

Bayesian Affect Control Theory

Jesse Hoey
School of Computer Science
University of Waterloo
Waterloo, Ontario, N2L3G1
Email: jhoey@cs.uwaterloo.ca

Tobias Schröder
Centre for Theoretical Neuroscience
University of Waterloo
Waterloo, Ontario, N2L3G1
Email: post@tobiasschroeder.de

Areej Althothali
School of Computer Science
University of Waterloo
Waterloo, Ontario, N2L3G1
Email: aalhotha@cs.uwaterloo.ca

Abstract—Affect Control Theory is a mathematical representation of the interactions between two persons, in which it is posited that people behave in a way so as to minimize the amount of deflection between their cultural emotional sentiments and the transient emotional sentiments that are created by each situation. Affect Control Theory presents a maximum likelihood solution in which optimal behaviours or identities can be predicted based on past interactions. Here, we formulate a probabilistic and decision theoretic model of the same underlying principles, and show this to be a generalisation of the basic theory. The model is more expressive than the original theory, as it can maintain multiple hypotheses about behaviours and identities simultaneously as a probability distribution. This allows the model to generate affectively believable interactions with people by learning about their identity and predicting their behaviours. We demonstrate this generalisation with a set of simulations. We then show how our model can be used as an emotional “plug-in” for systems that interact with humans. We demonstrate human-interactive capability by building a simple intelligent tutoring application and pilot-testing it in an experiment with 20 participants.

I. INTRODUCTION

Designers of intelligent systems have increasingly attended to theories of human emotion, in order to build software interfaces that allow users to experience naturalistic flows of communication with the computer. This endeavour requires a comprehensive mathematical representation of the relations between affective states and actions that captures, ideally, the subtle cultural rules underlying human communication and emotional experience. In this paper, we show that Affect Control Theory (ACT), a mathematically formalized theory of the interplays between cultural representations, interactants’ identities, and affective experience [1], is a suitable framework for developing emotionally intelligent agents. To accomplish this, we propose a probabilistic and decision theoretic generalisation of ACT, called *BayesAct*, which we argue is more flexible than the original statement of the theory for the purpose of modelling human-computer interaction. The key contributions of this theory are: (1) to represent emotions as probability distributions over a continuous space; (2) to allow affective identities to be dynamic and uncertain; (3) to endow agents with the ability to learn affective identities of their interactants; and (4) to introduce explicit utility functions that parsimoniously tradeoff emotional and propositional goals for a human-interactive agent. This final contribution is what allows *BayesAct* to be used in human-computer interaction: it provides the computerised *agent* with a mechanism for predicting how the affective state of an interaction will progress

(based on Affect Control Theory) and how this will affect the object of the interaction (e.g. the software application being used). The *agent* can then select its strategy of action in order to maximize the expected values of the outcomes based both on the application state and on its emotional alignment with the human. The main contribution of this paper is of a theoretical nature, demonstrated in simulation. We have also implemented the theory in a simple tutoring system, and we report the results of an empirical survey and pilot study with humans.

II. BASIC MODELS

A. Affect Control Theory

Affect Control Theory (ACT) is a comprehensive social psychological theory of human social interaction [1], proposing that peoples’ social perceptions, actions, and emotional experiences are governed by a psychological need to minimize deflections between culturally shared fundamental sentiments about social situations and transient impressions resulting from the dynamic behaviours of interactants in those situations.

Fundamental sentiments f are representations of social objects, such as interactants’ identities and behaviours, as vectors in a three-dimensional affective space, hypothesised to be a universal organising principle of human socio-emotional experience [2]. The basis vectors of affective space are called Evaluation/valence, Potency/control, and Activity/arousal (EPA). EPA profiles of concepts can be measured with the semantic differential, a survey technique where respondents rate affective meanings of concepts on numerical scales with opposing adjectives at each end (e.g., good, nice vs. bad, awful for E, weak, little vs. strong, big for P, and calm, passive vs. exciting, active for A). Affect control theorists have compiled databases of a few thousand words along with average EPA ratings obtained from survey participants who are knowledgeable about their culture [3]. For example, most English speakers agree that professors are about as nice as students (E), however more powerful (P) and less active (A). The corresponding EPA profiles are [1.7, 1.8, 0.5] for professor and [1.8, 0.7, 1.2] for student (values range by convention from -4.3 to $+4.3$ [3]). In general, within-cultural agreement about EPA meanings of social concepts is high even across subgroups of society, and cultural-average EPA ratings from as little as a few dozen survey participants have been shown to be extremely stable over extended periods of time [3].

Social events can cause transient impressions τ of identities and behaviours that deviate from their corresponding

fundamental sentiments f . ACT models this formation of impressions from events with a minimalist grammar of the form actor-behaviour-object. Consider for example a professor (actor) who yells (behaviour) at a student (object). Most observers would agree that this professor appears considerably less nice (E), a bit less potent (P), and certainly more aroused (A) than the cultural average of a professor. Such transient shifts in affective meaning caused by specific events can be described with models of the form $\tau = Mf$, where M is a matrix of statistically estimated prediction coefficients from empirical impression-formation studies. In these studies, survey respondents rated EPA affective meanings of concepts embedded in a few hundred sample event descriptions such as the example above [1]. In ACT, the weighted sum of squared Euclidean distances between fundamental sentiments and transient impressions is called deflection (weights: w_i):

$$D = \sum_i w_i (\mathbf{f}_i - \boldsymbol{\tau}_i)^2, \quad (1)$$

Affective Deflection is hypothesised to correspond to an aversive state of mind that humans seek to avoid. ACT is thus a variant of psychological consistency theories, which posit in general that humans strive for balanced mental representations whose elements form a coherent Gestalt [4], [5]. In cybernetic terminology, deflection is a control signal used for aligning everyday social interactions with implicit cultural rules and expectations [1]. For example, advising a student corresponds much better to the cultural expectation of a professor’s behaviour than yelling at a student. Correspondingly, the deflection for the former event as computed with the ACT equations is much lower than the deflection for the latter event. Many experimental and observational studies have shown that deflection is indeed inversely related to the likelihood of humans to engage in the corresponding social actions. For example, the deflection-minimization mechanism explains verbal behaviours of mock leaders in a computer-simulated business game [6] and non-verbal displays in dyadic interactions [7], among others.

B. Partially Observable Markov Decision Processes

A partially observable Markov decision process (POMDP) [8] is a general model of stochastic control that has been extensively studied in operations research and in artificial intelligence. A POMDP consists of a finite set \mathcal{X} of states; a finite set \mathcal{A} of actions; a stochastic transition model $\Pr : X \times A \rightarrow \Delta(X)$, with $\Pr(x'|x, a)$ denoting the probability of moving from state x to x' when action a is taken, and $\Delta(X)$ is a distribution over \mathcal{X} ; a finite observation set Ω ; a stochastic observation model with $\Pr(\omega|x)$ denoting the probability of making observation ω while the system is in state x ; and a reward assigning $R(a, x')$ to a state transition to x' induced by action a . A generic POMDP is shown as a decision network in Figure 1(a) (solid lines).

The POMDP can be used to monitor beliefs about the state using standard inference algorithms [9]. Finally, a *policy* can be computed that maps *belief states* (i.e., distributions over \mathcal{X}) into choices of actions, such that the expected discounted

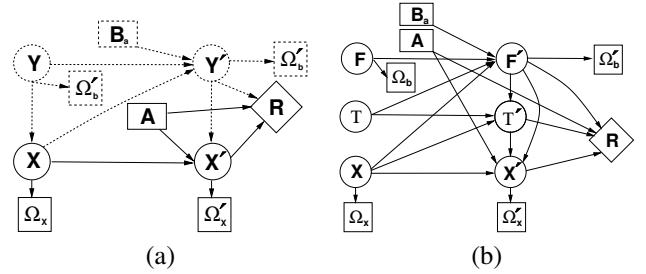


Fig. 1. Two time slices of (a) a general POMDP (solid lines) and a POMDP augmented with affective states (dotted lines); (b) a factored POMDP for Bayesian Affect Control Theory.

sum of rewards is (approximately) maximised. An interesting property of POMDP policies is that they may use “information gathering” actions. In the context of Affect Control Theory, an agent can take actions that temporarily increase deflection in order to discover something about the interactant, for example.

In this paper, we will be dealing with *factored* POMDPs in which the state is represented by the cross-product of a set of variables or features. Assignment of a value to each variable thus constitutes a state. Factored models allow for conditional independence to be explicitly stated in the model. POMDPs have been used as models for many human-interactive domains, including for intelligent tutoring systems [10], [11], and for human assistance systems [12].

C. Basic Formulation

Bayesian Affect Control Theory (*BayesAct* for short) gives us a principled way to add the emotional content to a human interactive system by making four key additions to the basic POMDP model, as shown by the dashed lines in Figure 1(a). (1) An unobservable variable, \mathbf{Y} , describes sentiments of the *agent* about identities and behaviours. The dynamics of \mathbf{Y} is given by empirical measurements in ACT (see below). (2) Observations Ω_b give evidence about the part of the sentiments \mathbf{Y} encoding behaviours of the interactant. (3) The actions of the agent are now expanded to be $\mathbf{B} = \{A, \mathbf{B}_a\}$. The normal state transition dynamics can still occur based only on A , but now the action space also must include an affective “how” for the delivery “what” of an action. (4) The application-specific dynamics of \mathbf{X} now depends on sentiments, $\Pr(\mathbf{X}'|\mathbf{X}, \mathbf{Y}', A)$, and will generally follow the original distribution $\Pr(\mathbf{X}'|\mathbf{X}, A)$, but now moderated by deflection. For example, \mathbf{X} may move towards a goal, but less quickly when deflection is high. We assume that time is discrete, and agents take turns acting (so the “turn” is one element of \mathbf{X}). This assumption does not limit the generality of the approach, as anything beyond simple turn-taking (e.g. backchannel responses, interruptions) could be included in \mathbf{X} , and time steps are defined by the transitions therein. *BayesAct* is shown as a graphical model in Figure 1(b). A fully detailed derivation is in [13].

Let $\mathbf{F} = \{F_{ij}\}$ denote the set of fundamental *agent* sentiments about itself where each feature F_{ij} , $i \in \{a, b, c\}$, $j \in \{e, p, a\}$ denotes the j^{th} fundamental sentiment (evaluation,

potency or activity) about the i^{th} interaction object: actor (*agent*), behaviour, or object (*client*). Let $\mathbf{T} = \{\mathbf{T}_{ij}\}$ be similarly defined and denote the set of transient *agent* sentiments. Variables F_{ij} and T_{ij} are continuous-valued. Affect Control Theory encodes the identities as being for “actor” (A, the person acting) and “object” (O, the person being acted upon). In our formulation, we encode identities as being for “agent” and “client” (regardless of who is currently acting), and encode who is acting in \mathbf{X} . The affective action $\mathbf{B}_a = \{\mathbf{B}_{ae}, \mathbf{B}_{ap}, \mathbf{B}_{aa}\}$ is continuous-valued and three dimensional: the agent sets an EPA value for the affective action component, not a propositional action label. The client behaviour is implicitly represented in the fundamental sentiment variables \mathbf{F}_b , and these are observed through an observations variable, Ω_b . We denote $\mathbf{Y}=\{\mathbf{F}, \mathbf{T}\}$, $\mathbf{S}=\{\mathbf{Y}, \mathbf{X}\}$ and $\Omega=\{\Omega_b, \Omega_x\}$.

The *deflection* in ACT is a nine-dimensional weighted Euclidean distance measure between fundamental sentiments \mathbf{F} and transient impressions \mathbf{T} (Section II-A). Here, we propose that this distance measure is the logarithm of a probabilistic potential $\varphi(\mathbf{f}', \boldsymbol{\tau}') \propto e^{-(\mathbf{f}' - \boldsymbol{\tau}')^T \Sigma^{-1} (\mathbf{f}' - \boldsymbol{\tau}')}$. The covariance Σ is a generalisation of the “weights” in the definition of deflection, Equation (1) [1]. The empirically derived prediction equations of ACT can be written as $\boldsymbol{\tau}' = \mathcal{M}\mathcal{G}(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x})$ where \mathcal{G} is a non-linear operator that combines $\boldsymbol{\tau}$, \mathbf{f}' , and \mathbf{x} , and \mathcal{M} is the prediction matrix (see Section II-A and [1]). Thus, we write $\varphi(\mathbf{f}', \boldsymbol{\tau}') \propto e^{-\psi(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x})}$, where ψ is the *deflection* between fundamental and transient sentiments: $\psi(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}) = (\mathbf{f}' - \mathcal{M}\mathcal{G}(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}))^T \Sigma^{-1} (\mathbf{f}' - \mathcal{M}\mathcal{G}(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}))$.

The probability distribution over fundamental sentiments, $\theta_f(\mathbf{f}'; \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a) = Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a, \theta_f)$, can be derived using the φ potential¹:

$$Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a, \varphi) \propto \int_{\theta_f, \boldsymbol{\tau}'} Pr(\theta_f, \mathbf{f}', \mathbf{f}, \boldsymbol{\tau}', \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a, \varphi) \\ = e^{-\psi(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x})} [\mathbb{E}_{Pr(\theta_f | \mathbf{x})}(\theta_f)] \quad (2)$$

The first term is a distribution over \mathbf{f}' that represents our assumption of minimal deflection, while the second is the expected value of the parameter θ_f given the prior. This expectation will give us the most likely value of θ_f given only the system state \mathbf{x} . We know two things about the transition dynamics (θ_f) that we can encode in the prior. First, we know that the behaviour will be set equal to the *agent*'s action if it is the *agent*'s turn (hence the dependence on \mathbf{x} , which encodes the turn). Second, we know that identities are not expected to change very quickly. Therefore, we have that

$$\mathbb{E}_{Pr(\theta_f | \mathbf{x})}(\theta_f) \propto e^{-\xi(\mathbf{f}', \mathbf{f}, \mathbf{b}_a, \mathbf{x})} \quad (3)$$

where $\xi(\mathbf{f}', \mathbf{f}, \mathbf{b}_a, \mathbf{x}) \equiv (\mathbf{f}' - \langle \mathbf{f}, \mathbf{b}_a \rangle)^T \Sigma_f^{-1}(\mathbf{x}) (\mathbf{f}' - \langle \mathbf{f}, \mathbf{b}_a \rangle)$, $\langle \mathbf{f}, \mathbf{b}_a \rangle$ is \mathbf{f} for the identities and \mathbf{b}_a for the behaviours and $\Sigma_f(\mathbf{x})$ is the covariance matrix for the inertia of the fundamentals, and including the setting of behaviour fundamentals

¹We are actually postulating an undirected link in the graph between $\boldsymbol{\tau}$ and \mathbf{f} . An easy way to handle this undirected link properly is to replace it with an equivalent set of directed links by adding a new Boolean variable, D , that is conditioned by both \mathbf{T} and \mathbf{F} , and such that $Pr(D = True | \boldsymbol{\tau}, \mathbf{f}) \propto \varphi(\boldsymbol{\tau}, \mathbf{f})$. We then set $D = True$ if we have the knowledge that \mathbf{T} and \mathbf{F} are close together according to $\varphi(\mathbf{F}, \mathbf{T})$, see [13]

by the *agent* action, and is a set of parameters governing the strength of our prior beliefs that the identities of client and agent will remain constant over time. Thus, Σ_f is a 9×9 block matrix with diagonal elements $[I_3 \beta_a^2, I_3 \beta_b^2(\mathbf{x}), I_3 \beta_c^2]$, where β_a^2 and β_c^2 are the variances of *agent* and *client* identity fundamentals (i.e. how much we expect *agent* and *client* to change their identities over time), and $\beta_b^2(\mathbf{x})$ is infinite for a *client* turn and is zero for an *agent* turn. We therefore have:

$$Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a, \varphi) \propto e^{-\psi(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}) - \xi(\mathbf{f}', \mathbf{f}, \mathbf{b}_a, \mathbf{x})} \quad (4)$$

which is a set of multivariate Gaussians indexed by \mathbf{x} that are non-linearly dependent on \mathbf{f} , \mathbf{b}_a and $\boldsymbol{\tau}$ through the matrices \mathcal{H} , \mathcal{C} (refactorizations of \mathcal{M} , \mathcal{G} , see Appendix A):

$$Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a, \varphi) \propto e^{-(\mathbf{f}' - \boldsymbol{\mu}_n)^T \Sigma_n^{-1} (\mathbf{f}' - \boldsymbol{\mu}_n)} \quad (5)$$

where:

$$\boldsymbol{\mu}_n = \Sigma_n \mathcal{H}^T(\boldsymbol{\tau}, \mathbf{x}) \Sigma^{-1} \mathcal{C}(\boldsymbol{\tau}, \mathbf{x}) + \Sigma_n \Sigma_f^{-1}(\mathbf{x}) (\mathbf{f}, \mathbf{b}_a) \\ \Sigma_n = (\mathcal{H}^T(\boldsymbol{\tau}, \mathbf{x}) \Sigma^{-1} \mathcal{H}(\boldsymbol{\tau}, \mathbf{x}) + \Sigma_f^{-1}(\mathbf{x}))^{-1}$$

We can now estimate what the most likely value of \mathbf{F}' by computing $\mathbf{F}'^* = \arg \max_{\mathbf{f}'} Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a, \varphi)$. Intuitively, this expression will be maximized for exactly the behaviour that minimizes the deflection as given by ψ , tempered by the inertia of changing identities given by ξ (see [13]).

Identities are treated as constants when optimising the behaviour in [1] (and similarly for identities: behaviours are held constant). However, ACT hypothesises that the overall deflection will be minimised. Therefore, our equations are the exact version of the approximations in [1].

The probability distribution over transient sentiments in *BayesAct* arises directly from the deterministic dynamics of ACT, written $P(\boldsymbol{\tau}' | \boldsymbol{\tau}, \mathbf{f}', \mathbf{x}) = \delta(\boldsymbol{\tau}' - \mathcal{M}\mathcal{G}(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}))$, where $\delta(z)$ is 1 if $z = 0$ and is 0 otherwise. Observation functions for the client behaviour sentiment and system state are $Pr(\omega_b | \mathbf{f})$ and $Pr(\omega_x | \mathbf{x})$, respectively. They are stochastic in general, but may be deterministic for the system state (so that \mathbf{X} is fully observable). The application dynamics is $Pr(\mathbf{x}' | \mathbf{x}, \mathbf{f}', \boldsymbol{\tau}', a)$ as discussed above. Finally, $R(\mathbf{f}, \boldsymbol{\tau}, \mathbf{x})$ is a reward function giving the immediate reward given to the *agent*. We assume an additive function $R(\mathbf{f}, \boldsymbol{\tau}, \mathbf{x}) = R_x(\mathbf{x}) + R_s(\mathbf{f}, \boldsymbol{\tau})$, where R_x encodes the application goals (e.g. to get a student to pass a test), and $R_s \propto -(\mathbf{f} - \boldsymbol{\tau})^2$ depends on the deflection.

The probability distribution over the sentiments and system state given the history of actions and observations is computed from the observations, *agent* action, and state at time t , $\Omega_t, \mathbf{B}_t, \mathbf{S}_t$:

$$b(\mathbf{s}_t) \equiv Pr(\mathbf{s}_t | \omega_0, \dots, \omega_t, \mathbf{b}_0, \dots, \mathbf{b}_t) \\ = \int_{\mathbf{s}_{t-1}} Pr(\mathbf{s}_t, \mathbf{s}_{t-1} | \omega_0, \dots, \omega_t, \mathbf{b}_0, \dots, \mathbf{b}_t) \\ \propto Pr(\omega_t | \mathbf{s}_t) \mathbb{E}_{b(\mathbf{s}_{t-1})} [Pr(\mathbf{s}_t | \mathbf{s}_{t-1}, \mathbf{b}_t)] \quad (6)$$

where $Pr(\mathbf{s}_t | \mathbf{s}_{t-1}, \mathbf{b}_t)$ factored according to Figure 1(b):

$$Pr(\mathbf{s}_t | \dots) = Pr(\mathbf{x}' | \mathbf{x}, \mathbf{f}', \boldsymbol{\tau}', a) Pr(\boldsymbol{\tau}' | \boldsymbol{\tau}, \mathbf{f}', \mathbf{x}) Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \mathbf{b}_a) \quad (7)$$

This gives us a recursive formula for computing the distribution over the state at time t as an expectation of the transition dynamics taken with respect to the distribution at time $t - 1$.

D. Computing Policies

Policies for POMDPs in general can be computed using a number of methods, but recent progress in using Monte-Carlo (sampling) based methods has shown that very large POMDPs can be solved tractably, and that this works equally well for continuous state and observation spaces [14], [15]. POMCP [15] is a Monte-Carlo based method for computing policies in POMDPs with discrete action spaces. POMCP can be generalised to a mixed continuous-discrete action space for *BayesAct* by leveraging the fact that we can *predict* what an agent would “normally” do in any state according to the underlying Affect Control Theory: it is the action that minimises the deflection. This normative prediction constrains the space of actions over which the *agent* must plan, and drastically reduces the branching factor of the search space.

To compute the policy, we denote the “normal” action distribution as $\pi^\dagger(\mathbf{s})$:

$$\begin{aligned}\pi^\dagger(\mathbf{s}') &= \int_{\mathbf{f}'_a, \mathbf{f}'_c, \mathbf{s}} Pr(\mathbf{f}' | \mathbf{f}, \boldsymbol{\tau}, \mathbf{x}, \varphi) b(\mathbf{s}) \\ &= \int_{\mathbf{f}'_a, \mathbf{f}'_c, \mathbf{s}} e^{-(\mathbf{f}' - \boldsymbol{\mu}_n^\dagger)^T (\boldsymbol{\Sigma}_n^\dagger)^{-1} (\mathbf{f}' - \boldsymbol{\mu}_n^\dagger)} b(\mathbf{s})\end{aligned}\quad (8)$$

where $\boldsymbol{\mu}_n^\dagger$ and $\boldsymbol{\Sigma}_n^\dagger$ are the same as Equation (5) except that $\boldsymbol{\Sigma}_f$ has $\beta_b^2(\mathbf{x})$ set to infinity (instead of zero) so the behaviour sentiments are unconstrained. Equation (8) computes the expected distribution over \mathbf{f}' given $b(\mathbf{s})$ and then marginalises (sums) out the identity components to get the distribution over \mathbf{f}_b . We can make a further approximation that $b(\mathbf{s}) = \delta(\mathbf{s}^* - \mathbf{s})$ where $\mathbf{s}^* = \{\mathbf{f}^*, \boldsymbol{\tau}^*, \mathbf{x}^*\} = \mathbb{E}_{b(\mathbf{s})}[\mathbf{s}] = \int_{\mathbf{s}} \mathbf{s} b(\mathbf{s})$ is the expected state (or one could use $\mathbf{s}^* = \arg \max_{\mathbf{s}} b(\mathbf{s})$ as the most likely state). We will denote the resulting action distribution as $\pi^{\dagger*}(\mathbf{s}')$. In this paper, we don’t use the full POMCP solution, instead only taking a “greedy” action that looks one step into the future by drawing samples from the “normal” action distribution in Equation (8) using these to compute the expected next reward, and selecting the (sampled) action $\mathbf{b}_a^{\dagger*}$ that maximizes this.

$$\mathbf{b}_a^{\dagger*} = \arg \max_{\mathbf{b}_a} \int_{\mathbf{s}'} Pr(\mathbf{s}' | \mathbf{s}^*, \mathbf{b}_a) R(\mathbf{s}') d\mathbf{s}', \quad \mathbf{b}_a \sim \pi^{\dagger*}(\mathbf{s}') \quad (9)$$

In practice we make two further simplifications: we avoid the integration over \mathbf{f}_a and \mathbf{f}_c in Equation (8) by drawing samples from the distribution over \mathbf{f}' and selecting the \mathbf{f}'_b components, and we compute the integration in Equation (9) by sampling from the integrand and averaging.

E. Sampling

To compute the belief distribution (Equation (6)), we represent $b(\mathbf{s})$ using a set of samples [16]. This will allow us to represent complex belief distributions, including multi-modal distributions over identities, which can be very useful in cases where the *agent* is uncertain about the the *client* identity. In such a case, the *agent* can maintain multiple hypotheses, and shift its belief to the one that agrees most with evidence from an interaction. If $\mathbf{s}_i = \{\mathbf{f}_i, \boldsymbol{\tau}_i, \mathbf{x}_i\}$ and w_i is the weight of the i^{th} sample, the belief state is $b(\mathbf{s}) \propto \sum_{i=1}^N w_i \delta(\mathbf{s} - \mathbf{s}_i)$, where $\delta(x) = \infty$ at $x = 0$ and 0 elsewhere.

Then, we implement Equation (6) using a sequential Monte Carlo method sampling technique, also known as a *particle filter* or *bootstrap filter* [16], as follows. We start at time $t = 0$ with a set of samples and weights $\{\mathbf{s}_i, w_i\}_{i=1 \dots N}$, which together define a belief state $b(\mathbf{s}_0)$. The *agent* then consults the policy to retrieve a new action \mathbf{b}_a , takes action \mathbf{b}_a and receives observation $\boldsymbol{\omega}$, samples (with replacement) unweighted samples from $b(\mathbf{s})$ (current weights), and then samples from Equation (7) to get $b(\mathbf{s}'_i)$. New weights are then computed for each sample using the observation functions $w_i = Pr(\boldsymbol{\omega} | \mathbf{s}'_i)$. We also add some “roughening” white noise (in $[-\sigma_r, \sigma_r]$, $\sigma_r = N^{-1/3}$) to the unweighted samples [16] to deal with the sparse initial sampling of the identity space (for unknown identity) coupled with the underlying assumption that the identities are very slowly changing.

We can compute expected values of quantities of interest, such as the deflection, by summing over the weighted set of samples $d(\mathbf{s}_i) = \sum_{i=1}^N w_i (\mathbf{f}_i - \boldsymbol{\tau}_i)^2$.

III. EXPERIMENTS AND RESULTS

Our goal in this section is to demonstrate that *BayesAct* can discover the affective identities of persons it interacts with, and that *BayesAct* can augment practical applications with affective dynamics. To establish these claims, we do the following. First, we demonstrate analytically that *BayesAct* can reproduce exactly the affective dynamics predicted by Affect Control Theory’s original mathematical model. The analytical derivation is done by reducing Equation (4) to the equations in [1] (see derivation in [13]). Second, we verify that *BayesAct* produces comparable simulation outcomes to *Interact*, the software that implements Affect Control Theory². To this end, we run the *BayesAct* software alongside *Interact* and can show that the identical sentiments and actions are generated across a range of different *agent* and *client* identities. These analytical and empirical demonstrations show that *BayesAct* can be used as a model of human affective dynamics, since it has been shown empirically that *Interact* is a close model of human affective dynamics [6], [7]. Thus, *BayesAct*, as *Interact*, can predict *agent-client* dynamics *when identities are fixed*. Third, we show that if we loosen the constraints on the *client* identity being known, *BayesAct* can go beyond *Interact* and “discover” or learn this identity during an interaction with a *BayesAct* client that knows the identity of the *BayesAct agent*. Fourth, we show that if both *agent* and *client* do not know the identity of their interactant, they can both learn this identity simultaneously. Finally, we postulate that, since the *agent* can learn the affective identity of its *client*, it can better serve the *client* in an effective manner. We provide some insights into this with a basic experiment with humans in Section III-B.

A. Simulations

We have two conditions: the *client* knows or doesn’t know the identity of the *agent* (denoted “agent id known” and “agent id hidden”, resp.). In both cases, the *agent* does not

²see indiana.edu/~socpsy/ACT. We used the Indiana 02-04 database for identities/behaviours and the USA1978 equations for dynamics.

N	<i>id-deflection</i>			
	agent id known		agent id hidden	
	agent	client($\times 10^3$)	agent	client
5	3.8 ± 1.8	0.18 ± 0.039	4.2 ± 2.3	3.8 ± 1.3
10	2 ± 1.2	0.16 ± 0.031	2.4 ± 1.1	2.2 ± 0.98
50	0.82 ± 0.48	0.09 ± 0.033	1 ± 0.57	0.88 ± 0.57
100	0.83 ± 0.6	0.07 ± 0.039	0.76 ± 0.39	0.92 ± 0.72
250	0.61 ± 0.59	0.055 ± 0.05	0.63 ± 0.49	0.5 ± 0.21
500	0.49 ± 0.33	0.038 ± 0.02	0.57 ± 0.48	0.57 ± 0.31
1000	0.42 ± 0.52	0.038 ± 0.03	0.4 ± 0.21	0.44 ± 0.22

TABLE I

IDENTITY DEFLECTIONS (SIMULATIONS): $\sigma_e = 1.0$, $N = \text{NO. SAMPLES}$.

know the identity of the client. We run 20 trials for each condition, and in each trial a new identity is chosen for each of *agent* and *client*. These two identities are independently sampled from the distribution of identities in the *Interact* database and are the personal identities for each *agent* and *client*. Then, *agent* and *client* *BayesAct* models are initialised with \mathbf{F}_a set to this personal identity, \mathbf{F}_c (identity of the other, if not known) set to the mean of the identities in the database, $[0.4, 0.4, 0.5]$, and \mathbf{F}_b set to zeros (this is irrelevant). The simulation runs as defined in Section II-E for 50 steps. Agents take turns, and actions are conveyed with the addition of some zero-mean normally distributed “environment” noise, with standard deviation σ_e . Agents use Gaussian observation models with uniform covariances with diagonal variances $\gamma = \max(0.5, \sigma_e)$. To enable comparisons with *Interact*, actions are selected by averaging draws of 100 samples from Equation (8). We perform 10 simulations per trial and average the results with $\beta_a = \beta_c = 0.001$ for both *agent* and *client*.

We use *id-deflection* to denote the sum of squared differences between one agent’s estimate of the other agent’s identity, and that other agent’s estimate of its’ own identity. Table I shows the mean (over 20 trials) of the average (over 10 experiments) final (at the end of the last step) *id-deflection* for *agent* and *client*, for $\sigma_e = 1.0$. We see that about 100 samples is sufficient to get an estimate within about 1 in identity deflection. Note that we are comparing the expected values of identities which may be different than any mode.

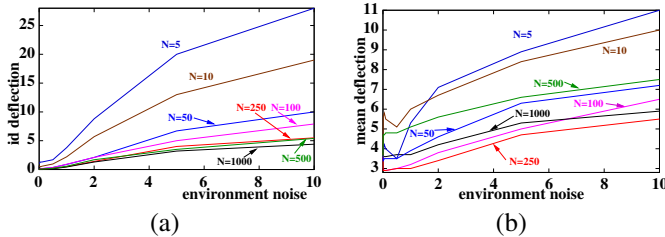


Fig. 2. Deflections of identities from simulations with different numbers of samples (N), and environment noise, σ_e . Roughening noise: $\sigma_r = N^{-1/3}$, model noise: $\gamma = \max(0.5, \sigma_e)$. (a) *id-deflection* and (b) mean deflection

Figure 2 shows a plot of how the *id-deflection* and total deflection changes with the environment noise, σ_e and sample numbers N , for *agent id hidden*. We see that only about 100 samples are needed to get a solution that is robust to values of

$\sigma_e = 2.0$ in environment noise³. Surprisingly, deflection is not as strongly affected by environment noise. This is because the agent has a correct model of the environment noise ($\gamma \approx \sigma_e$), and can average the noisy measurements and come up with a low deflection solution. The deterministic program *Interact* would have more trouble, as it must “believe” exactly what it gets (it has no model of environment noise).

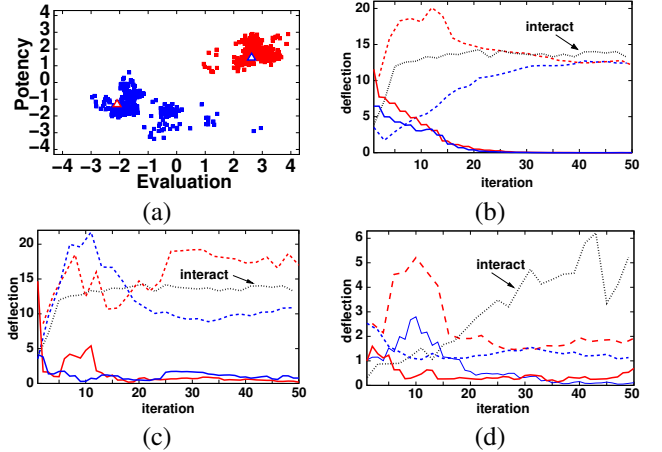


Fig. 3. (a) samples (squares) and true identities (triangles) after 7 iterations for one trial. (b-d) Closer look at five 200 sample experiments showing *id-deflection* (solid lines), total deflections (dashed lines) and deflection from *Interact* in dotted black (red=*agent*, blue=*client*).

Figure 3(a) shows a sample set after 7 iterations of one experiment, clearly showing the multimodal distributions centered around the true identities (triangles) of each interactant. Figures 3(b)-(d) look more closely at three of the trials done with $N = 200$ samples and hidden ids for both *client* and *agent*. The red and blue lines show the *agent-* and *client-* *id-deflection* (solid) and *agent* and *client* deflections (dashed), respectively, while the black line shows the deflections using *Interact* (which has the correct and fixed identities for both agents at all times). *BayesAct* allows identities to change, and starts with almost no information about the identity of the other interactant (for both *agent* and *client*). We can see that our model gets at least as low a deflection as *Interact*. In Figure 3(b), the *agent* had $\mathbf{F}_a = [2.7, 1.5, 0.9]$, and the *client* had $\mathbf{F}_a = [-2.1, -1.3, -0.2]$, and $\sigma_e = 0$ (noise-free communication). These two identities do not align very well⁴, and result in high deflection when identities are known and fixed in *Interact* (black line). *BayesAct* rapidly estimates the correct identities, and tracks the deflection of *Interact*. Figure 3(c) is the same, but with $\sigma_e = 1.0$. We see that *BayesAct* is robust to this level of noise. Figure 3(d) shows a simulation between a “tutor” ($\mathbf{F}_a = [1.5, 1.5, -0.2]$) and a “student” ($\mathbf{F}_c = [1.5, 0.3, 0.8]$) with $\sigma_e = 1.0$. Here *Interact* predicts larger deflections. *BayesAct* also gets a larger deflection, but manages to resolve it, and identities are properly learned.

³For example, this corresponds to enough noise to make a behaviour of “apprehend” be mis-communicated as “confide in”.

⁴These identities are closest to “lady” and “shoplifter” for *agent* and *client* respectively, but recall that identity labels come from mapping the computed EPA vectors to concepts in ACT repositories [3] and are not used by *BayesAct*.

B. Tutoring Application

To demonstrate the capability of *BayesAct* to control emotionally plausible behaviours of a computer program interacting with a human, we built a simple tutoring application in which the identities for *agent* and *client* are initially set to “tutor” and “student”, respectively, with low dynamics variances of $\beta_a = \beta_c = 0.01$. The application asks sample questions from the Graduate Record Exam (GRE) educational testing system, and the *client* clicks on a multiple-choice answer. The *agent* provides feedback as to whether the *client*’s answer is correct. The client then has the opportunity to “speak” by clicking on a labelled button (e.g., “that was too hard for me!”). The statement maps to a behaviour from the ACT database (e.g., “whine”). These mappings were determined in an empirical survey described below. The behaviour label in turn maps to the value for F_b found in the ACT database (in this case $[-1.4, -0.8, -0.5]$). *BayesAct* then computes an appropriate agent action, i.e. a vector in EPA space, which maps to a behaviour label (e.g., “apologize”), which again maps to a statement (e.g., “Sorry, I may have been too demanding on you.”).

Mappings of *agent* and *client* statements to behaviour labels contained in the ACT database were based on an empirical online survey. Participants were $N = 37$ (22 female) students (avg. age: 30.6 years). We presented them with four blocks of statements and behaviour labels, two blocks referring to *agent* and *client* behaviours conditional on a correct/incorrect answer of the *client*. We also asked participants to rate the affective meaning of each statement directly using the semantic differential [3]. The survey resulted in a list of 27 agent statements and 18 client statements for which a clear majority of participants agreed on a mapping. We implemented these behaviours and the corresponding mappings to behaviour labels from the ACT database as the possible actions in *BayesAct* tutoring system.

The tutor has three discrete elements of state $\mathbf{X} = \{X_d, X_s, X_t\}$ where X_d is the difficulty level, X_s is the skill level of the student and X_t is the turn. X_d and X_s have 3 integer “levels” where lower values indicate easier difficulty/skill. The tutor’s model of the student’s progress is $P(X'_s = x_s | x_s, \mathbf{f}, \boldsymbol{\tau}') = 0.9$ with the remaining probability mass distributed evenly over skill levels that differ by 1 from x_s . The dynamics for all values where $X'_s \leq x_s$ are then multiplied by $(\mathbf{f}' - \boldsymbol{\tau}')^2/2$ and renormalised. Thus, skill level changes inversely proportionally to deflection. The tutor gets observations of whether the student succeeded/failed ($\Omega_x = 1/0$), and has an observation function $P(\Omega_x | X_d, X_s)$ that favours success if X_d matches X_s . The reward is the sum of the negative deflection and $R_x(\mathbf{x}) = -(\mathbf{x} - 2)^2$ and Equation (9) is used for action selection.

We conducted a pilot experiment with 20 participants (7 female) who were mostly undergraduate students of engineering or related disciplines (avg. age: 25.8). We compared the experiences of 10 users interacting with the *BayesAct* tutor with those of 10 users interacting with a control tutor whose actions were selected randomly from the same set as the

Survey Question	BACT	rand.	T	p
Communication was similar to a human.	2.10	1.70	0.87	n.s.
MT acted like it understood my feelings	2.80	2.00	1.92	< .05
I felt emotionally connected with MT.	2.50	2.00	1.00	n.s.
I enjoyed interacting with MT.	3.30	3.10	0.44	n.s.
MT gave awkward responses (RC)	3.44	4.50	-2.70	< .01
I found MT to be flexible to interact with.	3.00	1.90	2.18	< .05
Using MT would improve my skills.	3.40	2.00	2.49	< .05
The dialogue was simple and natural.	3.50	2.00	3.14	< .01
Overall, I am satisfied with this system.	2.70	1.70	2.34	< .05

TABLE II

USER STUDY RESULTS. T HAS DF=18 AND p IS ONE-TAILED. SCALES RANGED FROM 1 (=NOT TRUE) TO 5 (=TRUE).
(BACT=*BayesAct*, MT=MATH TUTOR, RC=REVERSE-CODED)

BayesAct tutor⁵. Participants completed a short survey after using the system for an average of 20 minutes. Results are displayed in Table II. Users seemed to experience the flow of communication with the *BayesAct* tutor as more simple, flexible, and natural than with the random control tutor. The mean deflection for *BayesAct* was 2.9 ± 2.1 while for random it was 4.5 ± 2.2 . We have to treat these results from a small sample with caution. Future versions would need to be expanded and made to better reflect actual student development. In particular, one could expand the state space \mathbf{X} to include more features related to the application and student skills than the simple 3-valued difficulty and skill levels we have used. This would require more complex models of the transitions in \mathbf{X} (e.g. goals and problem dimensions [10], [11]), of the observations of \mathbf{X} (e.g. from sensors and sentiment mappings), and of the dependence of the sentiments on the state.

IV. RELATED WORK

ACT has important conceptual similarities with appraisal theories, which are a prominent psychological framework in affective computing research [17], [18]. Appraisal theories generally posit that emotional states are generated from cognitive appraisals of events in the environment in relation to the goals of the agent [19]. As a result, an agents’ emotions depend more on its subjective interpretations than on the physical features of an event. Appraisal theorists describe a set of fixed rules or criteria for mapping specific patterns of cognitive evaluations onto specific emotional states. The logic of ACT is quite similar: Emotional states result from (linguistic) interpretations of observed events. The EPA dimensions of affective space can be understood as very basic appraisal rules related to the goal congruence of an event (E), the agent’s coping potential (P), and the urgency implied by the situation (A). For discussion of ACT vs appraisal theories see [20]. However, ACT is also more general and more parsimonious than many appraisal theories, since it works without explicitly defining complex sets of rules relating specific goals and states of the environment to specific emotions. Instead, ACT treats the dynamics of emotional states and behaviours as continuous trajectories in affective space. Deflection minimisation is the

⁵*BayesAct* used 500 samples, $\beta_a = \beta_c = 0.01$, and took 4 seconds per interaction on an AMD phenom IIx4 955 3.20 GHz with 8GB RAM running Windows 7, while displaying the words “Thinking...”. The random tutor simply ignored the computed response.

only prescribed mechanism, while the more specific goals tied to types of agents and situations are assumed to emerge from the semantic knowledge in the model (i.e., the culture-specific databases of EPA meanings of concepts). *BayesAct* further allows *client* goals to be explicitly encoded and optimised.

Shank [21] and Troyer [22] describe experiments to gather EPA profiles of technology or computer terms. Shank gives positive answers to three questions about the use of Affect Control Theory in the modelling of human interactions with technology. He shows that people have shared cultural identity labels for technological actors, and that they share affective sentiments about these labels. He also shows that people view these technological actors as behaving socially, as explored in [23]. The answers to these questions open the doors for the usage of ACT in technology, as we do here.

POMDPs have been used as models for intelligent tutoring systems [10], including with emotional states [11]. Bayesian networks and probabilistic models have also seen recent developments [17] based on appraisal theory [19]. Our work has demonstrated that, by operating completely in a dimensional space, we can surmount computational issues, assure scalability (the size only grows with the amount of state necessary for the application, not with the number of emotion labels), and we can explicitly encode prior knowledge obtained empirically through a well-defined methodology [3].

V. CONCLUSIONS

This paper has presented a probabilistic and decision theoretic formulation of Affect Control Theory called *BayesAct*, and has shown its use for human interactive systems. The paper's main contributions are the theoretical model development, and a demonstration that a computational agent can use *BayesAct* to integrate reasoning about emotions with application decisions in a parsimonious and well-grounded way. In future, the measurement of EPA behaviours and the translation of EPA actions requires further study. We plan to develop the planning aspects, and to parallelize the code (for which the sampling method is ideally suited). We plan to investigate usages of the model for collaborative agents in more complex domains, for manipulative, competitive and therapeutic agents, and for social simulations.

Acknowledgements: We thank our study participants and Dr. Poupart. We acknowledge support from the Natural Sciences and Engineering Council of Canada (Hoey), DFG research fellowship #SCHR1282/1-1 (Schröder), and King Abdulaziz University, Saudi Arabia (Althohali).

REFERENCES

- [1] D. R. Heise, *Expressive Order: Confirming Sentiments in Social Actions*. Springer, 2007.
- [2] C. E. Osgood, W. H. May, and M. S. Miron, *Cross-Cultural Universals of Affective Meaning*. University of Illinois Press, 1975.
- [3] D. R. Heise, *Surveying Cultures: Discovering Shared Conceptions and Sentiments*. Wiley, 2010.
- [4] F. Heider, "Attitudes and cognitive organization," *Journal of Psychology: Interdisciplinary and Applied*, vol. 21, pp. 107–112, 1946.
- [5] P. Thagard, *Coherence in Thought and Action*. MIT Press, 2000.
- [6] T. Schröder and W. Scholl, "Affective dynamics of leadership: An experimental test of affect control theory," *Social Psychology Quarterly*, vol. 72, pp. 180–197, 2009.
- [7] T. Schröder, J. Netzel, C. Schermuly, and W. Scholl, "Culture-constrained affective consistency of interpersonal behavior," *Social Psychology*, vol. 44, pp. 47–58, 2013.

- [8] K. J. Åström, "Optimal control of Markov decision processes with incomplete state estimation," *J. Math. Anal. Appl.*, vol. 10, 1965.
- [9] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann, 1988.
- [10] J. T. Folsom-Kovarik, G. Sukthankar, and S. Schatz, "Tractable POMDP representations for intelligent tutoring systems," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 2, March 2013.
- [11] C. Conati and H. Maclaren, "Empirically building and evaluating a probabilistic model of user affect," *UMUAI*, vol. 19, 2009.
- [12] J. Hoey, C. Boutilier, P. Poupart, P. Olivier, A. Monk, and A. Mihailidis, "People, sensors, decisions: Customizable and adaptive technologies for assistance in healthcare," *ACM Trans. IIS*, vol. 2, no. 4, 2012.
- [13] J. Hoey, T. Schröder, and A. Althohali, "Affect control processes: Probabilistic and decision theoretic affective control in human-computer interaction," 2013. Available as University of Waterloo School of Computer Science Technical Report CS-2013-03.
- [14] J. M. Porta, N. Vlassis, M. T. Spaan, and P. Poupart, "Point-based value iteration for continuous POMDPs," *JMLR*, vol. 7, 2006.
- [15] D. Silver and J. Veness, "Monte-Carlo planning in large POMDPs," in *Proc. NIPS*, December 2010.
- [16] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo in Practice*. Springer-Verlag, 2001.
- [17] J. Sabourin, B. Mott, and J. C. Lester, "Modeling learner affect with theoretically grounded dynamic Bayesian networks," in *Proc. Affective Computing and Intelligent Interaction*, 2011.
- [18] K. R. Scherer, T. Banziger, and E. Roesch, *A Blueprint for Affective Computing*. Oxford University Press, 2010.
- [19] A. Ortony, G. Clore, and A. Collins, *The Cognitive Structure of Emotions*. Cambridge University Press, 1988.
- [20] K. B. Rogers, T. Schröder, and C. von Scheve, "Dissecting the sociality of emotion: A multi-level approach," *Emotion Review*, forthcoming.
- [21] D. B. Shank, "An affect control theory of technology," *Current Research in Social Psychology*, vol. 15, no. 10, pp. 1–13, 2010.
- [22] L. Troyer, "An affect control theory as a foundation for the design of socially intelligent systems," in *Proc. AAAI Spring Symp. on Architectures for Modeling Emotion*, 2004.
- [23] B. Reeves and C. Nass, *The media equation*. Cambridge University Press, 1996.

APPENDIX A

To derive Equation (5), we use symbolic indices in $\{a, b, c\}$ and $\{e, p, a\}$, and define two simple index dictionaries, d_i and d_a , to map between the symbols and numeric indices $\{0, 1, 2\}$ in matrices and vectors so that, $d_i(a) = 0, d_i(b) = 1, d_i(c) = 2$ and $d_a(e) = 0, d_a(p) = 1, d_a(a) = 2$. Thus, we can write an element of \mathbf{F} as \mathbf{F}_{ba} which will be the k^{th} element of the vector representation of \mathbf{F} , where $k = 3d_i(b) + d_a(p)$. We will use a "dot" to represent that all values are present if there is any ambiguity. For example, the behaviour component of \mathbf{F} is written \mathbf{F}_b , or simply \mathbf{F}_b when there is no ambiguity.

Since \mathcal{G} only uses the behaviour component of \mathbf{f}' , we can group terms together and write $\mathbf{M}\mathcal{G}(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}) = \mathcal{H}(\boldsymbol{\tau}, \mathbf{x})\mathbf{f}'_b - \mathcal{C}(\boldsymbol{\tau}, \mathbf{x})$, where \mathcal{H} and \mathcal{C} are 9×3 and 9×1 matrices of coefficients that are a refactoring of the operators $\mathbf{M}\mathcal{G}$ such that a simple linear function of \mathbf{f}'_b is obtained. If \mathcal{H}_{ijk} is the element at row $3d_i(i) + d_a(j)$, column $d_a(k)$ of \mathcal{H} , then the sum of all terms in row $3d_i(i) + d_a(j)$ of $\mathbf{M}\mathcal{G}(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x})$ that contain B_k , is then $\mathcal{H}_{ijk}B_k$. Similarly, the element at row $d_i(i)$ of \mathcal{C} , \mathcal{C}_i , is the sum of all terms in row $d_i(i)$ of $\mathbf{M}\mathcal{G}$ that contain no B . at all. Recall that, due to our model being over "agent" and "client", instead of over "actor" and "object", the matrices \mathcal{H} and \mathcal{C} will change depending on the turn, swapping "agent" and "object" in both \mathbf{M} and \mathcal{G} .

We can rewrite Equation (4) by defining

$$\mathcal{H} = \begin{bmatrix} \mathcal{H}_a \\ \mathcal{H}_b \\ \mathcal{H}_c \end{bmatrix} \quad \mathcal{K} = \begin{bmatrix} \mathbf{I}_3 & -\mathcal{H}_a & \mathbf{O}_3 \\ \mathbf{O}_3 & \mathbf{I}_3 - \mathcal{H}_b & \mathbf{O}_3 \\ \mathbf{O}_3 & -\mathcal{H}_c & \mathbf{I}_3 \end{bmatrix}$$

where $\mathcal{H}_a \equiv \mathcal{H}_{a..}$ (a 3×3 matrix giving the rows of \mathcal{H} in which \mathcal{H}_{ijk} have $i = a$) and similarly for \mathcal{H}_b and \mathcal{H}_c . We also define \mathbf{I}_3 as the 3×3 identity matrix and \mathbf{O}_3 as the 3×3 matrix of all zeros. Using \mathcal{K} , we arrive at the general form for ψ :

$$\psi(\mathbf{f}', \boldsymbol{\tau}, \mathbf{x}) = (\mathbf{f}' - \mathcal{K}^{-1}\mathcal{C})^T \mathcal{K}^T \boldsymbol{\Sigma}^{-1} \mathcal{K}(\mathbf{f}' - \mathcal{K}^{-1}\mathcal{C}) \quad (10)$$

Equation (4) is therefore a product of two Gaussians, itself also a Gaussian the mean and covariance of which can be obtained by completing the squares to find a covariance, $\boldsymbol{\Sigma}_n$ equal to the sum in quadrature of the covariances, and a mean, $\boldsymbol{\mu}_n$, that is proportional to a weighted sum of $\mathcal{K}^{-1}\mathcal{C}$ and $\langle \mathbf{f}, \mathbf{b}_a \rangle$, with weights given by the normalised covariances of $\boldsymbol{\Sigma}_n \boldsymbol{\Sigma}_\tau^{-1}$ and $\boldsymbol{\Sigma}_n \boldsymbol{\Sigma}_f^{-1}$, respectively, as shown in Equation (5), where $\boldsymbol{\Sigma}_\tau \equiv \mathcal{K}^{-1} \boldsymbol{\Sigma} (\mathcal{K}^T)^{-1}$.