Formal Modelling of Emotions in BDI Agents

David Pereira, Eugénio Oliveira and Nelma Moreira

Technical Report Series: DCC-2007-04

Departamento de Ciência de Computadores
&
Laboratório de Inteligência Artificial e Ciência de Computadores

Faculdade de Ciências da Universidade do Porto
Rua do Campo Alegre, 1021/1055,
4169-007 PORTO,
PORTUGAL

Tel: 220 402 900 Fax: 220 402 950
http://www.dcc.fc.up.pt/Pubs/

Abstract

Emotional-BDI agents are BDI agents whose behaviour is guided not only by beliefs, desires and intentions, but also by the role of emotions in reasoning and decision-making. The $\mathcal{E}_{\mathsf{BDI}}$ logic is a formal system for expressing the concepts of the Emotional-BDI model of agency. In this paper we present an improved version of the $\mathcal{E}_{\mathsf{BDI}}$ logic and show how it can be used to model the role of three emotions in Emotional-BDI agents: *fear*, *anxiety* and *self-confidence*. We also focus in the computational properties of $\mathcal{E}_{\mathsf{BDI}}$ which can lead to its use in automated proof systems.

1.1 Introduction

Emotional-BDI agents are BDI agents whose behaviour is guided not only by beliefs, desires and intentions, but also by the role of emotions in reasoning and decision-making. This conceptual model was developed by Pereira *et al.* [POMS05] and a first version of the $\mathcal{E}_{\mathsf{BDI}}$ logic was presented in [POM06], where a first formalisation of *fear* was given. In this paper we present an improved version of the $\mathcal{E}_{\mathsf{BDI}}$ logic in order to model the role of three emotions in Emotional-BDI agents: *fear*, *anxiety* and *self-confidence*. The aim of this paper is to show how $\mathcal{E}_{\mathsf{BDI}}$ logic has enough expressivity to model some of the properties of these emotions, following Oliveira & Sarmento's model of emotional agent [OS02, OS03, SMO04].

The main motivation for the current work was to provide a formal system in which the concepts of the Emotional-BDI model of agency could be logically expressed. Using these concepts we can specify distinct behaviours which are expected from agents under the influence of emotions. The existing formal systems for rational agency such as Rao & Georgeff's BDI logics [RG98, RG91] and Meyer's *et al.* **KARO** framework [vdHvLM94, STH04, vLvdHM98, vdHvLM00] do not allow a straight forward representation of emotions. However, both have properties which we can combine in order to properly model Emotional-BDI agents.

The $\mathcal{E}_{\mathsf{BDI}}$ logic is an extension of the $\mathsf{BDI}_{\mathsf{CTL}}$ logic, equipped with explicit reference to actions, capabilities and resources. The choice of $\mathsf{BDI}_{\mathsf{CTL}}$, and not the more powerfull $\mathsf{BDI}_{\mathsf{CTL}^*}$, was motivated by our interest in automated proof methods that will allow the development of executable specification languages of rational agency or of formal verification systems for the Emotional-BDI model of agency.

This paper is organised as follows. In Section 1.2 we define the $\mathcal{E}_{\mathsf{BDI}}$ logic. This logic is based in $\mathsf{BDI}_{\mathsf{CTL}}$ logic and we begin by referring the new operators that were added. Besides the syntax and semantics of $\mathcal{E}_{\mathsf{BDI}}$, we present the axiom systems for the new modal operators. We also establish the decidability of $\mathcal{E}_{\mathsf{BDI}}$-formulae, by transforming $\mathcal{E}_{\mathsf{BDI}}$-formulae into equivalent $\mathsf{BDI}_{\mathsf{CTL}}$ ones. In Section 1.3 we use the $\mathcal{E}_{\mathsf{BDI}}$-logic to define a set of conditions which are pre-requisites for defining how emotions are activated in Emotional-BDI agents and also special purpose actions which are executed by the agent when it "feels" these emotions. In Section 1.4 we model the activation and effects of each of the emotions in Emotional-BDI agents using the previous conditions. Finally, in Section 1.5 we present some conclusions about this work and point some topics for ongoing and future research in the $\mathcal{E}_{\mathsf{BDI}}$ logic.

1.2 The $\mathcal{E}_{\mathsf{BDI}}$ logic

The $\mathcal{E}_{\mathsf{BDI}}$ is an extension of Rao & Georgeff's $\mathsf{BDI}_{\mathsf{CTL}}$. This extension adds new modal operators for representing the concepts of fundamental desires, capabilities, action execution and resources. The semantics of $\mathcal{E}_{\mathsf{BDI}}$ is therefore given by the satisfiability of $\mathcal{E}_{\mathsf{BDI}}$-formulae on extended $\mathsf{BDI}_{\mathsf{CTL}}$-models, considering accessibility-relations and functions for modelling the new operators.

1.2.1 Informal description

The $\mathsf{BDI}_{\mathsf{CTL}}$ logic is a multi-modal logic which combines Emerson's *et al.* branching-time logic CTL [Eme90] and modal operators for representing the mental states of belief (Bel), desire (Des) and intention (Int) as defined by Bratman *et al.* in [BIP88]. The underlying model of $\mathsf{BDI}_{\mathsf{CTL}}$ has a two dimensional structure. One dimension is a set of possible worlds that correspond to the different perspectives of the agent representing his mental states. The other is a set of temporal states which describe the temporal evolution of the agent. A pair $\langle world, temporal\_state\rangle$ is called a *situation*.

In the $\mathcal{E}_{\mathsf{BDI}}$ logic we added the following modal operators:

Fundamental desire: a fundamental desire is a desire which represents vital conditions to the agent, like its life or alike propositions. We model this concept using the modal operator Fund.

Actions: in $\mathcal{E}_{\mathsf{BDI}}$ we consider regular actions as defined in Propositional Dynamic Logic PDL [HKT00]. In this way we can refer to the actions that the agent performs, in particular when he is under the influence of emotions. Given a finite set of atomic actions, regular actions are derived through the usual regular action operations (test, sequence, disjunction and Kleene closure).

Capabilities: a capability represents the operational structure of the execution of an action. This concept is similar to **KARO**'s *ability*. This is represented by the modal operator Cap.

Resources: resources are the means (physical or virtual) for engaging the execution of actions. For the modelling of resources we consider the operators:

- $\mathsf{Needs}(a, r)$: the atomic action a needs a unit of the resource r to be executed.
- $\mathsf{Available}^q(r)$: the agent has q units of the resource r, with $0 \leq q \leq MAX$, $MAX > 0$.
- $\mathsf{Saved}^q(r)$: the agent has q units of resource r saved for future usage.

We also consider the operator Res for representing the availability or not of all the resources needed to execute a regular action.

In terms of actions we consider three families of special purpose atomic actions, for the management of resource availability:

- $\mathtt{get}(r)$: the agent gets one more unit of the resource r.
- $\mathtt{save}(r)$: the agent saves one unit of the resource r.
- $\mathtt{free}(r)$: the agent frees one unit of the resource r which he has previously saved.

1.2.2 Syntax

As in $\mathsf{BDI}_{\mathsf{CTL}}$ we distinguish between *state formulae*, which are evaluated in a single situation, and *path formulae* which are evaluated along a path.

Definition 1. *Considering a non-empty set of propositional variables P, a finite set of atomic actions A_{At} that include the set of resource availability management actions, a finite set of resource symbols R and a set of resource quantities $\{0, \ldots, MAX\}$, with $MAX > 0$, the language of $\mathcal{E}_{\mathsf{BDI}}$-formulae is given by the following BNF-grammar:*

- *state-formulae:*
  $\varphi ::= \; p \,|\, \neg\varphi \,|\, \varphi \wedge \varphi \,|\, \langle\alpha\rangle\varphi \,|\, \mathsf{E}\psi \,|\, \mathsf{A}\psi \,|\, \mathsf{Bel}(\varphi) \,|\, \mathsf{Des}(\varphi) \,|\, \mathsf{Int}(\varphi) \,|$
  $\mathsf{Fund}(\varphi) \,|\, \mathsf{Needs}(a, r) \,|\, \mathsf{Available}^q(r) \,|\, \mathsf{Saved}^q(r) \,|\, \mathsf{Cap}(\alpha) \,|\, \mathsf{Res}(\alpha)$
  where $p \in P$, $a \in A_{\mathsf{At}}$, $r \in R$ and $0 \leq q \leq MAX$.

- *path-formulae:*
  $\psi ::= \mathsf{X}\varphi \,|\, (\varphi \mathsf{U} \varphi)$

- *regular actions:*
  $\alpha ::= id \,|\, a \in A_{\mathsf{At}} \,|\, \varphi? \,|\, \alpha;\alpha \,|\, \alpha + \alpha \,|\, \alpha^*$

In addition, we introduce the following abbreviations: $\top$, $\bot$, $\varphi \vee \psi$ and $\varphi \to \psi$ are abbreviations of $\neg(p \wedge \neg p)$ (with p being a fixed element of P), $\neg\top$, $\neg(\neg\varphi \wedge \neg\psi)$ and $\neg(\varphi \wedge \neg\psi)$, respectively; $\mathsf{AF}\varphi$, $\mathsf{EF}\varphi$, $\mathsf{AG}\varphi$ and $\mathsf{EG}\varphi$ are abbreviations of $\mathsf{A}(\top\mathsf{U}\varphi)$, $\mathsf{E}(\top\mathsf{U}\varphi)$, $\neg\mathsf{EF}\neg\varphi$ and $\neg\mathsf{AF}\neg\varphi$, respectively. The formula $[\alpha]\varphi$ stands for $\neg\langle\alpha\rangle\neg\varphi$. Iterated action α^n, with $n \geq 0$, are inductively defined by $\alpha^0 = id$ and $\alpha^{(n+1)} = \alpha; \alpha^n$.

Informally, X means *next temporal state*, U *true until*, F *in a future temporal state*, G *globally true*. The path quantification modal operators E and A mean, respectively, *in one path* and *in all paths*. The regular action modal operator $\langle\alpha\rangle$ means *possibly true after a successful execution of* α.

1.2.3 Semantics

$\mathcal{E}_{\mathsf{BDI}}$-formulae are interpreted in extended $\mathsf{BDI}_{\mathsf{CTL}}$ models, called $\mathcal{E}_{\mathsf{BDI}}$-models. We follow Schild's approach to $\mathsf{BDI}_{\mathsf{CTL}}$ [Sch00], by considering a *situation* as a pair $\delta = \langle w, s\rangle$, where s is a temporal state and w refers to a world (mental state perspective).

Definition 2. *Given a non-empty set of situations Δ, a non-empty set of propositional variables P, a finite set of atomic actions A_{At}, a set of resource symbols R and a positive constant MAX, we define an $\mathcal{E}_{\mathsf{BDI}}$-model as a tuple:*

$$M = \langle \Delta, \mathcal{R}_T, \{\mathcal{R}_a : a \in A_{\mathsf{At}}\}, \mathcal{B}, \mathcal{D}, \mathcal{I}, \mathcal{F}, V, C, avl, svd, needs\rangle$$

such that:

- $\mathcal{R}_T \subseteq \Delta \times \Delta$ *is a temporal accessibility-relation, such that:*
  - *it is serial,* i.e., $\forall \delta \in \Delta$, $\exists \delta' \in \Delta$ *such that* $(\delta, \delta') \in \mathcal{R}_T$*;*
  - *if* $(\langle w_i, s_j\rangle, \langle w_k, s_l\rangle) \in \mathcal{R}_T$*, then* $w_i = w_k$*.*
- $\mathcal{R}_a \subseteq \mathcal{R}_T$ *is an atomic action accessibility-relation, with* $a \in A_{\mathsf{At}}$*;*
- $\mathcal{B}, \mathcal{D}, \mathcal{I}, \mathcal{F} \subseteq \Delta \times \Delta$ *are accessibility-relations for the mental state operators. These relations have the following property (considering* $\mathcal{O} \in \{\mathcal{B}, \mathcal{D}, \mathcal{I}, \mathcal{F}\}$*):*

$$\text{if } (\langle w_i, s_j\rangle, \langle w_k, s_l\rangle) \in \mathcal{O} \text{ then } s_j = s_l;$$

- $V : P \to \wp(\Delta)$ *is a propositional variable labelling function;*
- $C : A_{\mathsf{At}} \to \wp(\Delta)$ *is a capability labelling function;*
- $needs : A_{\mathsf{At}} \to \wp(R)$ *is a function that defines which resource symbols in R are needed to execute each action of* A_{At}*;*
- $avl : \Delta \times R \to \{0, \ldots, MAX\}$ *is a function that for each situation defines which quantity of each resource is available;*
- $svd : \Delta \times R \to \{0, \ldots, MAX\}$ *is a function that for each situation defines which quantity of each resource is saved.*

As in $\mathsf{BDI}_{\mathsf{CTL}}$ path-formulae are evaluated along a path $\pi_\delta = (\delta_0, \delta_1, \delta_2, \ldots)$, such that $\delta = \delta_0$ and $\forall i \geq 0$, $(\delta_i, \delta_{i+1}) \in \mathcal{R}_T$. The k^{th} element of a path π_δ is denoted by $\pi_\delta[k]$.

The accessibility-relation and the capability labelling function for atomic actions are extended to regular actions, as usual in PDL and **KARO**. We denote them, respectively, by R^A and c^A.

R^A	$:$	$A_{\mathsf{Ra}} \to (\Delta \times \Delta)$	c^A	$:$	$A_{\mathsf{Ra}} \to \wp(\Delta)$
R^A_a	$=$	$\mathcal{R}_a$	c^A_a	$=$	$C(a)$
R^A_φ	$=$	$\{(\delta, \delta) \mid M, \delta \models \varphi\}$	$c^A_{\varphi?}$	$=$	Δ
R^A_{id}	$=$	Id_Δ	c^A_{id}	$=$	Δ
$R^A_{\alpha;\beta}$	$=$	$R^A_\alpha \circ R^A_\beta$	$c^A_{\alpha;\beta}$	$=$	$\{\delta \mid \delta \in c^A_\alpha \wedge \exists \delta' \in \Delta((\delta, \delta') \in R^A_\alpha \wedge \delta' \in c^A_\beta)\}$
$R^A_{\alpha+\beta}$	$=$	$R^A_\alpha \cup R^A_\beta$	$c^A_{\alpha+\beta}$	$=$	$c^A_\alpha \cup c^A_\beta$
$R^A_{\alpha^0}$	$=$	R^A_{id}	$c^A_{\alpha^0}$	$=$	c^A_{id}
$R^A_{\alpha^{(n+1)}}$	$=$	$R^A_{(\alpha;\alpha^n)}$	$c^A_{\alpha^{(n+1)}}$	$=$	$c^A_{\alpha;\alpha^n}$
$R^A_{\alpha^*}$	$=$	$\bigcup_{i\geq 0}\left(R^A_{\alpha^i}\right)$	$c^A_{\alpha^*}$	$=$	$\bigcup_{i\geq 0}\left(c^A_{\alpha^n}\right)$

For the modelling of resources the function avl and svd verify the following properties:

- the total amount of resources which the agent can deal with cannot be greater than MAX: $\forall \delta \in \Delta, \forall r \in R$, $0 \leq avl(\delta, r) + svd(\delta, r) \leq MAX$.

- the execution of an atomic action consumes one unit of each resource needed for the execution of that action:
  $\forall r \in needs(a), \forall(\delta, \delta') \in \mathcal{R}_a,\ avl(\delta', r) = avl(\delta, r) - 1$.

Also, we assume that for the resource management atomic actions we have:

$$needs(\texttt{get}(r)) = needs(\texttt{save}(r)) = needs(\texttt{free}(r)) = \emptyset,\ \forall r \in R$$

The availability of resources for executing regular actions is given by:

$$\begin{array}{ll}
res: & A_{\mathsf{Ra}} \to \wp(\Delta) \\
res_a = & \begin{cases} \{\delta \mid \text{if } r \in needs(a) \text{ then } avl(r,\delta) \geq 1\}, & \text{if } needs(a) \neq \emptyset \\ \\ \Delta, & \text{otherwise.} \end{cases} \\
res_{\varphi?} = & \Delta \\
res_{\alpha;\beta} = & \{\delta \mid \delta \in res_\alpha \wedge \exists(\delta,\delta') \in R^A_\alpha,\ \delta' \in res_\beta\} \\
res_{\alpha+\beta} = & res_\alpha \cup res_\beta \\
res_{\alpha^*} = & \{\delta \mid \exists n \in N(\delta \notin res_{\alpha^n} \wedge \forall m, 0 \leq m < n, \delta \in res_{\alpha^m})\}
\end{array}$$

The intuition behind the value of the resource availability function res for α^* is that the iterated execution of α is bounded to the existence of a finite amount of resources.

We are now in conditions to define the satisfiability for an $\mathcal{E}_{\mathsf{BDI}}$-formula.

Definition 3. *Let M be an $\mathcal{E}_{\mathsf{BDI}}$-model and δ a situation. The* satisfiability *of an $\mathcal{E}_{\mathsf{BDI}}$-formula is defined inductively as follows:*

- *state formulae satisfaction rules:*
  - $M, \delta \models p$ *iff* $\delta \in V(p)$
  - $M, \delta \models \neg\varphi$ *iff* $M, \delta \not\models \varphi$
  - $M, \delta \models \varphi \wedge \psi$ *iff* $M, \delta \models \varphi$ *e* $M, \delta \models \psi$
  - $M, \delta \models \mathsf{E}\psi$ *iff exists a path* π_δ *such that* $M, \pi_\delta \models \psi$
  - $M, \delta \models \mathsf{A}\psi$ *iff for all paths* π_δ, $M, \pi_\delta \models \psi$
  - $M, \delta \models \langle\alpha\rangle\varphi$ *iff exists* $(\delta, \delta') \in R^A_\alpha$ *such that* $M, \delta' \models \varphi$
  - $M, \delta \models \mathsf{Bel}(\varphi)$ *iff for all* $(\delta, \delta') \in \mathcal{B}$, $M, \delta' \models \varphi$
  - $M, \delta \models \mathsf{Des}(\varphi)$ *iff for all* $(\delta, \delta') \in \mathcal{D}$, $M, \delta' \models \varphi$
  - $M, \delta \models \mathsf{Int}(\varphi)$ *iff for all* $(\delta, \delta') \in \mathcal{I}$, $M, \delta' \models \varphi$
  - $M, \delta \models \mathsf{Fund}(\varphi)$ *iff for all* $(\delta, \delta') \in \mathcal{F}$, $M, \delta' \models \varphi$
  - $M, \delta \models \mathsf{Cap}(\alpha)$ *iff* $\delta \in c^A_\alpha$
  - $M, \delta \models \mathsf{Needs}(a, r)$ *iff* $r \in needs(a)$
  - $M, \delta \models \mathsf{Available}^q(r)$ *iff* $avl(\delta, r) = q$
  - $M, \delta \models \mathsf{Saved}^q(r)$ *iff* $svd(\delta, r) = q$
  - $M, \delta \models \mathsf{Res}(\alpha)$ *iff* $\delta \in res_\alpha$
- *path formulae satisfaction rules:*
  - $M, \pi_\delta \models \mathsf{X}\varphi$ *iff* $M, \pi_\delta[1] \models \varphi$
  - $M, \pi_\delta \models \varphi_1 \mathsf{U} \varphi_2$ *iff* $\exists\, k \geq 0$ *such that* $M, \pi_\delta[k] \models \varphi_2$ *and* $\forall j,\ 0 \leq j < k\ \big(M, \pi_\delta[j] \models \varphi_1\big)$

1.2.4 Properties of $\mathcal{E}_{\mathsf{BDI}}$

The axiomatic characterisation of $\mathcal{E}_{\mathsf{BDI}}$'s modal operators of time and BDI mental states are the same as in $\mathsf{BDI}_{\mathsf{CTL}}$-logic. The modal operator Fund, for fundamental desires, follows the axiom set of Des and Int operators, which is the KD system [HM92], *i.e.*, $\mathcal{F}$ is a serial accessibility-relation. The Bel operator verifies the $KD45$ axioms, *i.e.*, $\mathcal{B}$ is an equivalence relation.

The temporal operators follow the axioms of CTL and the action execution operators verify the axioms of PDL. Since both branching-time and regular action execution structures coexist, we have the following properties:

Theorem 1. *Let M be an $\mathcal{E}_{\mathsf{BDI}}$-model, a an atomic action and α a regular action. We have:*

1. *if $M, \delta \models \langle a \rangle \varphi$ then $M, \delta \models \mathsf{EX}\varphi$.*
2. *if $M, \delta \models \langle \alpha \rangle \varphi$ then $M, \delta \models \mathsf{EF}\varphi$.*
3. *if $M, \delta \models \langle \alpha^* \rangle \varphi$ then $M, \delta \models \mathsf{E}(\langle \alpha \rangle \top \mathsf{U} \varphi)$.*

Proof. For (1) supose that $M, \delta \models \langle a \rangle \varphi$ and $a \neq id$. So, exists $(\delta, \delta') \in \mathcal{R}_a$ such that $M, \delta' \models \varphi$. By definition $(\delta, \delta') \in \mathcal{R}_T$ and also $\pi_\delta = (\delta, \delta', \ldots)$ such that $M, \pi_\delta[1] \models \varphi$. By definition we get $M, \delta \models \mathsf{EX}\varphi$.

For (2) we present just the case of $\alpha = \beta + \gamma$. Supose that $\alpha = \beta + \gamma$. Thus we have $M, \delta \models \langle \beta \rangle \varphi$ or $M, \delta \models \langle \gamma \rangle \varphi$. Using the induction hypothesis we have $M, \delta \models \mathsf{EF}\varphi$ or $M, \delta \models \mathsf{EF}\varphi$, *i.e.* $M, \delta \models \mathsf{EF}\varphi$.

For (3) we have that, by definition, $\exists(\delta, \delta') \in R^A_{\alpha^*}$ such that $M, \delta' \models \varphi$ which is equivallent to $\exists(\delta, \delta') \in \left(\cup_{n \geq 0} R^A_{\alpha^*} \right)$ such that $M, \delta' \models \varphi$. Then we also have that $\exists i \geq 0, \exists(\delta, \delta') \in R^A_{\alpha^i}$ such that $M, \delta' \models \varphi$. Therefore we also have situations $\delta_0 = \delta, \delta_1, \ldots, \delta_{i-1}, \delta_i = \delta'$ such that $\forall k, 0 \leq k \leq i-1$, $(\delta_k, \delta_{k+1}) \in R^A_\alpha$. By definition we have $\forall k, 0 \leq k \leq i-1$, $(\delta_k, \delta_{k+1}) \in \mathcal{R}_T$. Now, considering the path $\pi'_\delta = (\delta, \delta_1, \ldots, \delta_{i-1}, \delta_i)$ we have $M, \pi'_\delta[i] \models \varphi$ e $\forall k$, $0 \leq k < i$, $M, \pi'_\delta[k] \models \langle \alpha \rangle \top$. Now, if we consider the path $\pi_\delta = \pi'_\delta . \pi''_{\delta_{i+1}}$, for any path $\pi''_{\delta_{i+1}} = (\delta_{i+1}, \ldots)$, with $(\delta_i, \delta_{i+1}) \in \mathcal{R}_T$ we have, by definition, that $M, \pi_\delta \models \langle \alpha \rangle \top \mathsf{U} \varphi$. Thus we get $M, \delta \models \mathsf{E}(\langle \alpha \rangle \top \mathsf{U} \varphi)$. □

Capabilities are characterised similarly to *abilities* in the **KARO** framework. The axioms for the Cap modal operator are:

- $\mathsf{Cap}(\varphi?) \rightarrow \top$
- $\mathsf{Cap}(\alpha; \beta) \rightarrow \mathsf{Cap}(\alpha) \wedge \langle \alpha \rangle \mathsf{Cap}(\beta)$
- $\mathsf{Cap}(\alpha + \beta) \rightarrow \mathsf{Cap}(\alpha) \vee \mathsf{Cap}(\beta)$
- $\mathsf{Cap}(\alpha^*) \rightarrow \mathsf{Cap}(\alpha) \wedge \langle \alpha \rangle \mathsf{Cap}(\alpha^*)$
- $\mathsf{Cap}(\alpha) \wedge \langle \alpha^* \rangle (\mathsf{Cap}(\alpha) \rightarrow \langle \alpha \rangle \mathsf{Cap}(\alpha)) \rightarrow \mathsf{Cap}(\alpha^*)$

Resource availability for regular actions follows almost the same axioms that characterise the Cap operator. However, the unbounded composition operator * behaves differently, bounding the execution of an action α^* to a finite number of compositions of α. This composition stops when there are no resources to execute α once more. The Res operator verifies the following axioms:

- $\mathsf{Res}(\varphi?) \rightarrow \top$
- $\mathsf{Res}(\mathtt{get}(r)) \rightarrow \top$
- $\mathsf{Res}(\mathtt{save}(r)) \rightarrow \top$
- $\mathsf{Res}(\mathtt{free}(r)) \rightarrow \top$
- $\mathsf{Res}(\alpha; \beta) \rightarrow \mathsf{Res}(\alpha) \wedge \langle \alpha \rangle \mathsf{Res}(\beta)$
- $\mathsf{Res}(\alpha + \beta) \rightarrow \mathsf{Res}(\alpha) \vee \mathsf{Res}(\beta)$

- $\mathsf{Res}(\alpha^*) \to \mathsf{E}(\langle\alpha\rangle\top\mathsf{U}\neg\mathsf{Res}(\alpha))$

Resources are also characterised by axioms which deal with the modal operators Available, Needs and Saved. First we define some abbreviations that represent, respectively, the maximum quantity of available and saved resources, in a situation:

- $\mathsf{MaxAvl}^q(r) =_{def} \mathsf{Available}^q(r) \wedge \neg\mathsf{Available}^{(q+1)}(r)$
- $\mathsf{MaxSvd}^q(r) =_{def} \mathsf{Saved}^q(r) \wedge \neg\mathsf{Saved}^{(q+1)}(r)$

The following axioms characterise the interaction between action execution and resource availability:

- $\mathsf{MaxAvl}^q(r) \wedge \mathsf{Needs}(a,r) \to [a]\mathsf{MaxAvl}^{(q-1)}(r)$, $0 < q \leq MAX$
- $\mathsf{MaxAvl}^q(r) \wedge \neg\mathsf{Needs}(a,r) \to [a]\mathsf{MaxAvl}^q(r)$, $0 \leq q \leq MAX$
- $\mathsf{MaxSvd}^q(r) \wedge \mathsf{Needs}(a,r) \to [a]\mathsf{MaxSvd}^q(r)$, $0 \leq q \leq MAX$
- $\mathsf{MaxSvd}^q(r) \wedge \neg\mathsf{Needs}(a,r) \to [a]\mathsf{MaxSvd}^q(r)$, $0 \leq q \leq MAX$

The following axioms characterise the dynamics of the availability of resources, considering both resource availability limits and the execution of the special actions to manage them. We have:

- resource availability limits:
  - $\mathsf{Available}^0(r)$, $\forall r \in R$
  - $\mathsf{Saved}^0(r)$, $\forall r \in R$
  - $\mathsf{Available}^q(r) \to \mathsf{Available}^{(q-1)}(r)$, $1 < q \leq MAX$
  - $\mathsf{Saved}^q(r) \to \mathsf{Saved}^{(q-1)}(r)$, $1 < q \leq MAX$
- resource availability and resource management actions:
  - $\mathsf{Needs}(\mathtt{get}(r), r') \to \bot$, $\forall r, r' \in R$
  - $\mathsf{Needs}(\mathtt{save}(r), r') \to \bot$, $\forall r, r' \in R$
  - $\mathsf{Needs}(\mathtt{free}(r), r') \to \bot$, $\forall r, r' \in R$
  - $\mathsf{MaxAvl}^q(r) \to [\mathtt{get}(r)]\mathsf{MaxAvl}^{(q+1)}(r)$, for $0 \leq q < MAX$
  - $\mathsf{MaxAvl}^q(r) \wedge \mathsf{MaxSvd}^{q'}(r) \to [\mathtt{save}(r)](\mathsf{MaxAvl}^{(q-1)}(r) \wedge \mathsf{MaxSvd}^{(q'+1)}(r))$, with $0 \leq q + q' \leq MAX$
  - $\mathsf{MaxAvl}^q(r) \wedge \mathsf{MaxSvd}^{q'}(r) \to [\mathtt{free}(r)](\mathsf{MaxAvl}^{(q+1)}(r) \wedge \mathsf{MaxSvd}^{(q'-1)}(r))$, with $0 \leq q + q' \leq MAX$

1.2.5 Decidability

The decidability of $\mathcal{E}_{\mathsf{BDI}}$ is obtained by transforming an original $\mathcal{E}_{\mathsf{BDI}}$-formula φ into a new formula φ' which is evaluated in a modified $\mathcal{E}_{\mathsf{BDI}}$-model. This modified model is a $\mathsf{BDI}_{\mathsf{CTL}}$-model which considers the accessibility relation $\mathcal{F}$ and special propositional variables which represent the execution of atomic actions, capabilities and resource availability.

Let $\mathcal{L}$ be an $\mathcal{E}_{\mathsf{BDI}}$ language and P the set of propositional variables. We define a new language $\mathcal{L}'$ equal to $\mathcal{L}$ except that it has a new set of propositions P' that is the union of the following disjunct sets:

- the set of propositional variables P,

- the set of propositional variables which represent the atomic actions: $\{done\_a \,|\, a \in A_{\mathsf{At}}\}$,

- the set of propositional variables which represent the capabilities for atomic actions: $\{cap\_a \,|\, a \in A_{\mathsf{At}}\}$,

- the set of propositional variables which represent the resources for atomic actions: $\{res\_a \,|\, a \in A_{\mathsf{At}}\}$,

- a set of propositional variables for representing the various quantities of resources available: $\{avl\_q\_r, svd\_q\_r \,|\, q \in \{0, \ldots, MAX\}, r \in R\}$,

- a set of propositional variables for representing the resources needed for the execution of each atomic action: $\{needs\_a\_r \,|\, a \in A_{\mathsf{At}}, r \in R\}$.

Considering an $\mathcal{E}_{\mathsf{BDI}}$-model M, the modified model M' is defined as follows, extending the propositional labelling function of M.

Definition 4. *Let M be an $\mathcal{E}_{\mathsf{BDI}}$-model such that:*

$$M = \langle \Delta, \mathcal{R}_T, \{\mathcal{R}_a : a \in A_{\mathsf{At}}\}, \mathcal{B}, \mathcal{D}, \mathcal{I}, \mathcal{F}, V, C, avl, svd, needs \rangle,$$

a model M' is a tuple:

$$M' = \langle \Delta, \mathcal{R}_T, \mathcal{B}, \mathcal{D}, \mathcal{I}, \mathcal{F}, V' \rangle,$$

such that $V' : P' \to \wp(\Delta)$ is defined as follows:

- $V'(p) = V(p)$,
- $V'(done\_a) = \{\delta' \,|\, \exists(\delta, \delta') \in R_a^A\}$,
- $V'(cap\_a) = C(a)$,
- $V'(res\_a) = res_a$,
- $V'(avl\_q\_r) = \{\delta \,|\, M, \delta \models \mathsf{Available}^q(r)\}$,
- $V'(svd\_q\_r) = \{\delta \,|\, M, \delta \models \mathsf{Saved}^q(r)\}$,
- $V'(needs\_a\_r) = \{\delta \,|\, M, \delta \models \mathsf{Needs}(a, r)\}$.

Note that in V' only atomic actions are considered. Therefore, any $\mathcal{E}_{\mathsf{BDI}}$-formula must be normalised into an equivalent one where only atomic actions can occur. The normalisation ξ is defined as follows.

Definition 5. *Let φ be a $\mathcal{E}_{\mathsf{BDI}}$-logic. It exists a formula $\varphi' \equiv \xi(\varphi)$, such that ξ is a normalisation inductively defined by:*

- *normalisation of regular action formulae:*

$$\begin{array}{lcl}
\xi(\langle a \rangle \varphi) & = & \langle a \rangle \xi(\varphi), \\
\xi(\langle \psi? \rangle \varphi) & = & \xi(\psi \wedge \varphi), \\
\xi(\langle \alpha \rangle (\varphi \vee \psi)) & = & \xi(\langle \alpha \rangle \varphi) \vee \xi(\langle \alpha \rangle \psi), \\
\xi(\langle \alpha ; \beta \rangle \varphi) & = & \xi(\langle \alpha \rangle \langle \beta \rangle \varphi), \\
\xi(\langle \alpha + \beta \rangle \varphi) & = & \xi(\langle \alpha \rangle \varphi) \vee \xi(\langle \beta \rangle \varphi), \\
\xi(\langle id \rangle \varphi) & = & \xi(\varphi), \\
\xi(\langle \alpha^{(n+1)} \rangle \varphi) & = & \xi(\langle \alpha \rangle \langle \alpha^n \rangle \varphi), \\
\xi(\langle \alpha^* \rangle \varphi) & = & \xi(\mathsf{E}(\langle \alpha \rangle \top \mathsf{U} \varphi)).
\end{array}$$

- *normalisation of capability formulae:*

$$\begin{array}{lcl}\xi(\mathsf{Cap}(a)) & = & \mathsf{Cap}(a),\\ \xi(\mathsf{Cap}(\varphi?)) & = & \top,\\ \xi(\mathsf{Cap}(\alpha;\beta)) & = & \xi(\mathsf{Cap}(\alpha)\wedge\langle\alpha\rangle\mathsf{Cap}(\beta)),\\ \xi(\mathsf{Cap}(\alpha+\beta)) & = & \xi(\mathsf{Cap}(\alpha))\vee\xi(\mathsf{Cap}(\beta)),\\ \xi(\mathsf{Cap}(\alpha^*)) & = & \xi(\mathsf{E}(\mathsf{Cap}(\alpha)\wedge\langle\alpha\rangle\mathsf{Cap}(\alpha))\mathsf{U}\top)).\end{array}$$

- *normalisation of resource formulae:*

$$\begin{array}{lcl}\xi(\mathsf{Res}(a)) & = & \mathsf{Res}(a),\\ \xi(\mathsf{Res}(\varphi?)) & = & \top,\\ \xi(\mathsf{Res}(\alpha;\beta)) & = & \xi(\mathsf{Res}(\alpha)\wedge\langle\alpha\rangle\mathsf{Res}(\beta)),\\ \xi(\mathsf{Res}(\alpha+\beta)) & = & \xi(\mathsf{Res}(\alpha))\vee\xi(\mathsf{Res}(\beta)),\\ \xi(\mathsf{Res}(\alpha^*)) & = & \xi(\mathsf{E}(\mathsf{Res}(\alpha)\wedge\langle\alpha\rangle\top))\mathsf{U}\neg\mathsf{Res}(\alpha))).\end{array}$$

- *normalisation of other formulae:*

$$\begin{array}{lcl}\xi(\top) & = & \top,\\ \xi(p) & = & p,\\ \xi(\neg\varphi) & = & \neg(\xi(\varphi)),\\ \xi(\varphi\wedge\psi) & = & \xi(\varphi)\wedge\xi(\psi),\\ \xi(\mathsf{A}\psi) & = & \mathsf{A}(\xi(\psi)),\\ \xi(\mathsf{E}\psi) & = & \mathsf{E}(\xi(\psi)),\\ \xi(\mathsf{X}\varphi) & = & \mathsf{X}(\xi(\varphi)),\\ \xi(\varphi_1\mathsf{U}\varphi_2) & = & (\xi(\varphi_1)\mathsf{U}\xi(\varphi_2)),\\ \xi(\mathsf{Bel}(\varphi)) & = & \mathsf{Bel}(\xi(\varphi)),\\ \xi(\mathsf{Des}(\varphi)) & = & \mathsf{Des}(\xi(\varphi)),\\ \xi(\mathsf{Int}(\varphi)) & = & \mathsf{Int}(\xi(\varphi)),\\ \xi(\mathsf{Fund}(\varphi)) & = & \mathsf{Fund}(\xi(\varphi)),\end{array}$$

After normalisation, we apply the transformation defined below, so that the resulting formula can be evaluated in a model M'.

Definition 6. *Let φ be an normalised $\mathcal{E}_{\mathsf{BDI}}$-formula. The transformation of φ to φ' is given by τ, inductively defined as follows:*

- *propositional-formulae:*

$$\begin{array}{lcl}\tau(\top) & = & \top,\\ \tau(p) & = & p,\\ \tau(\neg\varphi) & = & \neg(\tau(\varphi)),\\ \tau(\varphi\wedge\psi) & = & \tau(\varphi)\wedge\tau(\psi).\end{array}$$

- *temporal-formulae:*

$$\begin{array}{lcl}\tau(\mathsf{A}\psi) & = & \mathsf{A}(\tau(\varphi)),\\ \tau(\mathsf{E}\psi) & = & \mathsf{E}(\tau(\varphi)),\\ \tau(\mathsf{X}\varphi) & = & \mathsf{X}(\tau(\varphi)),\\ \tau(\varphi_1\mathsf{U}\varphi_2) & = & (\tau(\varphi_1)\mathsf{U}\tau(\varphi_2)).\end{array}$$

- *action execution formulae:*

$$\begin{array}{lcl}\tau(\langle a\rangle\varphi) & = & \mathsf{EX}(done\_a\wedge\tau(\varphi)),\\ \tau([a]\varphi) & = & \mathsf{AX}(done\_a\rightarrow\tau(\varphi)).\end{array}$$

- *mental-state formulae:*

$$\begin{array}{lcl}\tau(\mathsf{Bel}(\varphi)) & = & \mathsf{Bel}(\tau(\varphi)),\\ \tau(\mathsf{Des}(\varphi)) & = & \mathsf{Des}(\tau(\varphi)),\\ \tau(\mathsf{Int}(\varphi)) & = & \mathsf{Int}(\tau(\varphi)),\\ \tau(\mathsf{Fund}(\varphi)) & = & \mathsf{Fund}(\tau(\varphi)),\\ \tau(\mathsf{Fear}(\varphi)) & = & \mathsf{Fear}(\tau(\varphi)).\end{array}$$

- *capabilities and resources formulae:*

$$\begin{array}{lcl} \tau(\mathsf{Cap}(a)) & = & cap\_a, \\ \tau(\mathsf{Res}(a)) & = & res\_a, \\ \tau(\mathsf{Needs}(a,r)) & = & needs\_a\_r, \\ \tau(\mathsf{Available}^q(r)) & = & \bigwedge_{0\leq s\leq q}(avl\_s\_r), \\ \tau(\mathsf{Saved}^q(r)) & = & \bigwedge_{0\leq s\leq q}(svd\_s\_r). \end{array}$$

Now we can present the following theorem.

Theorem 2. *Let M be an $\mathcal{E}_{\mathsf{BDI}}$-model, δ a situation and φ a normalised $\mathcal{E}_{\mathsf{BDI}}$-formula. If $M,\delta \models \varphi$ then $M',\delta \models \tau(\varphi)$.*

Proof. Here we present just the cases of $\varphi = \mathsf{Cap}(a)$ and $\varphi = \langle a\rangle\psi$ from the full proof for the above theorem. The proof was done using induction on the structure of $\mathcal{E}_{\mathsf{BDI}}$-formulae.

Let $\varphi = \mathsf{Cap}(a)$ and assume $M,\delta \models \mathsf{Cap}(a)$. By definition of M' we have $\delta \in V'(cap\_a)$ and so $M',\delta \models cap\_a$. Since $\tau(\mathsf{Cap}(a)) = cap\_a$, we have $M',\delta \models \tau(\mathsf{Cap}(a))$.

Lets consider now that $\varphi = \langle a\rangle\psi$. Therefore we have that $\exists(\delta,\delta') \in R_a^A$ such that $M,\delta' \models \varphi$. By the induction hypothesis $M',\delta' \models \tau(\varphi)$. We also have $\delta' \in V'(done\_a)$ since $(\delta,\delta') \in R_a^A$. Thus $M',\delta' \models \tau(\varphi)$ e $M',\delta' \models done_a$ and so, by definition we have $M',\delta' \models \tau(\varphi) \wedge done\_a$. Since $(\delta,\delta') \in \mathcal{R}_T$ we have a path $\pi_\delta = (\delta,\delta',\ldots)$ such that $M',\pi_\delta[1] \models \tau(\varphi) \wedge done\_a$. By definition we also have $M',\delta \models \mathsf{EX}(\tau(\varphi) \wedge done\_a)$. □

Using this theorem, we obtain the decidability of a $\mathcal{E}_{\mathsf{BDI}}$-formula φ by transforming it into φ' and applying to the latter the tableau construction for $\mathsf{BDI_{CTL}}$, with a rule for expanding formulas containing the Fund modal operator. Therefore we have:

Theorem 3. *The $\mathcal{E}_{\mathsf{BDI}}$ logic is decidable.*

The details and proofs for the decidability of $\mathcal{E}_{\mathsf{BDI}}$ are presented in the extended version of this paper.

1.3 Preliminaries for modelling emotions in $\mathcal{E}_{\mathsf{BDI}}$

In this section we present a series of concepts which will be useful for modelling emotions in $\mathcal{E}_{\mathsf{BDI}}$. These concepts refer to conditions that are the basis for modelling the activation of emotions and the consequences that these emotions have in the behaviour of the agent.

1.3.1 Resource management actions

We begin by defining special regular actions for dealing with resource management. For that we consider the following abbreviations for regular actions:

- $\texttt{If}(\varphi,\alpha) =_{def} (\varphi?;\alpha)$
- $\texttt{IfE}(\varphi,\alpha,\beta) =_{def} \texttt{If}(\varphi,\alpha) + \texttt{If}(\neg\varphi,\beta)$
- $\texttt{WhileDo}(\varphi,\alpha) =_{def} ((\varphi?;\alpha)^*);\neg\varphi?$

We consider also a special function which given a finite set of regular actions, returns the composition of all the actions in that set:

$$\begin{array}{lcl} seq & : & \wp(A_{\mathsf{Ra}}) \rightarrow A_{\mathsf{Ra}} \\ seq(S) & = & \alpha_1;\alpha_2;\ldots;\alpha_n,\ \alpha_i \in S, 1 \leq i \leq n \end{array}$$

Based on the atomic actions for the of resource management, we define the following set of resource management regular actions:

GET: the agent gets all the resources needed to execute some action. Considering:
$Cond_1(a,r) \ = \ \mathsf{Needs}(a,r) \wedge \mathsf{MaxAvl}^0(r)$
we have:
$\mathtt{GET}(a) = seq(\{\mathtt{If}(Cond_1(a,r), \mathtt{get}(r)) \mid r \in R\})$

SAVE: the agent saves a unit of each resource needed to execute an action. Considering:
$Cond_2(a,r) \ = \ \mathsf{Needs}(a,r) \wedge \neg\mathsf{MaxSvd}^1(r)$
we have:
$\mathtt{SAVE}(a) = seq(\{\mathtt{If}(Cond_2(a,r), \mathtt{IfE}(\mathsf{Avl}(r), \mathtt{save}(r), \mathtt{get}(r); \mathtt{save}(r))) \mid r \in R\})$

FREE: the agent frees the resources previously saved for executing an action. Considering:
$Cond_3(a,r) \ =_{def} \ \mathsf{Needs}(a,r) \wedge \mathsf{Svd}(r)$
we have:
$\mathtt{FREE}(a) = seq(\{\mathtt{If}(Cond_3(a,r), \mathtt{free}(r)) \mid r \in R\})$

1.3.2 Proposition achievement

For the agent to succeed in the execution of an action it must have both the capability and resources for that action. We denote the existence of both of them as *effective capability.* Formally we have:

- $\mathsf{EffCap}(\alpha) \ =_{def} \ \mathsf{Cap}(\alpha) \wedge \mathsf{Res}(\alpha)$

The agent also considers if it *can* or *cannot* execute some action to achieve the truth of some proposition. Formally we have:

- $\mathsf{Can}(\alpha, \varphi) \ =_{def} \ \mathsf{Bel}(\langle\alpha\rangle\varphi \wedge \mathsf{EffCap}(\alpha))$
- $\mathsf{Cannot}(\alpha, \varphi) \ =_{def} \ \mathsf{Bel}(\neg\langle\alpha\rangle\varphi \vee \neg\mathsf{EffCap}(\alpha))$

1.3.3 Risk and favourable conditions

The activation of emotions is based on conditions of the environment that show to be positive or negative to the desires and fundamental desires of the agent. First we define the following conditions:

Risk condition: a proposition φ is said to be *at risk* if there is a next situation in which $\neg\varphi$ is true:
$\mathsf{AtRisk}(\varphi) \ =_{def} \ \mathsf{EX}(\neg\varphi)$

Possibly at risk: a proposition φ is said to be *possibly at risk* if there exists a future situation where $\neg\varphi$ is true. Formally this is defined as:
$\mathsf{PossAtRisk}(\varphi) \ =_{def} \ \mathsf{EF}(\neg\varphi)$

Safe: a proposition φ is said to be *safe* if it will always be true in the future. Formally we have:
$\mathsf{Safe}(\varphi) \ =_{def} \ \mathsf{AF}(\varphi)$

On believing on the above, and the propositions being either fundamental desires or only desires, the agent distinguishes between three types of conditions for activating emotions:

1. *Threats:* a *threat* is a condition of the environment in which a fundamental desire is in imminent risk of failure. We consider the following kinds of threats:

   - a fundamental desire φ is said to be *threatened* if the agent believes that φ is at risk:
     $\mathsf{Threatened}(\varphi) \ =_{def} \ \mathsf{Bel}(\mathsf{AtRisk}(\varphi)) \wedge \mathsf{Fund}(\varphi)$

- a fundamental desire φ is said to be *threatened by a proposition* ψ if the agent believes that the truth of ψ implies φ being at risk:
  $\mathsf{ThreatProp}(\psi, \varphi) =_{def} \mathsf{Bel}(\psi \to \mathsf{AtRisk}(\varphi)) \wedge \mathsf{Fund}(\varphi)$

- a fundamental desire φ is said to be *threatened by the execution of an action* a if the agent believes that the successful execution of a will put φ at risk:
  $\mathsf{ThreatAct}(a, \varphi) =_{def} \mathsf{Bel}(\langle a \rangle \mathsf{AtRisk}(\varphi)) \wedge \mathsf{Fund}(\varphi)$

- a fundamental desire φ is said to be *threatened by lack of effective resources to execute an action* a if the agent believes that the non-execution of a due to the lack of effective resources will put φ at risk:
  $\mathsf{ThreatsEffC}(a, \varphi) =_{def} \mathsf{Bel}(\neg \mathsf{EffCap}(a) \to \mathsf{AtRisk}(\langle a \rangle \varphi)) \wedge \mathsf{Fund}(\varphi)$

2. *Not favourable:* a condition is *not favourable* if it reveals a possible failure of one of the agent's desires, in the future. As in the case of the threats, we consider the following kinds of not favourable conditions:

   - $\mathsf{NotFavourable}(\varphi) =_{def} \mathsf{Bel}(\mathsf{PossAtRisk}(\varphi)) \wedge \mathsf{Des}(\varphi)$
   - $\mathsf{NotFavourableProp}(\psi, \varphi) =_{def} \mathsf{Bel}(\psi \to \mathsf{PossAtRisk}(\varphi)) \wedge \mathsf{Des}(\varphi)$
   - $\mathsf{NotFavourableAct}(\alpha, \varphi) =_{def} \mathsf{Bel}(\langle \alpha \rangle \mathsf{PossAtRisk}(\varphi)) \wedge \mathsf{Des}(\varphi)$
   - $\mathsf{NotFavourableEffC}(\alpha, \varphi) =_{def} \mathsf{Bel}(\neg \mathsf{EffCap}(\alpha) \to \mathsf{PossAtRisk}(\langle \alpha \rangle \varphi)) \wedge \mathsf{Des}(\varphi)$

   Note that here we consider regular actions instead of atomic ones since the risk condition is not bounded to verify in a next situation.

3. *Favourable:* a condition is said to be *favourable* if it refers to a current situation of the environment in which a desire of the agent has the possibility to be achieved. We define the following kinds of favourable conditions:

   - $\mathsf{Favourable}(\varphi) =_{def} \mathsf{Bel}(\mathsf{Safe}(\varphi)) \wedge \mathsf{Des}(\varphi)$
   - $\mathsf{FavourableProp}(\varphi, \psi) =_{def} \mathsf{Bel}(\psi \to \mathsf{Safe}(\varphi)) \wedge \mathsf{Des}(\varphi)$
   - $\mathsf{FavorableAct}(\alpha, \varphi) =_{def} \mathsf{Bel}(\langle \alpha \rangle \mathsf{Safe}(\varphi)) \wedge \mathsf{Des}(\varphi)$

1.4 Modelling emotions in $\mathcal{E}_{\mathsf{BDI}}$

In this section we present the modelling of three emotions within $\mathcal{E}_{\mathsf{BDI}}$ logic: *Fear*, *Anxiety* and *Self-Confidence*. For each of these emotions we model both its activation conditions and the effects that their presence have in the future behaviour of an Emotional-BDI agent. This modelling is based in the work of Oliveira & Sarmento in [OS03].

The activation condition of each of the emotions corresponds precisely to a condition defined in the previous section. We opted by this approach to avoid the logical omniscience problem [Whi03]. The use of a notation $Emotion(F(\varphi))$ allows a more intuitive meaning and can help in the future development of a formal calculus for (emotional) $\mathcal{E}_{\mathsf{BDI}}$-formulae.

1.4.1 Fear

The activation of fear occurs when a fundamental desire of the agent is put at risk of failure. Using other words, fear is activated when the agent detects a threat. Therefore we have the following kinds of fear:

- $\mathsf{Fear}(\neg\varphi) \equiv \mathsf{Threatened}(\varphi)$
- $\mathsf{Fear}(\psi \to \neg\varphi) \equiv \mathsf{ThreatsProp}(\psi, \varphi)$
- $\mathsf{Fear}(\langle a \rangle \neg\varphi) \equiv \mathsf{ThreatsAct}(a, \varphi)$
- $\mathsf{Fear}(\neg\mathsf{EffCap}(a) \to \neg\varphi) \equiv \mathsf{ThreatsEffC}(a, \varphi)$

The main effect of fear is bringing the agent into a cautious perspective towards the environment and, in particular, to the threat he detected. Depending on the kind of threat, the agent will aim at avoiding that threat. We consider the following behaviours under the effect of fear:

- if the agent can avoid a threat through the execution of an action a it is possible that he executes a:
  $\mathsf{Fear}(\neg\varphi) \wedge \mathsf{Can}(a, \varphi) \to \langle a \rangle(\mathsf{Bel}(\varphi) \wedge \neg\mathsf{Fear}(\neg\varphi))$

- if the agent cannot avoid the threat through an action a then he does not execute it:
  $\mathsf{Fear}(\neg\varphi) \wedge \mathsf{Cannot}(a, \varphi) \to \neg\langle a \rangle\top$

- if the agent can avoid a proposition which is a threat, or can make the proposition and the fundamental desire coexist – both through the execution of an action – then the agent executes that action:
  $\mathsf{Fear}(\psi \to \neg\varphi) \wedge \mathsf{Can}(a, \neg\psi) \to \langle a \rangle(\mathsf{Bel}(\neg\psi \wedge \varphi) \wedge \neg\mathsf{Fear}(\neg\varphi))$
  $\mathsf{Fear}(\psi \to \neg\varphi) \wedge \mathsf{Can}(a, \psi \wedge \varphi) \to \langle a \rangle(\mathsf{Bel}(\psi \wedge \varphi) \wedge \neg\mathsf{Fear}(\neg\varphi))$

- if the agent cannot avoid a proposition which is a threat through the execution of an action, then the agent does not execute that action:
  $\mathsf{Fear}(\psi \to \neg\varphi) \wedge \mathsf{Cannot}(a, \neg\psi \vee (\psi \wedge \varphi)) \to \neg\langle a \rangle\top$

- if the execution of an action is a threat to the agent then the agent will not execute it and will not intend to execute it until it not causes fear:
  $\mathsf{Fear}(\langle a \rangle \neg\varphi) \to \neg\langle a \rangle\top \wedge \mathsf{AXA}(\neg\mathsf{Int}(\langle a \rangle\top)\mathsf{U}\neg\mathsf{Fear}(\langle a \rangle\varphi))$

- if the lack of effective resources to execute an action a is a threat, because the agent will not be able to achieve a fundamental desire, then the agent behaves in one of the following ways:
  - if the agent can execute another action for achieving the fundamental desire φ, then he executes that action:
    $\mathsf{Fear}(\neg\mathsf{EffCap}(a) \to \neg\varphi) \wedge \mathsf{Can}(b, \varphi) \to \langle b \rangle(\mathsf{Bel}(\varphi) \wedge \neg\mathsf{Fear}(\neg\varphi))$
  - if the agent believes he can get the resources to execute a he will get those resources and then execute a:
    $\mathsf{Fear}(\neg\mathsf{EffCap}(a) \to \neg\varphi) \wedge \mathsf{Bel}(\langle \mathtt{GET}(a) \rangle\varphi) \to \langle \mathtt{GET}(a); a \rangle(\mathsf{Bel}(\varphi) \wedge \neg\mathsf{Fear}(\neg\varphi))$

1.4.2 Anxiety

The activation of anxiety occurs when the desires of the agent can be at risk in the future. Therefore, anxiety works as preventive alert system towards future situations which may compromise the overall performance of the agent. We consider the following kinds of anxiety activation:

- $\mathsf{Anx}(\mathsf{EF}\neg\varphi) \equiv \mathsf{NotFavourable}(\varphi)$
- $\mathsf{Anx}(\psi \to \mathsf{EF}\neg\varphi) \equiv \mathsf{NotFavourableProp}(\psi, \varphi)$

- $\mathsf{Anx}(\langle\alpha\rangle\mathsf{EF}\neg\varphi) \equiv \mathsf{NotFavourableAct}(\alpha,\varphi)$
- $\mathsf{Anx}(\neg\mathsf{EffCap}(\alpha) \rightarrow \mathsf{EF}\neg\varphi) \equiv \mathsf{NotFavourableEffC}(\alpha,\varphi)$

The effects of anxiety are mainly preparing the agent to face future risk conditions, or to avoid them before they occur. We consider the following cases:

- if an action α guarantees that the desire will not be at risk, the agent intends to execute α. If he does not have enough resources, he will save them:
  $\mathsf{Anx}(\mathsf{EF}\neg\varphi) \wedge \mathsf{Can}(\alpha,\mathsf{AF}\varphi) \rightarrow \mathsf{Int}(\langle\alpha\rangle\mathsf{AF}\varphi)$
  $\mathsf{Anx}(\mathsf{EF}\neg\varphi) \wedge \mathsf{Int}(\langle\alpha\rangle\mathsf{AF}\varphi) \wedge \neg\mathsf{Res}(\alpha) \rightarrow \langle\mathtt{Save}(\alpha)\rangle\mathsf{Int}(\langle\alpha\rangle\mathsf{AF}\varphi)$

- if a proposition causes anxiety and the agent has a way to either negate that proposition or make that proposition coexist with the desire possibly at risk, then the agent will execute that action:
  $\mathsf{Anx}(\psi \rightarrow \mathsf{EF}\neg\varphi) \wedge \mathsf{Can}(\alpha,\mathsf{AF}(\neg\psi \vee (\psi \wedge \varphi)) \rightarrow \mathsf{Int}(\langle\alpha\rangle\mathsf{AF}(\neg\psi \vee (\psi \wedge \varphi)))$

- if the execution of an action is causing anxiety and the execution of that action is an intention of the agent, the agent will not intend it until it becomes harmful:
  $\mathsf{Anx}(\langle\alpha\rangle\mathsf{EF}\neg\varphi) \wedge \mathsf{Int}(\langle\alpha\rangle\varphi) \rightarrow \mathsf{AX}(\mathsf{A}(\neg\mathsf{Int}(\langle\alpha\rangle\varphi)\mathsf{U}\mathsf{Bel}(\mathsf{AF}\varphi)))$

- if the lack of effective capabilities is causing anxiety, the agent will intend to get the effective capabilities until they become available:
  $\mathsf{Anx}(\neg\mathsf{EffCap}(\alpha) \rightarrow \mathsf{EF}\neg\varphi) \rightarrow \mathsf{A}(\mathsf{Int}(\langle\mathtt{Get}(\alpha);\alpha\rangle\mathsf{AF}\varphi)\mathsf{U}\mathsf{Bel}(\mathsf{Available}(\alpha)))$

1.4.3 Self-confidence

Self-confidence represents the well-being of the agent relatively to the future achievement of one of its desires. Using other words, if a desire is in a favourable condition to be achieved, the agent feels self-confidence about its achievement. We consider the following kinds of self-confidence:

- $\mathsf{SConf}(\varphi) \equiv \mathsf{Favourable}(\varphi)$
- $\mathsf{Sconf}(\psi \rightarrow \varphi) \equiv \mathsf{FavourableProp}(\psi,\varphi)$
- $\mathsf{SConf}(\langle\alpha\rangle\varphi) \equiv \mathsf{FavourableAct}(\alpha,\varphi)$

Self-confidence deals mostly with the maintainance of intentions. Since the desires are considered to be achievable, the agent only cares about maintaining them in the set of intentions until he believes he achieved them. We consider the following kinds of behaviour:

- if the agent already intends a desire to which he is self-confident about, the agent will continue to intend it until he believes it is achieved:
  $\mathsf{SConf}(\varphi) \wedge \mathsf{Int}(\langle\alpha\rangle\varphi) \rightarrow \mathsf{A}(\mathsf{Int}(\langle\alpha\rangle\varphi)\mathsf{U}\mathsf{Bel}(\varphi))$

- if the agent still does not intend the desire, he will begin to intend it from the next situation on:
  $\mathsf{SConf}(\varphi) \wedge \mathsf{Can}(\alpha,\varphi) \wedge \neg\mathsf{Int}(\langle\alpha\rangle\varphi) \rightarrow \mathsf{AXInt}(\langle\alpha\rangle\varphi)$

- if a proposition causes self-confidence about a desire, then the agent will start intending that proposition and also intend both the proposition and the desire itself:
  $\mathsf{SConf}(\psi \rightarrow \varphi) \wedge \mathsf{Can}(\alpha,\psi) \wedge \neg\mathsf{Int}(\langle\alpha\rangle\psi) \rightarrow \mathsf{AXInt}(\langle\alpha\rangle\varphi)$
  $\mathsf{SConf}(\psi \rightarrow \varphi) \rightarrow \mathsf{Int}(\psi \wedge \varphi)$

- if the agent has the resources needed to execute an action which will guarantee the achievement of a desire to which it is self-confident about, then the agent will free those resources and intend to get them right before executing the action:
$\mathsf{SConf}(\langle\alpha\rangle\varphi) \wedge \mathsf{Int}(\langle\alpha\rangle\varphi) \wedge \mathsf{Saved}(\alpha) \rightarrow \langle\mathtt{Free}(\alpha)\rangle\mathsf{Int}(\langle\mathtt{Get}(\alpha);\alpha\rangle\varphi)$

1.5 Conclusions and future work

In this paper we have presented an improved version of the $\mathcal{E}_{\mathsf{BDI}}$ logic to model the activation and effects of emotions in the behaviour exhibited by a Emotional-BDI agent. The emotions analysed were fear, anxiety and self-confidence. This formalisation was based in the $\mathsf{BDI}_{\mathsf{CTL}}$ logic, which was extended with the notions of fundamental desire, explicit reference to actions, capabilities and resources.

We have shown that the satisfiability of $\mathcal{E}_{\mathsf{BDI}}$-formulae can be reduced to the satisfiability of $\mathsf{BDI}_{\mathsf{CTL}}$-formulae. We have implemented an extended version of the $\mathsf{BDI}_{\mathsf{CTL}}$'s tableau decision procedure for $\mathcal{E}_{\mathsf{BDI}}$-formulae.

We plan to obtain a direct characterisation of a sound and complete axiomatic system for $\mathcal{E}_{\mathsf{BDI}}$ that would allow the development of deduction systems suitable for automatic theorem proving, along the lines of the work of Naoyuki & Takata [NT02].

Schild [Sch00] has shown that $\mathsf{BDI}_{\mathsf{CTL}^*}$ can be formalised in the μ-calculus [Koz83] and thus complete axiomatizable. Therefore, it is an interesting topic of research to see how it is possible to develop an automated proof system for $\mathsf{BDI}_{\mathsf{CTL}^*}$ using this approach and extend it to $\mathcal{E}_{\mathsf{BDI}}$.

1.6 Acknowledgements

This work was partially founded by Fundação para a Ciência e Tecnologia (FCT) and program POSI.

Bibliography

[BIP88] Michael E. Bratman, David Israel, and Martha E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.

[Eme90] E. Allen Emerson. Temporal and modal logic. In *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, pages 995–1072. 1990.

[HKT00] David Harel, Dexter Kozen, and Jerzy Tiuryn. *Dynamic Logic*. MIT Press, 2000.

[HM92] Joseph Y. Halpern and Yoram Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artif. Intell.*, 54(3):319–379, 1992.

[Koz83] Dexter Kozen. Results on the propositional mu-calculus. *Theor. Comput. Sci.*, 27:333–354, 1983.

[NT02] Naoyuki Nide and Shiro Takata. Deduction systems for bdi logics using sequent calculus. In *The First International Joint Conference on Autonomous Agents & Multiagent Systems, AAMAS 2002, July 15-19, 2002, Bologna, Italy, Proceedings*, pages 928–935. ACM, 2002.

[OS02] Eugenio Oliveira and Luís Sarmento. Emotional valence-based mechanisms and agent personality. In Guilherme Bittencourt and Geber Ramalho, editors, *SBIA*, volume 2507 of *Lecture Notes in Computer Science*, pages 152–162. Springer, 2002.

[OS03] Eugenio Oliveira and Luís Sarmento. Emotional advantage for adaptability and autonomy. In *AAMAS*, pages 305–312, 2003.

[POM06] D. Pereira, E. Oliveira, and N. Moreira. Modelling emotional BDI agents. Riva Del Garda, Italy, August 2006. Workshop on Formal Approaches to Multi-Agent Systems (FAMAS 2006).

[POMS05] D. Pereira, E. Oliveira, N. Moreira, and L. Sarmento. Towards an architecture for emotional BDI agents. In *EPIA05 – 12th Portuguese Conference on Artificial Intelligence*, pages 40–46. Universidade da Beira Interior, IEEE, December 2005. ISBN 0-7803-9365-1.

[RG91] Anand S. Rao and Michael P. Georgeff. Modeling rational agents within a BDI-architecture. In James Allen, Richard Fikes, and Erik Sandewall, editors, *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, pages 473–484. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA, 1991.

[RG98] Anand S. Rao and Michael P. Georgeff. Decision procedures for bdi logics. *J. Log. Comput.*, 8(3):293–342, 1998.

[Sch00] Klaus Schild. On the relationship between bdi logics and standard logics of concurrency. *Autonomous Agents and Multi-Agent Systems*, 3(3):259–283, 2000.

[SMO04] Luís Sarmento, Daniel Moura, and Eugénio Oliveira. Fighting fire with fear. Proceedings of 2nd European Workshop on Multi-Agent Systems (EUMAS 2004), December 2004.

[STH04] Renate A. Schmidt, Dmitry Tishkovsky, and Ullrich Hustadt. Interactions between knowledge, action and commitment within agent dynamic logic. *Studia Logica*, 78(3):381–415, 2004.

[vdHvLM94] Wiebe van der Hoek, Bernd van Linder, and John-Jules Ch. Meyer. A logic of capabilities. In Anil Nerode and Yuri Matiyasevich, editors, *LFCS*, volume 813 of *Lecture Notes in Computer Science*, pages 366–378. Springer, 1994.

[vdHvLM00] Wiebe van der Hoek, Bernd van Linder, and John-Jules Ch. Meyer. On agents that have the ability to choose. *Studia Logica*, 66(1):79–119, 2000.

[vLvdHM98] Bernd van Linder, Wiebe van der Hoek, and John-Jules Ch. Meyer. Formalising abilities and opportunities of agents. *Fundamenta Informaticae*, 34(1-2):53–101, 1998.

[Whi03] M. Whitsey. Logical omniscience: A survey, 2003.