fig. 3. Connectivity of the entire second-order self-modeling network model

these causal 'antecedents' of Y get higher activation levels, the adaptation speed (starting from 0: no adaptation initially) will increase, as will be shown in the example scenarios.

For full specifications of the adaptive network model, see the Appendix at URL https://www.researchgate.net/publication/344165044. For example, for connectivity characteristics all connection weights not determined bij $\mathbf{W}$-states are 1, except for the connection from driving to $\mathbf{H}_{\mathbf{T}_{\text{blockstart}}}$, which is -1. For aggregation characteristics, the logistic sum combination function is used for the base states for assessment and support options (with steepness $\boldsymbol{\sigma} = 8$ and adaptive excitability threshold) and the second-order $\mathbf{H}_{\mathbf{T}}$-states (with steepness $\boldsymbol{\sigma} = 4$ and excitability threshold $\boldsymbol{\tau} = 0.7$ or 1.4 depending on the number of incoming connections). All other states use the Euclidean combination function with $\boldsymbol{n} = 1$ and $\boldsymbol{\lambda} = 1$ (see Table 1), which actually is just a sum function. For timing characteristics, the speed factors of the base states for assessment and support options are 0.5 and for the second-order $\mathbf{H}_{\mathbf{T}}$-states 0.05. All other speed factors are adaptive (the base states for assessment and support options) or 0 (for the other base states and for all $\mathbf{W}$-states). The initial values for all $\mathbf{W}$-states (which are constant due to speed factor 0) are 1 when they represent a positive connection; negative ones are $\mathbf{W}_{\text{driving, blockstart}}$, $\mathbf{W}_{\text{longdrive}}$, $\mathbf{T}_{\text{exhrisk}}$, $\mathbf{W}_{\text{alcohol}}$, $\mathbf{T}_{\text{drivingrisk}}$, $\mathbf{W}_{\text{unstabsteer}}$, $\mathbf{T}_{\text{drivingrisk}}$, $\mathbf{W}_{\text{unfocgaze}}$, $\mathbf{T}_{\text{drivingrisk}}$, $\mathbf{W}_{\text{exhrisk}}$, $\mathbf{T}_{\text{restadvice}}$ which have initial value -1, and $\mathbf{W}_{\text{drivingrisk}}$, $\mathbf{T}_{\text{slowdown}}$, $\mathbf{W}_{\text{driving}}$, $\mathbf{T}_{\text{slowdown}}$, $\mathbf{W}_{\text{drivingrisk}}$, $\mathbf{T}_{\text{blockstart}}$, $\mathbf{W}_{\text{driving}}$, $\mathbf{T}_{\text{blockstart}}$ with initial value -0.5. The initial values of all $\mathbf{H}_{\mathbf{T}}$-states are 0 as are they for all base states except the observables shown in Table 3, which depend on the chosen scenario. Finally, the initial values for the five $\mathbf{T}$-states were on purpose set on too high values 2, 1.4, 2.4, 2.8, 2.4, respectively (in relation to the number of their incoming connections), in order to let adaptation happen.

5 Outcomes of Example Simulation Scenarios

In Figs. 4, 5 and 6 simulation results are shown for three realistic scenarios, defined by the common settings as shown in the role matrices in the Appendix discussed in the last paragraph of Sect. 4 and specific constant values 0 or 1 for the states X_1 to X_4 and X_7 as shown in Table 3. In these graphs the following are shown:

- the relevant assessment (resulting from the analysis model) and support action (resulting from the support model)
- how the excitability thresholds used within the analysis model and the support model adapt over time and how the adaptation speed for them changes over time (resulting from the adaptation model)

The initial values for the excitabilty thresholds for the analysis model and support model were deliberately set too high, so that the adaptation process that was needed to get results is illustrated. Note that the adaptation speeds have initial values 0 so that in the first phase nothing happens in the analysis model and support model until indeed a suitable adaptation process has started and in a next phase has resulted in successful adaptation of the analysis and support models.

Table 3. The three displayed scenarios

	X_1 longdrive	X_2 alcohol	X_3 unstabsteer	X_4 unfocgaze	X_7 driving	Explanation
Scenario 1 (Fig. 4)	1	0	0	0	1	A driver who has been driving too long
Scenario 2 (Fig. 5)	0	0	0	1	1	A driver who drives with an unfocused gaze
Scenario 3 (Fig. 6)	0	1	0	0	0	A driver who consumed alcohol and wants to start driving

For Scenario 1, it can be seen in Fig. 4 that by the second-order self-model the adaptation speed for the exhaustion risk excitability threshold (within the analysis model) increases from time 0 on (the purple line); this is conform to the 'Plasticity Versus Stability Conundrum' discussed in [17], p. 773: only adapt when relevant (adaptation speed > 0), otherwise keep stable (adaptation speed 0). This increase in adaptation speed (due to stimulus exposure) results in adaptation of this excitability threshold (conform to (3) from [7]): starting at value 2, it goes down to finally (after time 13) reach values between 0.2 and 0.4 (the brown line). Apparently this is low enough, as after time 10 the exhaustion risk assessment is generated and reaches value 1 after time 15 (the red line), which makes a successful analysis model outcome for this scenario.

This in turn makes that by the adaptation model after time 10 the adaptation speed for the excitability threshold of the support action rest advice (in the support model) gets

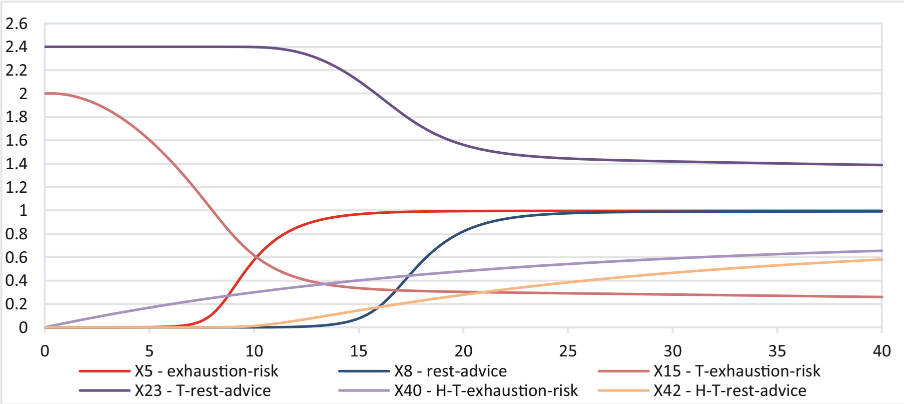

Fig. 4. Long drive leads to an exhaustion risk assessment and to the support action rest advice

higher (the orange line). This results in adaptation of that threshold: the value which initially was 2.4 starts to decrease after time 10 and reaches values between 1.4 and 1.6 after time 18 (the dark purple line). Again, apparently this is low enough as the support action rest advice comes up after time 18 and reaches 1 after time 25 (the dark green line). This makes a successful support outcome.

For Scenario 2, it can be seen in Fig. 5 that by the adaptation model the adaptation speed for the driving risk excitability threshold (within the analysis model) increases from time 0 on (the light blue line), which results in adaptation of this threshold: starting at value 1.4, it goes down to (after time 7) reach values below 0.7 (the light green line).

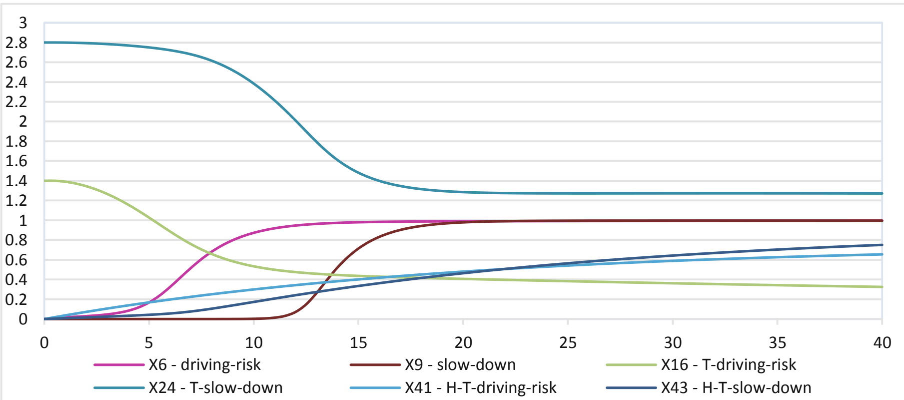

Fig. 5. Driving with an unfocused gaze leads to a driving risk assessment and to the support action slow down

Apparently this is low enough, as from time 5–10 the driving risk assessment is generated and reaches value 1 after time 15 (the pink line), which makes a successful analysis model outcome for this case. This in turn makes that by the adaptation model after time 5 the adaptation speed for the excitability threshold of the support action slow down (in the support model) gets higher (the dark green line). This results in adaptation of that threshold: the value which initially was 2.8 starts to decrease after time 10 and reaches values between 1.4 and 1.6 after time 18 (middle green line). Again, apparently this is low enough as the support action slow down comes up after time 18 and reaches 1 after time 20 (the brown line). This makes a successful support model outcome for this case.

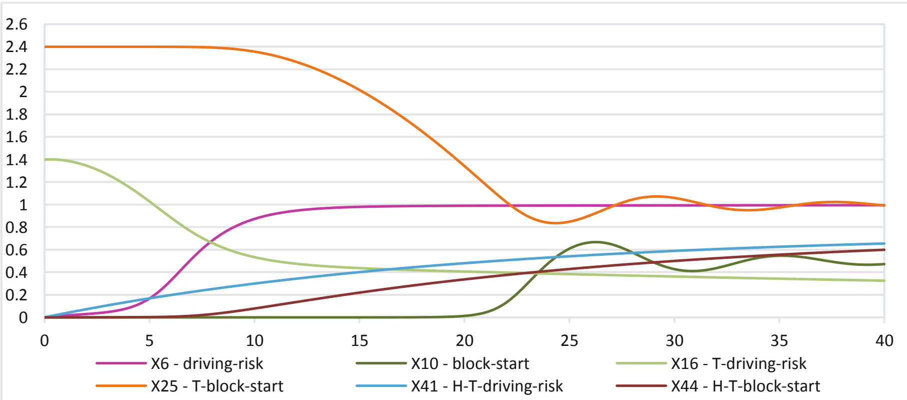

Fig. 6. Alcohol usage leads to a driving risk assessment and to the support action block start

For Scenario 3, Fig. 6 shows initially (for the analysis model) a similar pattern as in Scenario 2. However, for the second part of the process (for the support model), this scenario shows how also fluctuating patterns can occur. More specifically, this illustrates how the adaptation of the excitability threshold threshold gets reinforcement from the outcome of the support model, so that in the end they reach an equilibrium according to a fluctuating pattern.

6 Discussion

In complex cognitive processes, often internal mental models are used; e.g., [9, 10, 12, 15]. Such models can just be applied, but they are also often adaptive, in order to form and improve them. The focus in this paper was on adaptive cognitive analysis and support processes for the situation and states of a human in a demanding task; the adaptive network model was illustrated for a car driver. Within these processes internal mental models are used for the analysis and support processes.

An adaptive network model was presented that models such adaptive cognitive analysis and support processes. The network model makes use of adaptive first-order self-models for the internal mental models used for the cognitive analysis and support processes. To control the adaptation of these first-order self-models, second-order self-models are included. The adaptive network model was illustrated for realistic scenarios for a car driver who gets exhausted, shows unstable steering or shows an unfocused gaze and/or used alcohol.

For the adaptativity and its control, the network model makes use of two biologically plausible adaptation principles informed by the Cognitive Neuroscience literature, one within the first-order self-model for adaptation of aggregation characteristics of the base network, in particular the excitability threshold [7], and the other one [14] within the second-order self-model for adaptation of timing characteristics for the first-order self-model by metaplasticity [1, 8, 13, 14, 16, 17]. This study shows how complex adaptive cognitive processes based on internal mental models can be modeled in an adequate manner by multi-order self-modeling networks.

References

1. Abraham, W.C., Bear, M.F.: Metaplasticity: the plasticity of synaptic plasticity. Trends Neurosci. **19**(4), 126–130 (1996)
2. Bhalwankar, R., Treur, J.: Modeling the development of internal mental models by an adaptive network model. In: Proceedings of the 11th Annual International Conference on Brain-Inspired Cognitive Architectures for AI, BICA*AI'20. Advances in Intelligent Systems and Computing, Springer Nature Publishers (2020)
3. Bosse, T., Both, F., Duell, R., Hoogendoorn, M., van Lambalgen, R., Klein, M.C.A., van der Mee, A., Oorburg, R., Sharpanskykh, A., Treur, J., de Vos, M.: An ambient agent system assisting humans in complex tasks by analysis of a human's state and performance. Int. J. Intell. Inf. Database Syst. **7**, 3–3 (2013)
4. Bosse, T., Both, F., Gerritsen, C., Hoogendoorn, M., Treur, J.: Methods for model-based reasoning within agent-based ambient intelligence applications. Knowl.-Based Syst. J. **27**, 190–210 (2012)
5. Bosse, T., Hoogendoorn, M., Klein, M.C.A., Lambalgen, R.M. van, Maanen, P.P. van, Treur, J.: Incorporating human aspects in ambient intelligence and smart environments. In: Chong, N.Y., Mastrogiovanni, F. (eds.) Handbook of Research on Ambient Intelligence and Smart Environments: Trends and Perspectives, pp. 128–164. IGI Global (2011)
6. Bosse, T., Hoogendoorn, M., Klein, M.C.A., Treur, J.: An ambient agent model for monitoring and analysing dynamics of complex human behaviour. J. Ambient Intell. Smart Environ. **3**, 283–303 (2011)
7. Chandra, N., Barkai, E.: A non-synaptic mechanism of complex learning: modulation of intrinsic neuronal excitability. Neurobiol. Learn. Mem. **154**, 30–36 (2018)
8. Garcia, R.: Stress, Metaplasticity, and antidepressants. Curr. Mol. Med. **2**, 629–638 (2002)
9. Gentner, D., Stevens, A.L.: Mental models. Erlbaum, Hillsdale NJ (1983)
10. Greca, I.M., Moreira, M.A.: Mental models, conceptual models, and modelling. Int. J. Sci. Educ. **22**(1), 1–11 (2000)
11. Hebb, D.O.: The Organization of Behavior: A Neuropsychological Theory. Wiley, New York (1949)
12. Kieras, D.E., Bovair, S.: The role of a mental model in learning to operate a device. Cognit. Sci. **8**(3), 255–273 (1984)
13. Magerl, W., Hansen, N., Treede, R.D., Klein, T.: The human pain system exhibits higher-order plasticity (metaplasticity). Neurobiol. Learn. Mem. **154**, 112–120 (2018)
14. Robinson, B.L., Harper, N.S., McAlpine, D.: Meta-adaptation in the auditory midbrain under cortical influence. Nat. Commun. **7**, 13442 (2016)
15. Seel, N.M.: Mental Models in Learning Situations. In: Advances in Psychology, vol. 138, pp. 85–107. North-Holland, Amsterdam (2006)

16. Sehgal, M., Song, C., Ehlers, V.L., Moyer, J.R., Jr.: Learning to learn – intrinsic plasticity as a metaplasticity mechanism for memory formation. Neurobiol. Learn. Mem. **105**, 186–199 (2013)
17. Sjöström, P.J., Rancz, E.A., Roth, A., Hausser, M.: Dendritic excitability and synaptic plasticity. Physiol. Rev. **88**, 769–840 (2008)
18. Treur, J.: Network-Oriented Modeling: Addressing Complexity of Cognitive, Affective and Social Interactions. Springer (2016)
19. Treur, J.: Modeling higher-order adaptivity of a network by multilevel network reification. Network Sci. **8**, S110–S144 (2020)
20. Treur, J.: Network-oriented modeling for adaptive networks: Designing Higher-order Adaptive Biological, Mental and Social Network Models. Springer Nature Publishing, Cham (2020)