ECAI 2008
M. Ghallab et al. (Eds.)
IOS Press, 2008

doi:10.3233/978-1-58603-891-5-256

A Formal Model of Emotions: Integrating Qualitative and Quantitative Aspects

Bas R. Steunebrink and **Mehdi Dastani** and **John-Jules Ch. Meyer**[1]

Abstract. When constructing a formal model of emotions for intelligent agents, two types of aspects have to be taken into account. First, qualitative aspects pertain to the conditions that elicit emotions. Second, quantitative aspects pertain to the actual experience and intensity of elicited emotions. In this paper, we show how the qualitative aspects of a well-known psychological model of human emotions can be formalized in an agent specification language and how its quantitative aspects can be integrated into this model. Furthermore, we discuss several unspecified details and implicit assumptions in the psychological model that are explicated by this effort.

1 INTRODUCTION

Psychological models of emotions are currently being studied for their applicability in intelligent agents. These models of emotions can help in solving nondeterminism in an individual agent's decision making, they can be useful as a coordination mechanism in multi-agent systems, and they can make artificial agents more believable for human users (both in the actions that they select and the affective expressions they show based on experienced emotions). However, psychological models do not always take formalization into account, and those that do still leave a lot of details unspecified.

We will look specifically at the model of Ortony, Clore & Collins [4] and discuss the aspects of their model of emotions that are left open to interpretation when formalizing it. The "OCC model" describes very concisely what are the conditions that elicit an emotion, but upon formalization, the concepts used in these descriptions need to be translated to notions used in the chosen agent specification language. Doing so results in a qualitative formalization of emotion 'triggers.' However, with respect to quantitative aspects of emotions, which pertain to the actual experience and intensity of them, the OCC model only describes the types of quantities they distinguish and the factors that influence these quantities. Thus details on how these quantities should be calculated are not given.

In this paper, we will present part of a qualitative formalization of the OCC model, highlighting our interpretation choices and showing why we think these are reasonable. Furthermore, we will present how quantitative aspect of the OCC model can be integrated into this formalization and propose a reasonable way of calculating these quantities. Related work on computational models of emotions includes EMA [1], CogAff [6], and the work of Picard [5].

Outline: In section 2 we give an overview of the OCC model and define the agent specification language used for formalization. We present in section 3 a qualitative formalization of one emotion from the OCC model, and in section 4 the integration of quantitative aspects into the specification language. Section 5 contains a discussion on implicit assumptions explicated by the presented formalization.

[1] Utrecht University, The Netherlands. {bass,mehdi,jj}@cs.uu.nl

2 LANGUAGE AND SEMANTICS

The OCC model describes a hierarchy that classifies 22 emotions. The hierarchy contains three branches, namely emotions concerning aspects of objects (e.g., love and hate), actions of agents (e.g., pride and admiration), and consequences of events (e.g., joy and pity). Additionally, some branches combine to form a group of compound emotions, namely emotions concerning consequences of events *caused* by actions of agents (e.g., gratitude and anger). Because the objects of all these emotions (i.e. objects, actions, and events) correspond to notions commonly used in agent models (i.e. agents, plans, and goal accomplishments, respectively), this makes the OCC model suitable for use in the deliberation and practical reasoning of artificial agents. It should be emphasized that emotions are not used to describe the entire cognitive state of an agent, but emotions are always relative to individual objects, actions, and events.

The OCC model defines both qualitative and quantitative aspects of emotions. Qualitatively, it defines the conditions that *elicit* an emotion; quantitatively, it describes how a potential, threshold, and intensity are associated with each elicited emotion and what are the variables affecting these quantities. For example, the compound emotion *gratitude* is qualitatively specified as "approving of someone else's praiseworthy action and being pleased about the related desirable event." The variables affecting its (quantitative) intensity are 1) the judged praiseworthiness of the action, 2) the unexpectedness of the event, and 3) the desirability of the event.

We use KARO [2, 3] as a framework for the formalization of the 22 emotions of the OCC model. The KARO framework is a mixture of dynamic logic, epistemic / doxastic logic, and several additional (modal) operators for dealing with the motivational aspects of artificial agents. We present a modest modification of the KARO framework, so that the eliciting conditions of the emotions of the OCC model can be appropriately translated and modeled. Below we explain how 'OCC ingredients' are translated into 'KARO ingredients.' When formalizing the branch (of the OCC hierarchy) of emotions concerning consequences of events, we will translate OCC's notion of an event as the accomplishment or undermining of a goal (or part thereof). For goal-directed agents, such goal-related events are useful for determining how well plans are progressing. For example, replanning may be triggered when fear for failure of a plan to reach a goal is greater than hope for accomplishment of the goal [7]. When formalizing the branch (of the OCC hierarchy) of emotions concerning actions of agents, we will translate OCC's notion of actions as plans consisting of domain actions and sequential compositions of actions.

We now go into the formal details of the agent specification language that we use to formalize the OCC model. The KARO framework is designed to specify goal-directed agents. However, in contrast to KARO, we do not allow arbitrary formulas as goals; instead,

we define a (declarative) goal as a conjunction of literals, where each literal represents a subgoal. This is because we want to be able to break up goals into the part that has already been accomplished and the remaining part. Furthermore, we require goals to be (logically) consistent and nonempty, so they are drawn from the set $\mathcal{K}'$ below.

Definition 1. *(Consistent conjunctions). Let P be a set of atomic propositions, $Lits = P \cup \{\, \neg p \mid p \in P \,\}$ be the set of literals, and $\bigwedge \emptyset = \top$ (verum). Then $\mathcal{K}$ is the set of all consistent conjunctions of literals, and $\mathcal{K}'$ does not contain the empty conjunction:*

$$\mathcal{K} = \{\, \textstyle\bigwedge \Phi \mid \Phi \subseteq Lits,\ \Phi \not\models_{\mathrm{CL}} \bot \,\}, \qquad \mathcal{K}' = \mathcal{K} \setminus \{\top\} \tag{1}$$

where CL *stands for Classical Logic (so Φ is consistent).*

In the following we assume the existence of a set $\mathcal{A}$ of atomic actions and a set $Plans$ consisting of all actions and sequential compositions of actions, i.e., $Plans$ is the smallest set such that $\mathcal{A} \subseteq Plans$ and if $\alpha \in \mathcal{A}$ and $\pi \in Plans$ then $(\alpha; \pi) \in Plans$.

We define *emotion triggering fluents* to represent each of the 22 emotion types of the OCC model. The emotions are outlined below such that each row contains two emotions that are defined by OCC to be each other's opposites, with the positive (for agent i) emotions on the left and the negative ones on the right. It should be noted that an agent is allowed to have 'mixed feelings,' i.e. opposing emotions can be triggered simultaneously. However, our model ensures that the objects of opposing emotions are distinct (e.g., an agent can experience both gratification and remorse in response to some event, but the objects of these two emotions will concern different parts of the event).

Definition 2. *(Emotion triggering fluents). Let $\mathcal{G}$ be a set of agent names. $EmoTriggers = \{\, \epsilon \in Em(i) \mid i \in \mathcal{G} \,\}$ is the set of all emotion triggering fluents, where*

$$\begin{aligned} Em(i) = \{\ & \mathbf{gratification}_i(\alpha,\kappa), && \mathbf{remorse}_i(\alpha,\kappa), \\ & \mathbf{gratitude}_i(j,\alpha,\kappa), && \mathbf{anger}_i(j,\alpha,\kappa), \\ & \mathbf{pride}_i(\alpha), && \mathbf{shame}_i(\alpha), \\ & \mathbf{admiration}_i(j,\alpha), && \mathbf{reproach}_i(j,\alpha), \\ & \mathbf{joy}_i(\kappa), && \mathbf{distress}_i(\kappa), \\ & \mathbf{happy\text{-}for}_i(j,\kappa), && \mathbf{resentment}_i(j,\kappa), \\ & \mathbf{gloating}_i(j,\kappa), && \mathbf{pity}_i(j,\kappa), \\ & \mathbf{hope}_i(\pi,\kappa), && \mathbf{fear}_i(\pi,\neg\kappa), \\ & \mathbf{satisfaction}_i(\pi,\kappa), && \mathbf{disappointment}_i(\pi,\kappa), \\ & \mathbf{relief}_i(\pi,\neg\kappa), && \mathbf{fears\text{-}confirmed}_i(\pi,\neg\kappa), \\ & \mathbf{love}_i(j), && \mathbf{hate}_i(j) \\ & \mid j \in \mathcal{G},\ i \neq j,\ \alpha \in \mathcal{A},\ \pi \in Plans,\ \kappa \in \mathcal{K}' \,\}. \end{aligned} \tag{2}$$

The informal reading of $\mathbf{gratification}_i(\alpha,\kappa)$ is: agent i has performed action α accomplishing (sub)goal(s) κ eliciting gratification. Due to space limitations we can take only this emotion as example.

Definition 3. *(Agent specification language). Let the sets P, $\mathcal{K}'$, $Plans$, $\mathcal{G}$, and $EmoTriggers$ be defined as above. The agent specification language $\mathcal{L}_{PAG}$ is the smallest set closed under:*

- $P \cup EmoTriggers \subseteq \mathcal{L}_{PAG}$.
- *If* $\varphi_1, \varphi_2 \in \mathcal{L}_{PAG}$ *then* $\neg\varphi_1, (\varphi_1 \wedge \varphi_2), (\varphi_1 \rightarrow \varphi_2) \in \mathcal{L}_{PAG}$.
- *If* $\varphi \in \mathcal{L}_{PAG}$ *and* $i \in \mathcal{G}$ *then* $\mathbf{B}_i\varphi \in \mathcal{L}_{PAG}$.
- *If* $\kappa, \kappa' \in \mathcal{K}$ *and* $i \in \mathcal{G}$ *then* $\mathbf{G}_i\kappa, \mathbf{Acc}_i(\kappa,\kappa') \in \mathcal{L}_{PAG}$.
- *If* $\kappa_1, \kappa_2, \kappa_3 \in \mathcal{K}$ *then* $\mathbf{Diff}(\kappa_1,\kappa_2,\kappa_3) \in \mathcal{L}_{PAG}$.
- *If* $\pi \in Plans$, $\varphi \in \mathcal{L}_{PAG}$, *and* $i \in \mathcal{G}$ *then* $[\mathbf{do}_i(\pi)]\varphi \in \mathcal{L}_{PAG}$.

$\mathbf{B}_i\varphi$ means agent i believes in φ; $\mathbf{G}_i\kappa$ means agent i has the (declarative) goal to accomplish κ; $\mathbf{Acc}_i(\kappa_1,\kappa_2)$ means agent i believes that the part of κ_1 that has been accomplished is κ_2; $\mathbf{Diff}(\kappa_1,\kappa_2,\kappa_3)$ means κ_3 is the difference between κ_1 and κ_2; $[\mathbf{do}_i(\pi)]\varphi$ means φ holds after agent i has performed π.

With respect to the semantics of $\mathcal{L}_{PAG}$, we model the belief and action operators in a standard way using Kripke semantics, while using sets for goals, accomplishments, and emotional fluents. We use system KD45 for belief models, which have the form $\mathrm{M} = \langle S, R_{\mathbf{B}}, \vartheta \rangle$. We denote the class of belief models as $\mathcal{M}$. With slight abuse of notation, we define a selector for the set of states S of model M as: $states_{\mathrm{M}} = S$. The semantics of actions are defined *over* the Kripke models of belief, as actions may change the mental state of an agent. They have the form $\mathfrak{M} = \langle \Sigma, R_{\mathcal{A}}, \mathrm{Emo}, \mathrm{Aux} \rangle$, where $\Sigma = \{\, (\mathrm{M}, s) \mid \mathrm{M} \in \mathcal{M},\ s \in states_{\mathrm{M}} \,\}$ and $R_{\mathcal{A}}$ is an accessibility relation on Σ. $\mathrm{Emo} = \{\, Gratification, \ldots, Hate \}$ is a set of 22 functions designed to define the semantics of the emotion triggering fluents such as **gratification**. These functions are defined per agent ($\mathcal{G}$) and model–state pair (Σ) and their mappings can be derived directly from Definition 2, e.g., $Gratification : \mathcal{G} \times \Sigma \rightarrow \wp(\mathcal{A} \times \mathcal{K}')$. $\mathrm{Aux} = \langle \Gamma, \mathrm{H}, Acc, Diff \rangle$ is a structure of auxiliary functions, where $\Gamma : \mathcal{G} \times \Sigma \rightarrow \wp(\mathcal{K}')$ is a function returning the set of goals an agent has per model–state pair; the history (H), accomplishment (Acc), and difference ($Diff$) functions are explained below.

A history of the multi-agent system needs to be kept to formalize emotions concerning states visited by the multi-agent system in the past. To this end, a history function H is defined, mapping model–state pairs to tuples recording the model–state pairs *actually* visited by the multi-agent system, the actions performed in these states, and the agents that performed these actions. A history is formally denoted as $\mathrm{H}(\mathrm{M}, s) = \langle (\mathrm{M}_{-1}, s_{-1}, i_{-1}, \alpha_{-1}), (\mathrm{M}_{-2}, s_{-2}, i_{-2}, \alpha_{-2}), \ldots, (\mathrm{M}_{-n}, s_{-n}, i_{-n}, \alpha_{-n}) \rangle$, i.e., a mapping of model–state pairs to a sequence of model–state–agent–action tuples $(\mathrm{M}_{-k}, s_{-k}, i_{-k}, \alpha_{-k})$. Such a tuple in $\mathrm{H}(\mathrm{M}, s)$ thus specifies that the multi-agent system, currently in state s of model M, was once in state s_{-k} of model M_{-k} and left that state because agent i_{-k} performed action α_{-k}. Furthermore, H should satisfy two constraints such that it is well-behaved through the past it records[2] and with future actions[3].

The gratification emotion considered here is defined with respect to actions and goals of an agent (why this is so will become clear in the next section). However, for determining whether a positive event-related emotion should be triggered, it must be possible to determine which parts (i.e. subgoals) of a goal have been accomplished by an action. Moreover, such an ability would allow us to compare the accomplished parts of a goal at subsequent states, so that we can talk about actions accomplishing new subgoals. To this end, an accomplishment function Acc is defined as follows. If κ is a consistent conjunction of literals (i.e. $\kappa \in \mathcal{K}$) and $(\kappa, \kappa') \in Acc(i)(\mathrm{M}, s)$, then $\kappa' \in \mathcal{K}$ is a conjunction containing exactly those literals from κ that agent i believes to be true in state s of model M. More formally,

$$\begin{aligned} Acc(i)(\mathrm{M}, s) = \{\, & (\textstyle\bigwedge(\Phi_1 \cup \Phi_2), \bigwedge \Phi_1) \mid \Phi_1, \Phi_2 \subseteq Lits,\ \Phi_1 \cup \Phi_2 \not\models_{\mathrm{CL}} \bot, \\ & \forall\ p \in \Phi_1 : \forall s' \in R_{\mathbf{B}}(i)(s) : p \in \vartheta(s'), \\ & \forall \neg p \in \Phi_1 : \forall s' \in R_{\mathbf{B}}(i)(s) : p \notin \vartheta(s'), \\ & \forall\ p \in \Phi_2 : \exists s' \in R_{\mathbf{B}}(i)(s) : p \notin \vartheta(s'), \\ & \forall \neg p \in \Phi_2 : \exists s' \in R_{\mathbf{B}}(i)(s) : p \in \vartheta(s') \,\} \end{aligned} \tag{3}$$

where $\mathrm{M} = \langle S, R_{\mathbf{B}}, \vartheta \rangle$. Note that by construction, Φ_1 and Φ_2 are mutually exclusive. Because for all $\kappa \in \mathcal{K}$ there exists exactly one $\kappa' \in \mathcal{K}$ such that $(\kappa, \kappa') \in Acc(i)(\mathrm{M}, s)$ (for arbitrary agent i and model–state pair M, s), we will write $Acc(i)(\mathrm{M}, s)(\kappa) = \kappa'$, treating Acc as a *function* determining the accomplished part of a conjunction of literals. To determine the part of a goal that an agent has *not* yet accomplished, we define the convenience function $Diff$. $Diff(\kappa, \kappa')$ returns the part of the conjunction κ that does not appear

[2] That is, $\mathrm{H}(\mathrm{M}_{-1}, s_{-1}) = \langle (\mathrm{M}_{-2}, s_{-2}, i_{-2}, \alpha_{-2}), \ldots, (\mathrm{M}_{-n}, s_{-n}, i_{-n}, \alpha_{-n}) \rangle$ if $\mathrm{H}(\mathrm{M}, s)$ is as above.

[3] That is, $\mathrm{H}(\mathrm{M}', s') = \langle (\mathrm{M}, s, i, \alpha), (\mathrm{M}_{-1}, s_{-1}, i_{-1}, \alpha_{-1}), \ldots, (\mathrm{M}_{-n}, s_{-n}, i_{-n}, \alpha_{-n}) \rangle$ for all $(\mathrm{M}', s') \in R_{\mathcal{A}}(i, \alpha)(\mathrm{M}, s)$.

in κ'. So we write $\mathit{Diff}(\kappa, \kappa') = \kappa''$, which should be read as "the difference between κ and κ' is κ''." *Diff* is defined as a set of triples:

$$\mathit{Diff} = \{\, (\bigwedge \Phi_1, \bigwedge \Phi_2, \bigwedge(\Phi_1 \setminus \Phi_2)) \mid \Phi_1, \Phi_2 \subseteq \mathit{Lits} \,\} \quad (4)$$

in which the first two elements of each triple determine the third element. So *Diff* can also be regarded as a function, taking two arguments $\kappa, \kappa' \in \mathcal{K}$ and computing their difference $\kappa'' \in \mathcal{K}$. Note that for convenience, its syntactic counterpart **Diff**, as shown below in Definition 4, requires that the difference κ'' is not empty (i.e. $\kappa'' \neq \top$), because this is needed in all instances where **Diff** is used.

Definition 4. *(Interpretation of formulas). Let* $\mathrm{M} = \langle S, R_{\mathbf{B}}, \vartheta \rangle$ *and* $\mathfrak{M} = \langle \Sigma, R_{\mathcal{A}}, \mathrm{Emo}, \mathrm{Aux} \rangle$ *be structures defined as above. Formulas in language* $\mathcal{L}_{PAG}$ *are interpreted in model–state pairs as follows:*

$$\begin{aligned}
&\mathrm{M}, s \models p && \Leftrightarrow p \in \vartheta(s) \quad \textit{for } p \in P\\
&\mathrm{M}, s \models \neg\varphi && \Leftrightarrow \mathrm{M}, s \not\models \varphi\\
&\mathrm{M}, s \models \varphi_1 \wedge \varphi_2 && \Leftrightarrow \mathrm{M}, s \models \varphi_1 \;\&\; \mathrm{M}, s \models \varphi_2\\
&\mathrm{M}, s \models \mathbf{B}_i\varphi && \Leftrightarrow \forall s' \in R_{\mathbf{B}}(i)(s) : \mathrm{M}, s' \models \varphi\\
&\mathrm{M}, s \models \mathbf{G}_i\kappa && \Leftrightarrow \kappa \in \Gamma(i)(\mathrm{M}, s)\\
&\mathrm{M}, s \models \mathbf{Acc}_i(\kappa, \kappa') && \Leftrightarrow \mathit{Acc}(i)(\mathrm{M}, s)(\kappa) = \kappa'\\
&\mathrm{M}, s \models \mathbf{Diff}(\kappa, \kappa', \kappa'') && \Leftrightarrow \mathit{Diff}(\kappa, \kappa') = \kappa'' \;\&\; \kappa'' \neq \top\\
&\mathrm{M}, s \models [\mathbf{do}_i(\pi)]\varphi && \Leftrightarrow \forall (\mathrm{M}', s') \in R_{\mathcal{A}}(i, \pi)(\mathrm{M}, s) : \mathrm{M}', s' \models \varphi\\
&\mathrm{M}, s \models \mathbf{gratification}_i(\alpha, \kappa) && \Leftrightarrow (\alpha, \kappa) \in \mathit{Gratification}(i)(\mathrm{M}, s)
\end{aligned}$$

Note that we evaluate formulas in state s of model M. The Kripke structure $\langle \Sigma, R_{\mathcal{A}} \rangle$ is then used for the interpretation of $[\mathbf{do}_i(\pi)]\varphi$ formulas. We express that some formula φ is a validity as $\models \varphi$.

3 FORMALIZING QUALITATIVE ASPECTS

The OCC model provides for each of the 22 emotion types a concise description of the conditions that elicit such an emotion. Below we show for gratification how we translate these descriptions to ingredients of the agent specification language introduced above. We have a complete formalization of all 22 emotion types of the OCC model, but because of space limitations we can present only one example.

According to OCC, *gratification is approving of one's own praiseworthy action and being pleased about the related desirable event.* So gratification for an agent i should be defined with respect to an action α and an event κ_{acc}. In our formalization, a desirable event is translated as the accomplishment of one or more subgoals by an action. To check whether this is the case in a state M, s, we first examine the history to verify that the last executed action was α by i, i.e. $\mathrm{H}(\mathrm{M}, s) = \langle (\mathrm{M}', s', i, \alpha), \ldots \rangle$, so the state of the multi-agent system before the execution of α was s' of model M'. Next we take a goal κ from i's goal base at M', s', i.e. $\kappa \in \Gamma(i)(\mathrm{M}', s')$, and determine which part of κ had already been accomplished by i, i.e. $\mathit{Acc}(i)(\mathrm{M}', s')(\kappa) = \kappa_{old}$ (so the 'old' accomplished part is called κ_{old}). In the current state M, s we also determine the part of κ that has been accomplished, i.e. $\mathit{Acc}(i)(\mathrm{M}, s)(\kappa) = \kappa_{new}$ (so the 'new' accomplished part is called κ_{new}). For α to be praiseworthy to i, we need to determine whether α has accomplished any additional subgoals of κ, i.e. $\mathit{Diff}(\kappa_{new}, \kappa_{old}) = \kappa_{acc}$ (so the conjunction of newly accomplished subgoals is called κ_{acc}). Also, κ must either still be in i's goal base, i.e. $\kappa \in \Gamma(i)(\mathrm{M}, s)$, or action α must have caused goal κ to have become accomplished in its entirety, i.e. $\kappa = \kappa_{new}$ (so the accomplished part of goal κ is κ itself). If κ_{acc} contains one or more subgoals, i.e. $\kappa_{acc} \neq \top$, then κ_{acc} will constitute a desirable event for i and gratification about having performed action α resulting in the accomplishment of the subgoals κ_{acc} will be triggered for agent i in state M, s. This explanation can be directly transcribed to the following definition of the semantic function *Gratification*:

$$\begin{aligned}
&\mathit{Gratification}(i)(\mathrm{M}, s) = \{\, (\alpha, \kappa_{acc}) \mid \mathrm{H}(\mathrm{M}, s) = \langle (\mathrm{M}', s', i, \alpha), \ldots \rangle,\\
&\kappa \in \Gamma(i)(\mathrm{M}', s'), \mathit{Acc}(i)(\mathrm{M}', s')(\kappa) = \kappa_{old}, \mathit{Acc}(i)(\mathrm{M}, s)(\kappa) = \kappa_{new},\\
&\kappa \in \Gamma(i)(\mathrm{M}, s) \text{ or } \kappa = \kappa_{new}, \mathit{Diff}(\kappa_{new}, \kappa_{old}) = \kappa_{acc}, \kappa_{acc} \neq \top \,\} \quad (5)
\end{aligned}$$

We can now derive the following propositions about gratification.

$$\begin{aligned}
\models \mathbf{G}\kappa \wedge \mathbf{Acc}(\kappa, \kappa_{old}) \rightarrow [\mathbf{do}(\alpha)](&\mathbf{G}\kappa \wedge \mathbf{Acc}(\kappa, \kappa_{new}) \quad (6)\\
&\wedge \mathbf{Diff}(\kappa_{new}, \kappa_{old}, \kappa_{acc}) \rightarrow \mathbf{gratification}(\alpha, \kappa_{acc}))
\end{aligned}$$

The intuitive reading of this proposition is as follows. Suppose an agent has goal κ of which it has already accomplished part κ_{old}. After performing some action α, it inspects its goal κ again and determines that the part it has accomplished is κ_{new}. Now if the difference between κ_{new} and κ_{old}, called κ_{acc}, is not empty, gratification is elicited with respect to action α and the accomplished subgoals κ_{acc}.

$$\begin{aligned}
\models \mathbf{G}\kappa \wedge \mathbf{Acc}(\kappa, \kappa_{old}) \rightarrow & \quad (7)\\
[\mathbf{do}(\alpha)](&\mathbf{B}\kappa \wedge \mathbf{Diff}(\kappa, \kappa_{old}, \kappa_{acc}) \rightarrow \mathbf{gratification}(\alpha, \kappa_{acc}))
\end{aligned}$$

In case the agent believes action α has accomplished all of the remaining subgoals of goal κ, $\mathbf{B}\kappa$ will hold after performing α. Of course in this situation gratification should also be elicited. Note that this proposition corresponds to formula (5) where $\kappa = \kappa_{new}$ holds in the condition "$\kappa \in \Gamma(i)(\mathrm{M}, s)$ or $\kappa = \kappa_{new}$", whereas Proposition (6) corresponds to the situation where $\kappa \in \Gamma(i)(\mathrm{M}, s)$ holds. Also note that it is possible in both Propositions (6) and (7) that $\kappa = \kappa_{acc}$; this is exactly the case if $\kappa_{old} = \top$, i.e. if none of the subgoals of κ had initially been accomplished yet.

4 INTEGRATING QUANTITATIVE ASPECTS

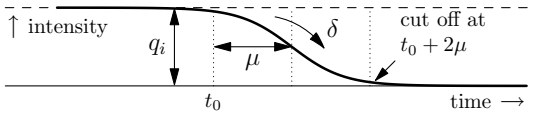
In the preceding section we have explained how the eliciting conditions of the emotions of the OCC model can be formalized in dynamic logic, creating a formal qualitative model of emotions. This model specifies precisely *when* emotions are triggered, but quantitative aspects responsible for the *experience* of emotions are missing.

In the OCC model, quantitative aspects of emotions are described in terms of *potentials*, *thresholds*, and *intensities*. For each of the 22 emotions, OCC provide a list of variables (e.g., desirability, praiseworthiness, effort) that affect the intensity of that emotion if its eliciting conditions hold. The idea is that the weighted sum of these variables equals the emotion's potential. The intensity of an emotion is defined as its potential minus its threshold, or zero if the threshold is greater than the potential. The values of thresholds of emotions are not specified by OCC, but they are hinted to depend on global variables indicating an agent's 'mood.' For example, if an agent is in a 'good mood,' the thresholds of the eleven negative emotions are increased, causing a lower (or zero) intensity to be associated with a negative emotion, if one is triggered. Emotions that are assigned a nonzero intensity may in turn influence the mood of an agent, entangling the dynamics of short-term emotions and a long-term mood.

Here we do not study the variables affecting intensities but instead focus on the integration of intensities into the qualitative formalization. Given the values of the potential and threshold of a triggered emotion, OCC indicate how the (initial) intensity value can be calculated (i.e. the maximum of zero and their difference), but not how this value changes over time. We expect the intensity of an emotion to gradually decrease after the time it was triggered, dropping to zero within a finite amount of time. We propose that a reasonable default choice would be an inverse sigmoid function, i.e., of the form $\frac{1}{1+e^x}$. Of course, there are several parameters that have to be set to give the inverse sigmoid function the shape desired for a particular emotion. Specifically, given the initial intensity q_i, the time at which the emotion was triggered t_0, the half-life time μ, and the fall-off speed δ, we define the intensity value as a function of time (x-axis below) as:

$$\mathit{int}(q_i, t_0, \mu, \delta)(x) = \frac{q_i}{1 + e^{(x - t_0 - \mu)\delta}} - c$$

where c is used to cut off the intensity for a large enough x (because an inverse sigmoid function only reaches zero in the limit, which we do not consider to be an intuitive property of emotion intensities).[4] This function and its parameters can be visualized as below:



Quantitative aspects of emotions can be modeled on top of the described qualitative model as follows. The satisfaction of an emotion triggering fluent in a certain state (e.g., $\mathrm{M}, s \models \mathbf{joy}_i(\kappa)$) is regarded as a *trigger* for associating a potential, threshold, and intensity with the emotional fluent and for calculating their quantities. For this purpose we define a function $newTrigEm$ returning for each model–state pair the set of 'new' emotions triggered in that model–state pair, i.e., all emotion triggering fluents satisfied in the model–state pair. Moreover, it associates with each such emotion the time it was triggered (t_0), its initial intensity (q_i), initial half-life (μ), and initial fall-off speed (δ). We define $newTrigEm$ for all $(\mathrm{M}, s) \in \Sigma$ as:

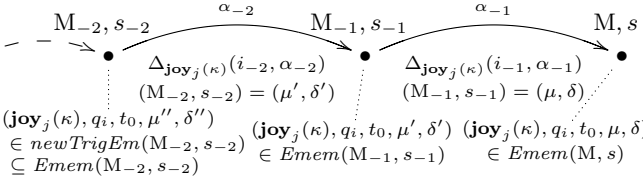
$$newTrigEm(\mathrm{M}, s) = \{\, (\epsilon, q_i, t_0, \mu, \delta) \mid \epsilon \in EmoTriggers,\ \mathrm{M}, s \models \epsilon,\ t_0 = T(\mathrm{M}, s),\ q_i = \max(0, pot(\mathrm{M}, s)(\epsilon) - thr(\mathrm{M}, s)(\epsilon)),\ \mu = \mu_0(\mathrm{M}, s)(\epsilon),\ \delta = \delta_0(\mathrm{M}, s)(\epsilon) \,\}$$

Note that $\mathrm{M}, s \models \epsilon$ represents the triggering condition and that it is assumed there exists an external clock function $T : \Sigma \to \mathbb{R}^+$. How exactly emotion potentials and thresholds are calculated by pot and thr is not the focus of this paper and will be a subject of future work. The function μ_0 for calculating the initial half-life may depend on q_i, e.g., the greater the initial intensity, the greater the half-life. A reasonable default choice for δ_0 may be to just return 1 to obtain the normal sigmoid curve. However, keeping the two inverse sigmoid function parameters (i.e. μ and δ) fixed may not be desirable. For example, one may want to prolong the duration of a high intensity of an anger emotion after another agent's execution of an action constituting profanity, which can be done by increasing μ. Each such change caused by an action can be encoded in a function Δ. Formally, the change to the intensity function parameters of emotion ϵ caused by agent i performing action α in state M, s is denoted as $\Delta_\epsilon(i, \alpha)(\mathrm{M}, s) = (\mu', \delta')$. These new parameters do not have to differ from the old ones. The choices of μ and δ are application-dependent; here we follow the level of abstraction of the OCC model.

Given $newTrigEm$ and Δ, it is a straightforward task to define a function $Emem$ (for 'emotion memory') which returns for each model–state pair all emotions that have been triggered, both in previous states and in the model–state pair. As with $newTrigEm$, each emotion triggering fluent in the set returned by $Emem$ is accompanied by four values as described above. For all $(\mathrm{M}, s) \in \Sigma$, $Emem(\mathrm{M}, s)$ is defined as follows.

$$Emem(\mathrm{M}, s) = newTrigEm(\mathrm{M}, s) \cup \{\, (\epsilon, q_i, t_0, \mu', \delta') \mid \mathrm{H}(\mathrm{M}, s) = \langle (\mathrm{M}_{-1}, s_{-1}, i_{-1}, \alpha_{-1}), \ldots \rangle,\ (\epsilon, q_i, t_0, \mu, \delta) \in Emem(\mathrm{M}_{-1}, s_{-1}),\ \Delta_\epsilon(i_{-1}, \alpha_{-1})(\mathrm{M}_{-1}, s_{-1}) = (\mu', \delta') \,\}$$

Note that the set comprehension denotes that all previously triggered emotions must be included, possibly with changes to μ and δ (which is prescribed by Δ), but not to q_i and t_0. By varying the inverse sigmoid function parameters, virtually any other kind of function can be simulated. Also note that if M, s is an initial state (i.e. $\mathrm{H}(\mathrm{M}, s) = \langle\,\rangle$), the set comprehension reduces to $\emptyset$. The figure below visualizes the dynamics of $Emem$ for an emotion $\mathbf{joy}_j(\kappa)$ being triggered in state M_{-2}, s_{-2} and having its intensity parameters adjusted by Δ at each action (these transitions are available in M, s as the history $\mathrm{H}(\mathrm{M}, s) = \langle (\mathrm{M}_{-1}, s_{-1}, i_{-1}, \alpha_{-1}), (\mathrm{M}_{-2}, s_{-2}, i_{-2}, \alpha_{-2}), \ldots \rangle$).



Having $Emem$ means that we know for each model–state pair exactly which emotions have been triggered and when, allowing for the investigation of properties relating to the *experience* of these emotions. For example, if the emotion *anger* is triggered and this emotion is assigned a positive intensity (i.e. strictly greater than zero), we say that the agent in question is *angry*. If we construct such sentences for all 22 emotion types of the OCC model, we see that most are represented by a noun (see formula 2), while one can think of a corresponding adjective for each emotion as well. Since the usage of emotion adjectives corresponds more naturally to the notion of *experiencing the emotion* in question, we define a second set of emotional fluents, called $EmoExp$ for "emotional experience fluents." Emotional experience fluents are denoted as $\hat{\epsilon}$, where ϵ is an emotion triggering fluent. Because of space limitations we do not present a complete definition of $EmoExp$ as in formula (2), but use, e.g., $\widehat{\mathbf{joy}}_i(\kappa)$ as the adjective form of $\mathbf{joy}_i(\kappa)$, which should be read as "agent i is joyous about having accomplished (sub)goal(s) κ." By convention, we write $\epsilon \in EmoTriggers$ and $\hat{\epsilon} \in EmoExp$. With slight abuse of notation, we also use the 'cap' to convert from $EmoTriggers$ to $EmoExp$, e.g., if $\epsilon = \mathbf{joy}_i(\kappa)$ then $\hat{\epsilon} = \mathbf{joyous}_i(\kappa)$. Now we can model the experience of an emotion as the satisfaction of an *emotional experience fluent*. Specifically, an emotional experience fluent $\hat{\epsilon} \in EmoExp$ is satisfied in a model–state pair M, s if and only if the intensity associated with it is greater than zero at the current time:

$$\mathrm{M}, s \models \hat{\epsilon} \;\Leftrightarrow\; \exists(\epsilon', q_i, t_0, \mu, \delta) \in Emem(\mathrm{M}, s) : \big[\hat{\epsilon} = \hat{\epsilon}' \;\&\; int(q_i, t_0, \mu, \delta)(T(\mathrm{M}, s)) > 0\big]$$

Note that if the potential of a newly triggered emotion is less than its threshold (i.e. for that emotion $q_i = 0$), the inequality above will reduce to $0 > 0$, so its emotional experience fluent is never satisfied.

In order to formulate a frame property for emotional experience fluents, we introduce several additional constructs. First, a plan π of agent i possibly affects the intensity function of an emotion $\hat{\epsilon}$, written as $\mathbf{affects}_i(\pi, \hat{\epsilon})$, if and only if there exists a state resulting from i performing π where the intensity parameters of $\hat{\epsilon}$ have changed:

$$\mathrm{M}, s \models \mathbf{affects}_i(\pi, \hat{\epsilon}) \;\Leftrightarrow\; \exists(\epsilon', q_i, t_0, \mu, \delta) \in Emem(\mathrm{M}, s) : \big[\hat{\epsilon} = \hat{\epsilon}' \;\&\; \exists \mu', \delta' \in \mathbb{R}, (\mathrm{M}', s') \in R_{\mathcal{A}}(i, \pi)(\mathrm{M}, s) : (\epsilon', q_i, t_0, \mu', \delta') \in Emem(\mathrm{M}', s') \;\&\; (\mu, \delta) \neq (\mu', \delta')\big]$$

Note that no relation whatsoever is assumed between i (i.e. the possible performer of π) and the agent subscript of $\hat{\epsilon}$ (i.e. the agent experiencing $\hat{\epsilon}$). Furthermore, we specify that the remaining duration of an emotion $\hat{\epsilon}$, given the current parameters, is greater than d if and only if its intensity is greater than zero at the current time plus d:

$$\mathrm{M}, s \models \mathbf{duration}(\hat{\epsilon}) > d \;\Leftrightarrow\; \exists(\epsilon', q_i, t_0, \mu, \delta) \in Emem(\mathrm{M}, s) : \big[\hat{\epsilon} = \hat{\epsilon}' \;\&\; int(q_i, t_0, \mu, \delta)(T(\mathrm{M}, s) + d) > 0\big]$$

Finally, we extend the dynamic operator with a subscript duration d, with the interpretation that the formula following the dynamic operator holds if the execution of the action took time less than d:

[4] We should note to the interested reader that the intensity function we actually envisage is $int(x) = \max(0, \frac{\eta q_i}{1+e^{(x-t_0-\mu)\delta}} - \frac{\eta q_i}{1+e^{\mu\delta}})$ where $\eta = (\frac{1}{1+e^{-\mu\delta}} - \frac{1}{1+e^{\mu\delta}})^{-1}$, having the following properties: $int(t_0) = q_i$, $int(t_0 + \mu) = \frac{1}{2} q_i$, and $int(x) = 0$ for all $x \geq t_0 + 2\mu$.

$$\mathrm{M}, s \models [\mathbf{do}_i(\pi)]_d\varphi \;\Leftrightarrow\; \forall(\mathrm{M}', s') \in R_{\mathcal{A}}(i,\pi)(\mathrm{M}, s) : \big[T(\mathrm{M}', s') - T(\mathrm{M}, s) \leq d \Rightarrow \mathrm{M}', s' \models \varphi\big]$$

Given the constructs introduced above, we can form a frame property stating that if an action does not possibly affect the intensity parameters of a currently experienced emotion, then for any time span within which the emotion's intensity remains positive, the emotion will still be experienced if the action finishes within the time span:

$$\models \hat{\epsilon} \wedge \neg\mathbf{affects}_j(\pi, \hat{\epsilon}) \wedge \mathbf{duration}(\hat{\epsilon}) > d \rightarrow [\mathbf{do}_j(\pi)]_d\hat{\epsilon} \quad (8)$$

A proof of this proposition is omitted due to space limitations. As is to be expected, propositions relating any $\epsilon \in \mathit{EmoTriggers}$ with a corresponding $\hat{\epsilon} \in \mathit{EmoExp}$ cannot be made; that is, we have

$$\not\models \epsilon \rightarrow \hat{\epsilon} \qquad \text{and} \qquad \not\models \hat{\epsilon} \rightarrow \epsilon. \quad (9)$$

Note that here we mean propositions of the form, e.g., $\not\models \mathbf{joy}_i(\kappa) \rightarrow \mathbf{joyous}_i(\kappa)$ and $\not\models \mathbf{joyous}_i(\kappa) \rightarrow \mathbf{joy}_i(\kappa)$. Intuitively, $\not\models \epsilon \rightarrow \hat{\epsilon}$ states that the fact that the eliciting conditions of an emotion hold does not mean that a positive intensity is assigned to the emotion. Conversely, $\not\models \hat{\epsilon} \rightarrow \epsilon$ states that the fact that an emotion is currently being experienced, does not mean that its eliciting conditions currently hold (i.e. the emotion may have been triggered sometime in the past but its effect is still being 'felt').

5 DISCUSSION

The reason given in the OCC model for splitting quantitative aspects into potentials, thresholds, and intensities is so that a distinction can be made between what influences the intensity of an emotion (i.e. the variables constituting potential) and how strongly an emotion is actually experienced (i.e. initially, potential minus threshold, then decreasing over time). Translating this idea to doxastic logic, one may expect that if an emotional experience fluent is satisfied, this should be believed by the agent in question to emphasize the 'actual experience.' Formally, for any agent $i \in \mathcal{G}$, one may expect $\hat{\epsilon} \rightarrow \mathbf{B}_i\hat{\epsilon}$ to be derivable, where $\hat{\epsilon} \in \{\, \hat{\epsilon} \mid \epsilon \in \mathit{Em}(i) \,\}$ (i.e. all emotional experience fluents with i as agent subscript). We can easily attain this as a proposition by placing the following constraint on our models:

$$\forall(\epsilon, q_i, t_0, \mu, \delta) \in \mathit{Emem}(\mathrm{M}, s) : \forall s' \in R_{\mathbf{B}}(\mathit{agent}_\epsilon)(s) : (\epsilon, q_i, t_0, \mu, \delta) \in \mathit{Emem}(\mathrm{M}, s')$$

for all $(\mathrm{M}, s) \in \Sigma$ (we slightly abuse notation by writing agent_ϵ to extract the agent index from ϵ). Of course, we do *not* want (and indeed do not have) a similar proposition for emotion triggering fluents, i.e. $\epsilon \rightarrow \mathbf{B}_i\epsilon$ for all $i \in \mathcal{G}$ and $\epsilon \in \mathit{Em}(i)$ is not derivable.

In [7] the eliciting conditions of the emotions *hope* and *fear* from the OCC model are formalized and propositions relating the resulting (qualitative) emotion triggering fluents are investigated. However, there is a discussion in [4] on the relation between their intensities, which must be taken into account when investigating the quantitative aspects of hope and fear. Specifically, this discussion applies to simultaneous hope with respect to a prospective desirable event and fear with respect to the absence of the same event. It is noted that in such a case, the intensities of these emotions should sum to a constant. We are now in a position to uncover the assumptions that have to be made in order to make this the case. First, in the formal qualitative model, the formula $\mathbf{hope}_i(\pi, \kappa) \wedge \mathbf{fear}_i(\pi, \neg\kappa)$ (i.e. hope that plan π will accomplish goal κ and fear it may not) must be contingent, which is indeed the case [7]. Note that the accomplishment of a goal κ constitutes the desirable event while commitment to a plan π to bring about κ constitutes the prospect. According to OCC, the intensities of hope and fear are determined by the (un)desirability of the prospective event and its likelihood, i.e. in more formal terms:

$$pot(\mathrm{M}, s)(\mathbf{hope}_i(\pi, \kappa)) = w_1\, des(i)(\mathrm{M}, s)(\kappa) + w_2\, lik(i)(\mathrm{M}, s)(\pi, \kappa)$$
$$pot(\mathrm{M}, s)(\mathbf{fear}_i(\pi, \neg\kappa)) = w_3\, undes(i)(\mathrm{M}, s)(\neg\kappa) + w_4\, lik(i)(\mathrm{M}, s)(\pi, \neg\kappa)$$

where $(un)des$ is a function returning the (un)desirability of a goal, lik is a function returning the (estimated) likelihood of a plan accomplishing a goal, and $w_{\_}$ are state-dependent weights. Without detailing their definitions, it is reasonable to assume that lik behaves such that $lik(i)(\mathrm{M}, s)(\pi, \kappa) = 1 - lik(i)(\mathrm{M}, s)(\pi, \neg\kappa)$, while des and $undes$ applied to complementary arguments should behave such that $des(i)(\mathrm{M}, s)(\kappa) = undes(i)(\mathrm{M}, s)(\neg\kappa)$ [4]. Assuming $w_2 = w_4$, we obtain $pot(\mathrm{M}, s)(\mathbf{hope}_i(\pi, \kappa)) + pot(\mathrm{M}, s)(\mathbf{fear}_i(\pi, \neg\kappa)) = (w_1 + w_3)\, des(i)(\mathrm{M}, s)(\kappa) + w_2$. So as long as the desirability of the event and the weights remain constant for the duration of the prospect (which is not an unreasonable assumption), the sum of the potentials of complementary hope and fear emotions also remains constant. Thus, the (estimated) likelihood of the event can vary freely over time without affecting the sum above. Furthermore, if the assumption is made that $thr(\mathrm{M}, s)(\mathbf{hope}_i(\pi, \kappa)) = -thr(\mathrm{M}, s)(\mathbf{fear}_i(\pi, \neg\kappa))$ (which is reasonable if thresholds are defined in terms of a 'mood' variable), the initial intensities (q_i above) will sum to the same constant (assuming both potentials are greater than the respective thresholds). The reader may have noticed the frequent usage of the word "assumption" in this paragraph, which shows that this formalization is capable of explicating many constraints (although reasonable) that are needed to capture the intuitions of the OCC model. Adopting these constraints will render our model completely in line with OCC.

6 CONCLUSIONS AND FUTURE WORK

In this paper we have presented part of a qualitative formalization of the OCC model of emotions, specifying the conditions that elicit emotions. The thrust of this formalization was to show how emotion-related concepts can be translated to an agent specification language. Moreover, we have shown how quantitative aspects of emotions can be integrated into the qualitative model in order to model the actual experience of emotions. However, certain details pertaining to the calculation of emotion quantities are missing in the OCC model. We have proposed a method for calculating emotion intensities and investigated its properties. Finally, we have explicated some of the implicit assumptions that underlie intuitions of the OCC model.

For future work, relations between intensities of opposing emotions with respect to the same objects should be investigated for emotions other than hope and fear, in a fashion similar to the Discussion above. Other issues that need to be addressed include the specifications of emotion potentials and thresholds, the dynamics of intensity function parameters, and the influence of the experience of emotions on the deliberation and decision making of agents.

ACKNOWLEDGEMENTS

This work is supported by SenterNovem, Dutch Companion project grant nr: IS053013.

REFERENCES

[1] J. Gratch and S. Marsella, 'A domain-independent framework for modeling emotions', *J. of Cognitive Systems Research*, **5**(4), 269–306, (2004).
[2] J.-J.Ch. Meyer, 'Reasoning about emotional agents', in *Proceedings of ECAI'04*, pp. 129–133. IOS Press, (2004).
[3] J.-J.Ch. Meyer, W. v.d. Hoek, and B. v. Linder, 'A logical approach to the dynamics of commitments', *Artificial Intelligence*, **113**, 1–40, (1999).
[4] A. Ortony, G.L. Clore, and A. Collins, *The Cognitive Structure of Emotions*, Cambridge University Press, Cambridge, UK, 1988.
[5] R.W. Picard, *Affective Computing*, MIT Press, 1997.
[6] A. Sloman, 'Beyond shallow models of emotion', *Cognitive Processing*, **2**(1), 177–198, (2001).
[7] B.R. Steunebrink, M. Dastani, and J.-J.Ch. Meyer, 'A logic of emotions for intelligent agents', in *Proceedings of AAAI'07*. AAAI Press, (2007).