CHALMERS
UNIVERSITY OF TECHNOLOGY

Chance-constrained active inference

Citation for the original published paper (version of record):
Van De Laar, T., Şenöz, İ., Ozcelikkale, A. et al (2021). Chance-constrained active inference. Neural Computation, 33(10): 2710-2735. http://dx.doi.org/10.1162/neco_a_01427

Chance-Constrained Active Inference

Thijs van de Laar[1], İsmail Şenöz[1], Ayça Özçelikkale[2], and Henk Wymeersch[3]

[1]Eindhoven University of Technology, The Netherlands.

[2]Uppsala University, Sweden.

[3]Chalmers University of Technology, Sweden.

Keywords: Active Inference, Message Passing, Chance Constraints, Variational Bayes

Abstract

Active Inference (ActInf) is an emerging theory that explains perception and action in biological agents, in terms of minimizing a free energy bound on Bayesian surprise. Goal-directed behavior is elicited by introducing prior beliefs on the underlying generative model. In contrast to prior beliefs, which constrain all realizations of a random variable, we propose an alternative approach through chance constraints, which allow

for a (typically small) probability of constraint violation, and demonstrate how such constraints can be used as intrinsic drivers for goal-directed behavior in ActInf. We illustrate how chance-constrained ActInf weights all imposed (prior) constraints on the generative model, allowing e.g., for a trade-off between robust control and empirical chance constraint violation. Secondly, we interpret the proposed solution within a message passing framework. Interestingly, the message passing interpretation is not only relevant to the context of ActInf, but also provides a general purpose approach that can account for chance constraints on graphical models. The chance constraint message updates can then be readily combined with other pre-derived message update rules, without the need for custom derivations. The proposed chance-constrained message passing framework thus accelerates the search for workable models in general, and can be used to complement message-passing formulations on generative neural models.

1 Introduction

Similar to biological agents, learning to make decisions based on observations and feedback from the environment is also an essential task for autonomous artificial agents. Traditionally, adaptive linear control and model predictive control have been successfully applied in this area (Borrelli et al., 2017). Over the past few years, reinforcement learning has become the predominant approach (Recht, 2019). An emerging alternative perspective to decision making under uncertainty is *active inference* (ActInf) (Friston, 2010). ActInf is a neuroscience-based theory that has been used extensively to explain behavior of biological agents in dynamic environments (Friston, 2010).

ActInf is based in the *free energy principle* (FEP), and postulates that perception and action in biological agents minimize a free energy bound on Bayesian surprise. The free energy is an information-theoretic measure that bounds the current and the future expected statistical surprise, i.e., how unpredictable are the observations under a given generative model (GM). The free energy is associated with the Kullback-Leibler (KL) divergence (i.e., the distance) between the approximate and the true posterior. In particular, according to the free energy principle, the agent acts in such a way as to minimize a free-energy bound on the surprise, i.e., Bayesian surprise which, informally speaking, provides a quantification of the difference between the agent's predictions about the system behavior and the observed system behavior. Minimization of free energy is closely related to variational Bayesian methods, reinforcement learning (Sallans and Hinton, 2001; Tschantz et al., 2020; Sajid et al., 2021), and deep generative models (Ueltzhöffer, 2018; Fountas et al., 2020), another set of popular machine learning approaches (Goodfellow et al., 2014). ActInf is closely related to message passing on graphical models (de Vries and Friston, 2017; Friston et al., 2017), and several widely used message passing algorithms, including (loopy) belief propagation, variational message passing and expectation propagation can be derived as fixed-point equations of the (Bethe) free energy (Heskes, 2003; Yedidia et al., 2005; Dauwels, 2007; Zhang et al., 2017). This relation has been harnessed to develop elegant automated methods for ActInf (Schwöbel et al., 2018; van de Laar and de Vries, 2019).

In addition to investigation of motivating connections with the behavior of the biological systems (Friston et al., 2006; Ramstead et al., 2018), ActInf has been successfully utilized in applications in the traditional stochastic control scenarios, such as linear

quadratic Gaussian (LQG) control and similar standard problems such as maze problems (Hoffmann and Rostalski, 2017; Ueltzhöffer, 2018; Schwöbel et al., 2018; Baltieri and Buckley, 2019; Millidge et al., 2020; Imohiosen et al., 2020), and exploration-exploitation balancing in multi-armed bandit problems (Markovic et al., 2021).

Despite these promising developments, the ActInf framework lacks certain desirable features present in model predictive control. In particular, there is no off-the-shelf standard ActInf formulation that allows inclusion of chance constraints in the problem setting. Chance constraints provide an attractive approach for on-line decision making for uncertain systems (Mesbah, 2016), i.e., systems where the dynamics are not fully known or the system contains certain components that are best modeled in a stochastic manner. In such settings, constraints on the system behavior, such as the agent remaining in a given region of the environment with a given probability, cannot directly be encoded in terms of prior beliefs. In contrast to approaches that constrain all realizations of the random variables, chance constraints allow for a (typically small) probability of constraint violation, which can significantly improve performance since chance constraints enable the decision maker trade performance with probability of constraint violation (Blackmore et al., 2011).

This paper proposes a computationally tractable approach to chance-constrained decision making, and applies it to an ActInf context. We include chance constraints in the ActInf objective (i.e., the free energy) by using the Lagrangian formalism. We then solve the Lagrangian optimization problem by variational calculus. Finally, we show that the proposed solution not only leads to a modular and scalable message passing framework for ActInf problems, but also provides a general purpose message passing

framework that can account for chance constraints on graphical models in general. We claim the following main contributions:

1. We show that the analytic solution to the chance-constrained free energy problem yields posterior beliefs in the form of truncated mixtures. (Theorem 1)

2. We show how this solution can be interpreted in terms of message passing on a factor-graph representation of the generative model. (Theorem 2)

3. Consequently, our results provide a message passing framework that is specifically designed to account for chance constraints.

Message passing is inherently modular, and (variational) message update rules can be pre-derived and stored in a lookup table for later use (Korl, 2005; van de Laar, 2019). The chance-constrained message updates can then be readily combined with these pre-derived rules, without the need for laborious derivations. Our results illustrate that the proposed framework can successfully find solutions so that the rate of constraint violation specified in the original problem and the one that is actually observed during the closed-loop operation are close. The results also illustrate how to balance the constraints on the actions and the states through the usage of a tuning parameter, which enables exploration of different trade-offs between immediate and delayed intervention.

2 Problem Statement

We start by defining a general factorized generative model f with respect to an (arbitrary) collection of variables x. As a notational convention, individual variables are

indexed by $i, j \in \mathcal{V}$, and factors by $a, b, c \in \mathcal{F}$, unless stated otherwise. The model then factorizes as

$$f(x) = \prod_{a \in \mathcal{F}} f_a(x_a)\,, \tag{1}$$

with non-negative real functions f_a, and where $x_a \subset x$ collects the arguments of f_a. In a probabilistic generative model, the individual factors usually represent conditional probability distributions. Probabilistic inference is then concerned with obtaining an (approximate) posterior belief $q_j(x_j) \propto \int f(x) \mathrm{d}x_{\setminus j}$ over a variable of interest x_j, where $x_{\setminus j}$ indicates the integration over all model variables except x_j.

We now briefly recap how the computation of these beliefs can be performed efficiently and automated over a factor graph (Loeliger et al., 2004), and how this process can be interpreted as a Bethe free energy minimization problem (Yedidia et al., 2005). With these concepts firmly in place, we move to chance constraints and the formal problem statement in Sec. 2.4.

2.1 Factor Graphs for Marginal Belief Computation

A factor graph can be used to visually represent a factorized function. In this paper we use the bi-partite factor graph representation. A bi-partite factor graph

$$\mathcal{G} = (\mathcal{F}, \mathcal{V}, \mathcal{E})\,,$$

consists of variable-nodes $\mathcal{V}$, factor-nodes $\mathcal{F}$, and edges $\mathcal{E}$ that connect variable-nodes with factor-nodes. A variable-node $i \in \mathcal{V}$ is connected to a factor-node $a \in \mathcal{F}$ by an edge $(i, a) \in \mathcal{E}$ if (and only if) the variable x_i is an argument of the factor-function f_a. An example section of a graph is drawn in Fig.1, where the circle and square represent

a variable- and factor-node respectively. We write the neighborhood of a variable-node

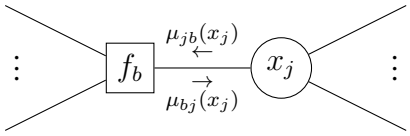

Figure 1: Bi-partite subgraph of a model around a variable-node j (circle) and factor-node b (square), with indicated messages. Ellipses represent a continuation of the model.

i as $\mathcal{F}(i)$, which collects all factor-nodes in $\mathcal{F}$ that are direct neighbors of i. Similarly, $\mathcal{V}(a)$ collects all variable-nodes in $\mathcal{V}$ that are direct neighbors of a.

Suppose we are interested in obtaining a posterior belief $q_j(x_j)$. The belief propagation algorithm (Pearl, 1982) then prescribes we send messages from the branches of the graph towards the variable-node of interest, following the recursive application of the belief propagation update rules:

$$\mu_{jb}(x_j) = \prod_{\substack{a \in \mathcal{F}(j) \\ a \neq b}} \mu_{aj}(x_j) \tag{2a}$$

$$\mu_{bj}(x_j) = \int f_b(x_b) \prod_{\substack{i \in \mathcal{V}(b) \\ i \neq j}} \mu_{ib}(x_i) \, \mathrm{d}x_{b \setminus j} \,, \tag{2b}$$

where $x_{b \setminus j}$ collects all x_b with the exception of x_j. Here, $\mu_{jb}(x_j)$ represents the message from a variable-node $j \in \mathcal{V}$ to a neighboring factor-node $b \in \mathcal{F}(j)$; and reversely for $\mu_{bj}(x_j)$. These messages are illustrated in Fig. 1. The posterior belief can then be expressed as

$$q_j(x_j) = \frac{1}{Z_j} \mu_{jb}(x_j) \mu_{bj}(x_j) \,, \tag{3}$$

with $Z_j = \int \mu_{jb}(x_j)\mu_{bj}(x_j)\,\mathrm{d}x_j$ a normalizing constant.

In practice, for numerical stability, messages are often re-normalized after computation. Furthermore, messages are usually scheduled for computation, and are often referred to by their position in the schedule instead of their location in the graph. We will use a similar notation in Sec. 4. See (Bishop, 2006) for a more detailed introduction to (approximate) inference on bi-partite graphs.

2.2 Bethe Free Energy Interpretation

The Bethe free energy for a factorized model of the form of (1) is defined as

$$F[q] = \sum_{a\in\mathcal{F}} U_a[q_a] - \sum_{a\in\mathcal{F}} H[q_a] + (d_i - 1)\sum_{i\in\mathcal{V}} H[q_i]\,, \tag{4}$$

where d_i represents the degree of variable x_i. Here $U_a[q_a] = -\int q_a(x_a)\log f_a(x_a)\,\mathrm{d}x_a$ denotes the average energy for factor f_a, and $H[q_a] = -\int q_a(x_a)\log q_a(x_a)\,\mathrm{d}x_a$ denotes the entropy. The Bethe free energy is optimized with imposed normalization and marginalization constraints:

$$\int q_a(x_a)\,\mathrm{d}x_{a\setminus j} = q_j(x_j), \forall a \in \mathcal{F}, \forall j \in \mathcal{V}(a) \tag{5a}$$

$$\int q_a(x_a)\,\mathrm{d}x_a = 1, \forall a \in \mathcal{F} \tag{5b}$$

$$\int q_i(x_i)\,\mathrm{d}x_i = 1, \forall i \in \mathcal{V}\,, \tag{5c}$$

such that the q_a and q_i represent (approximate) posterior probability distributions (beliefs).

2.3 Free Energy Minimization for Active Inference

Active Inference usually defines dynamic models that specialize variables into parameters, states, observation and control sequences for past and future times. Free energy minimization for ActInf is then presented as a dual objective, where minimization of free energy for a model of past variables accounts for state and parameter estimation (perception), and free energy minimization of free energy for a model of future variables accounts for policy planning (Baltieri and Buckley, 2018; van de Laar et al., 2019).

In the present paper we assume that the current state is observed and that model parameters are given. Therefore, this paper only concerns inference for policy planning. Extensions for perception are however straightforward. Chance constraints only affect inference for planning, and therefore standard techniques for state estimation and parameter learning can be employed (van de Laar and de Vries, 2019).

Furthermore, the current paper employs the Bethe Free Energy (BFE) formulation (4) for policy planning (Schwöbel et al., 2018; van de Laar and de Vries, 2019) instead of the more traditional Expected Free Energy (EFE) (Friston et al., 2015). The BFE is known to lack the epistemic qualities of the EFE (Schwöbel et al., 2018), which can be compensated for by introducing an additional mutual information term between the states and the observations to the BFE objective (Parr and Friston, 2019). The benefit of the uncompensated BFE however, is that traditional message passing algorithms, including (loopy) belief propagation, variational message passing, expectation propagation and generalized belief propagation algorithms can all be derived as fixed-point equations of the variational free energy by the use of variational calculus, see (Yedidia et al., 2000; Heskes, 2003; Yedidia et al., 2005; Dauwels, 2007; Zhang et al., 2017).

2.4 Chance Constraints

A chance constraint imposes that the probability mass of a belief $q_j(x_j), j \in \mathcal{V}$ outside of a 'safe' region $\mathcal{S}_j \subset \mathcal{X}_j$ cannot exceed a pre-set threshold $\epsilon \in [0, 1]$. Formally, a chance constraint imposes the inequality

$$\begin{aligned} 1 - \epsilon &\leq \int_{\mathcal{S}_j} q_j(x_j) \, \mathrm{d}x_j \\ &= \int_{\mathcal{X}_j} q_j(x_j) \, g_j(x_j) \, \mathrm{d}x_j \, , \end{aligned} \tag{6}$$

with

$$g_j(x_j) = \begin{cases} 1 \text{ if } x_j \in \mathcal{S}_j \\ \\ 0 \text{ otherwise} \, . \end{cases}$$

Our problem statement then becomes two-fold, namely:

1. Find the stationary points of the Bethe free energy (4) under the normalization and marginalization constraints of (5) and chance constraints of the form (6) (Theorem 1);

2. Interpret the retrieval of stationary points of the chance-constrained Bethe free energy as message passing on a factor graph (Theorem 2).

The simulations of Sec. 4 further specialize the model variables into state, observation and control sequences and demonstrate the added value of chance constraints in an ActInf setting. Crucially, with an interpretation of chance constraints in terms of message passing on a factor graph, chance constraints can be readily applied to any factorized model. Formulating chance constraints as a click-on module for approximate inference then greatly improves the application range of chance constraints.

3 Chance-Constrained Message Passing

In this section we formulate the method of chance-constrained message passing. We identify the stationary points of the chance-constrained Bethe free energy and interpret the result in terms of message passing on a factor graph. We work towards a practical message-passing update rule for chance-constrained variables, as summarized in Algorithm 1. A brief introduction to variational calculus is available in Appendix A. Proofs can be found in Appendix B.

3.1 Stationary Points

From the Bethe free energy (4) and the constraints of (5), (6), we can construct the Lagrangian

$$\begin{aligned} L[q] = F[q] &+ \sum_{i\in\mathcal{V}} \gamma_i \left[\int q_i(x_i)\,\mathrm{d}x_i - 1\right] + \sum_{a\in\mathcal{F}} \gamma_a \left[\int q_a(x_a)\,\mathrm{d}x_a - 1\right] \\ &+ \sum_{a\in\mathcal{F}} \sum_{i\in\mathcal{V}(a)} \int \zeta_{ia}(x_i) \left[q_i(x_i) - \int q_a(x_a)\,\mathrm{d}x_{a\setminus i}\right] \mathrm{d}x_i \\ &+ \sum_{i\in\mathcal{V}} \eta_i \left[\int q_i(x_i) g_i(x_i)\,\mathrm{d}x_i - (1-\epsilon)\right], \end{aligned} \tag{7}$$

where the Lagrange multipliers γ, ζ, η enforce the constraints of (5), (6).

Under strong duality, for the inequality constraint in (6) we have the complementary slackness condition (Boyd and Vandenberghe, 2004, Ch. 5). This condition states that for optimality we have $\eta_i \left[\int q_i(x_i) g_i(x_i)\,\mathrm{d}x_i - (1-\epsilon)\right] = 0$. Therefore, either $\eta_i > 0$, which implies that the chance constraint of (6) holds with equality (active) or $\eta_i = 0$, which implies that the chance constraint may hold without equality (inactive). In other words, the complementary slackness condition requires us to consider two scenarios: i)

(6) holds with equality for $\eta_i > 0$ and ii) (6) is satisfied under $\eta_i = 0$. Hence, if $\eta_i > 0$, the chance constraint is activated and enforced with equality.

In Lemmas 1, 2 we express the stationary points of $L[q]$ in terms of the beliefs. The proofs are presented in Appendix B.1 and Appendix B.2.

Lemma 1. *Stationary points of* (7) *as a functional of* $q_b, b \in \mathcal{F}$*, are of the form*

$$q_b^*(x_b) = \frac{1}{Z_b} f_b(x_b) \prod_{i \in \mathcal{V}(b)} \mu_{ib}(x_i)\,, \tag{8}$$

with

$$Z_b = \int f_b(x_b) \prod_{i \in \mathcal{V}(b)} \mu_{ib}(x_i)\, \mathrm{d}x_b$$

a normalizing constant.

Proof. See Appendix B.1. □

Note that the μ_{ib} have not yet been identified or interpreted as messages. We will explicitly make this connection in Sec. 3.3.

Lemma 2. *Stationary points of* (7) *as a functional of* $q_j, j \in \mathcal{V}$*, are of the form*

$$q_j^*(x_j; \eta_j) = \frac{1}{Z_j(\eta_j)} \exp(-\eta_j g_j(x_j)) \prod_{a \in \mathcal{F}(j)} \mu_{aj}(x_j)\,, \tag{9}$$

with

$$Z_j(\eta_j) = \int \exp(-\eta_j g_j(x_j)) \prod_{a \in \mathcal{F}(j)} \mu_{aj}(x_j)\, \mathrm{d}x_j$$

a normalizer that still depends on η_j*.*

Proof. See Appendix B.2. □

Note that, in contrast to (3), this result incorporates an additional exponential term for η_j. We will identify this multiplier in Sec. 3.2. However, we already know that when the chance constraint for j is inactive, hence $\eta_j = 0$ as a consequence of the complementary slackness condition. In this case, (9) reduces to (3).

3.2 Active Chance Constraint

In this section, we identify the stationary points under active chance constraint. The result is stated in Theorem 1.

Theorem 1. *Under active chance constraint, stationary points of* (7) *as a functional of* $q_j, j \in \mathcal{V}$ *are of the form*

$$q_j^*(x_j; \eta_j = \eta_j^*) = \begin{cases} \frac{1-\epsilon}{\Phi_j^{(0)}} q_j^{(0)}(x_j) & \text{if } x_j \in \mathcal{S}_j \\ \frac{\epsilon}{1-\Phi_j^{(0)}} q_j^{(0)}(x_j) & \text{otherwise,} \end{cases} \tag{10}$$

with

$$q_j^{(0)}(x_j) = q_j^*(x_j; \eta_j = 0)\,, \tag{11a}$$

$$\Phi_j^{(0)} = \int_{\mathcal{S}_j} q_j^{(0)}(x_j)\,\mathrm{d}x_j\,, \tag{11b}$$

$$\eta_j^* = \log(\epsilon\Phi_j^{(0)}) - \log(1-\epsilon) - \log(1-\Phi_j^{(0)})\,. \tag{11c}$$

Proof. See Appendix B.3. □

This remarkable result tells us that the corrected belief $q_j^*(x_j; \eta_j = \eta_j^*)$ is obtained by scaling the probability mass of the uncorrected belief $q_j^{(0)}(x_j)$ over the respective safe and unsafe regions. This defines the corrected belief as a mixture of truncated beliefs. The optimal scaling of (10) ensures that the overflow is equal to ϵ.

The complementary slackness condition ensures that the chance constraint is only enforced if the probability mass of the *unconstrained* belief overflows the 'safe' region $\mathcal{S}_j$ by more than ϵ; i.e., the uncorrected belief is 'unsafe' when

$$\epsilon < 1 - \Phi_j^{(0)}\,, \tag{12}$$

where we refer to $\Phi_j^{(0)}$ as the 'safe mass'.

If (12) is satisfied, then the posterior density $q_j^{(0)}(x_j)$ is corrected according to (10), which 'pushes' the probability mass (just) back inside the safe region.

3.3 Chance-Constrained Message Passing

In this section, we show that chance constraints (10) can be interpreted as auxiliary factor-nodes (with a specific node-function), and can be enforced by belief propagation in an augmented graph.

Theorem 2. *Given a bipartite graph $\mathcal{G} = (\mathcal{F}, \mathcal{V}, \mathcal{E})$ with a variable node $j \in \mathcal{V}$, and an associated Bethe free energy* (4) *with a chance constraint* (6) *on the belief $q_j(x_j)$. Then, stationary points of* (7) *can be obtained by belief propagation on an augmented graph $\mathcal{G}' = (\mathcal{F}', \mathcal{V}, \mathcal{E}')$, where*

$$\mathcal{F}' = \mathcal{F} \cup g \tag{13a}$$

$$\mathcal{E}' = \mathcal{E} \cup (j, g)\,, \tag{13b}$$

and auxiliary node function

$$f_g(x_j) = \begin{cases} \frac{1-\epsilon}{\Phi_j^{(0)}} & \text{if } x_j \in \mathcal{S}_j \\ \\ \frac{\epsilon}{1-\Phi_j^{(0)}} & \text{otherwise.} \end{cases} \tag{14}$$

Proof. See Appendix B.4. □

Theorem 2 shows that chance-constrained message passing can be seamlessly incorporated within the belief propagation framework. Chance constraints simply enter the model definition as auxiliary factors, whose factor function depends upon the incoming message, see Fig. 2. Because uncorrected belief (11a) is being represented by the (re-normalized) incoming message $\mu_{jg}(x_j)$, this allows for a modular application of chance constraints by augmenting the original graphical model with auxiliary nodes.

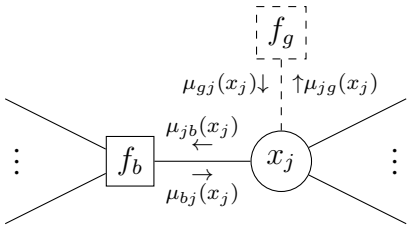

Figure 2: Bi-partite graph around a chance-constrained variable x_j, with indicated auxiliary factor f_g (dashed square) and messages. Ellipses represent the continued model by an arbitrary (possibly zero) number of connected edges.

3.4 Gaussian Approximation

Since the message $\mu_{gj}(x_j)$ introduces discontinuities, the computations for dependent messages may grow prohibitively complex. For efficient computations, it can be helpful to make a Gaussian approximation $\tilde{q}_j(x_j)$ to the corrected belief $q_j^*(x_j; \eta_j = \eta_j^*)$, e.g., by moment matching. The resulting (approximate) message then follows from

$$\mu_{gj}(x_j) = \tilde{q}_j^{(n)}(x_j)/\mu_{jg}(x_j)\,.$$

If the message $\mu_{jg}(x_j)$ is also Gaussian, this computation is easily performed by subtracting the canonical statistics. This procedure then resembles the expectation propagation algorithm (Minka, 2001; Cox and de Vries, 2018). Interestingly, the expectation propagation algorithm can also be derived in terms of Bethe free energy optimization, where the marginalization constraints (5a) are replaced by moment-matching constraints (Zhang et al., 2017). This makes the Gaussian approximation consistent with the Lagrangian approach as presented in this paper.

The approximated belief $\tilde{q}_j(x_j)$ however renders the chance constraint (6) inexact. As a result, the approximated belief needs to be iteratively re-corrected:

$$q_j^{(n)}(x_j) = \begin{cases} \frac{1-\epsilon}{\Phi_j^{(n-1)}} \tilde{q}_j^{(n-1)}(x_j) & \text{if } x_j \in \mathcal{S}_j \\ \frac{\epsilon}{1-\Phi_j^{(n-1)}} \tilde{q}_j^{(n-1)}(x_j) & \text{otherwise,} \end{cases} \tag{15}$$

where n denotes an iteration counter. This leads to the procedure summarized in Alg. 1, and depicted in Fig. 3.

With this algorithm, we have derived a practical chance-constrained message update from the first principles. The message update can be readily applied to any continuous variable that requires a chance constraint. Note however, that when multiple chance constraints are imposed on the model, the message passing algorithm itself becomes an iterative procedure because of circular message dependencies. For example, a message incoming to an auxiliary node g might (indirectly) depend on a message that exits another auxiliary node h. In turn, this exiting message depends on the incoming message to h (1), which depends on the message exiting g, etcetera. In order to break this circular message dependency, uninformative messages can be used to initialize the algorithm.

Algorithm 1 Chance-constrained message passing with Gaussian approximation

Given a Gaussian inbound message $\mu_{jg}(x_j)$

Compute the uncorrected belief $q_j^{(0)}(x_j)$ through (11a)

Compute the safe mass $\Phi_j^{(0)}$ through (11b)

Initialize the approximated belief $\tilde{q}_j^{(0)}(x_j) = q_j^{(0)}(x_j)$

Initialize the iteration counter $n = 0$

while $\epsilon + \delta < 1 - \Phi_j^{(n)}$ **do**

% Chance constraint is violated with some tolerance δ

Increase the counter $n \leftarrow n + 1$

Compute the corrected belief $q_j^{(n)}(x_j)$ through (15)

Approximate $\tilde{q}_j^{(n)}(x_j) \approx q_j^{(n)}(x_j)$ by Gaussian moment matching

Compute $\Phi_j^{(n)} = \int_{\mathcal{S}_j} \tilde{q}_j^{(n)}(x_j)\,\mathrm{d}x_j$, the safe mass of the approximated belief

end while

return The message $\mu_{gj}(x_j) = \tilde{q}_j^{(n)}(x_j)/\mu_{jg}(x_j)$

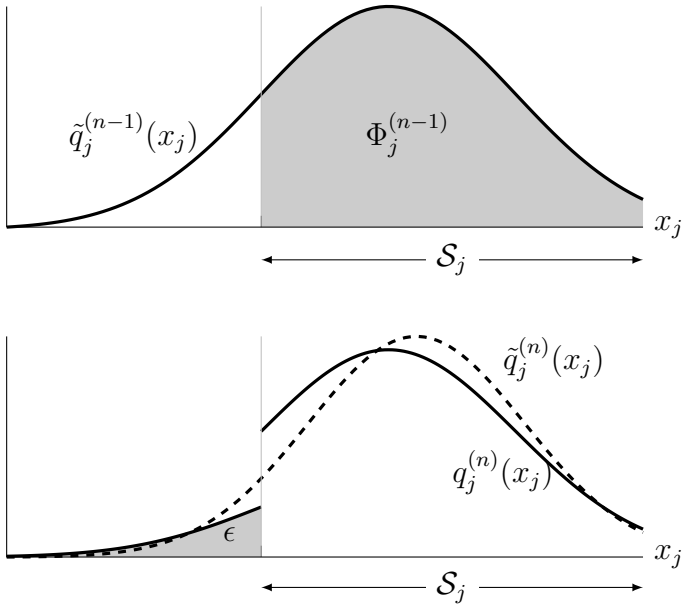

Figure 3: Example of beliefs as computed by Algorithm 1. The top figure evaluates the probability mass within the "safe" zone. The bottom figure applies the correction (solid curve) and approximates the corrected belief by Gaussian moment matching (dashed curve).

4 Simulations

In this section we simulate a drone that aims to elevate itself above a given height threshold with a preset probability, under the influence of a stochastic vertical wind. We define the drone elevation level over time by $x = \{x_0, \ldots, x_t, \ldots, x_L\}, x_t \in \mathbb{R}$, and actions (ascension velocity) $a = \{a_0, \ldots, a_t, \ldots, a_L\}, a_t \in \mathbb{R}$. A time-dependent $m_{w,t}$ defines the expected wind velocity that acts upon the agent. The discrete-time stochastic

system is defined as:

$$w_t \sim \mathcal{N}(m_{w,t}, v_w)$$

$$x_{t+1} = x_t + a_t + w_t \,,$$

where v_w defines the wind velocity variance.

We define an agent that directly observes its elevation level and has knowledge of the statistical system properties $m_{w,t}$ and v_w. The agent models future states of the system with a fixed time horizon T. As a shorthand notation, we write the future (including current) states $\overline{x}_t = \{x_t, \ldots, x_{t+T}\}$ and control variables $\overline{u}_t = \{u_t, \ldots, u_{t+T-1}\}$. For notational convenience, we drop the t subscript from these collections. The agent model at time t is defined as:

$$f_t(\overline{x}, \overline{u}) = \prod_{k=t}^{t+T-1} p_{x,k}(x_{k+1}|u_k, x_k) p_u(u_k) \,, \tag{16}$$

with a respective state transition model and control prior

$$p_{x,k}(x_{k+1}|u_k, x_k) = \mathcal{N}(x_{k+1}|x_k + u_k + m_{w,k}, v_w) \tag{17a}$$

$$p_u(u_k) = \mathcal{N}\big(u_k|0, \lambda^{-1}\big) \,. \tag{17b}$$

We factorize and constrain the variational posterior distribution such that (van de Laar and de Vries, 2019)

$$q_t(\overline{x}_{\setminus t}, \overline{u}) = q_t(\overline{x}_{\setminus t}) \prod_{k=t}^{t+T-1} \delta(u_k - a_k) \,, \tag{18}$$

where $\overline{x}_{\setminus t}$ indicates the collection of latent states (the state sequence $\overline{x}$ without the observed current state x_t). The goal of the agent controller then becomes to find the policy $\pi_t = \{a_t, \ldots, a_{t+T-1}\}$ that minimizes the Bethe free energy

$$F[q_t; x_t, \pi_t] = \int \cdots \int q_t(\overline{x}_{\setminus t}, \overline{u}) \log \frac{q_t(\overline{x}_{\setminus t}, \overline{u})}{f_t(\overline{x}, \overline{u})} \, \mathrm{d}\overline{x}_{\setminus t} \, \mathrm{d}\overline{u} \,, \tag{19}$$

under the normalization and marginalization constraints of (5) and chance constraints

$$1 - \epsilon \leq \int_{\mathcal{S}} q_{x,k}(x_k)\,\mathrm{d}x_k,\ \forall k \in \{t+1, \ldots, t+T\}\,,$$

where the safe region $\mathcal{S} = (1, \infty)$ and violation probability ϵ are identical for all future state variables.

4.1 Graphical Model and Schedule

As detailed in Sec. 3, Bethe free energy minimization under chance constraints can be performed by message passing on an augmented model. The graphical representation of the augmented model is depicted in Fig. 4.

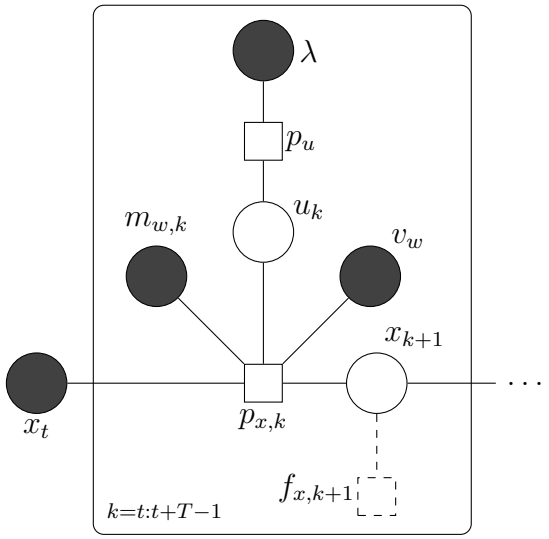

Figure 4: Augmented graphical representation of the agent model (16). Circles and squares indicate variable- and factor-nodes respectively. Auxiliary factor-nodes (14) are dashed, and dark circles indicate observed variables or fixed parameters. Ellipses indicate a continuation of the framed section until the lookahead time horizon.

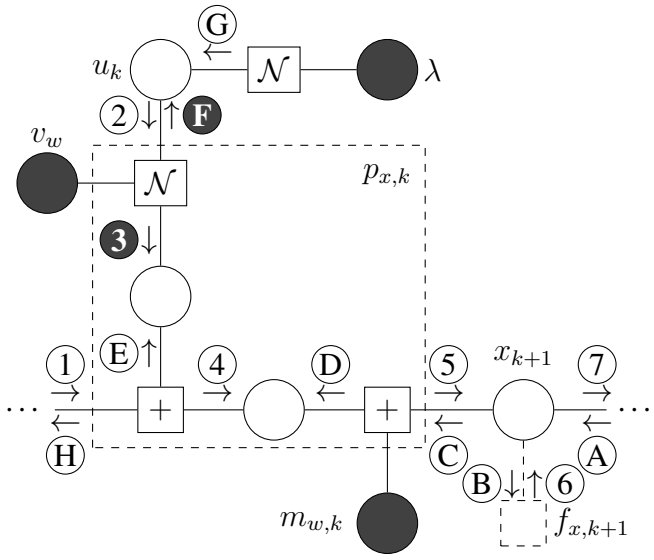

Figure 5: Augmented agent model (16), with $p_{x,k}$ expanded according to (17) (dashed rectangle), and indicated forward (numbers) and backward (letters) message passing schedules for optimization of (19). Circle and square nodes indicate variable- and factor-nodes respectively. Dark nodes indicate observed variables or fixed parameters, and auxiliary factor-nodes (14) are dashed. Ellipses indicate a continuation of the model. Dark messages are computed by the variational update rule, see (Winn and Bishop, 2005; Dauwels, 2007).

The schedule comprises a forward-backward scheme, as illustrated in Fig. 5. Four message updates in Fig. 5 are of particular interest. Firstly, since (18) constrains the belief over controls to a point-mass, it follows that

$$\mu_{②}^{(i)}(u_k) = \delta(u_k - a_k^{(i-1)}) \,,$$

where i counts the number of schedule (forward-backward) iterations. The schedule is initialized with $a_k^{(0)} = 0$ for all $k \geq t$. Secondly, $\mu_{Ⓑ}^{(i)}(x_{k+1})$ takes on the role of $\mu_{jg}(x_j)$ in Alg. 1. Because the noise in the model is Gaussian, this message will be an

(unnormalized) Gaussian as well. Therefore, by application of Alg. 1, the third message of interest, $\mu^{(i)}_{⑥}(x_{k+1})$ is computed. For the initial forward pass, $\mu^{(0)}_{Ⓑ}(x_{k+1}) = 1$ is considered uninformative. Fourthly, $\mu^{(i)}_{Ⓕ}(u_k)$ carries information upward to the control variables. Because the variational posterior is chosen to factorize between the state and control sequence (18), the $\mu^{(i)}_{Ⓕ}(u_k)$ message is computed by a variational update rule as detailed in (Winn and Bishop, 2005) and (Dauwels, 2007).

The action for the next iteration then follows from

$$q^{(i)}_{u,k}(u_k) \propto \mu^{(i)}_{Ⓕ}(u_k)\, \mu^{(i)}_{Ⓖ}(u_k)$$

$$a^{(i)}_k = \operatorname{mode} q^{(i)}_{u,k}(u_k)\,.$$

Iterating the schedule then corresponds with an expectation maximization scheme. The expectation step of this scheme computes the $\mu^{(i)}_{Ⓕ}(u_k)$ message from the actions $a^{(i-1)}_k$. The maximization step then chooses the updated actions $a^{(i)}_k$ as the current MAP-estimate of u_k. The schedule is iterated until the policy converges.

Message passing simulations[1] are performed with the ForneyLab probabilistic programming toolbox (Cox et al., 2019), version 0.11.3.

4.2 Control Law

Note that the Bethe free energy of (19) is still a function of the observed current elevation x_t. We can then evaluate the optimal action a_t as a function of the current elevation x_t (the control law), for a given wind profile, chance constraint and model parameters. In order to gain an intuition for controller behavior, we fix $m_{w,t} = 0$ for all t. We

[1] Source code for the simulations is available for download at `http://biaslab.github.io/materials/cc_simulations.zip`

plot the control law in Fig. 6, for varying values of the lookahead horizon T, chance constraint threshold ϵ, wind variance v_w and control prior precision λ.

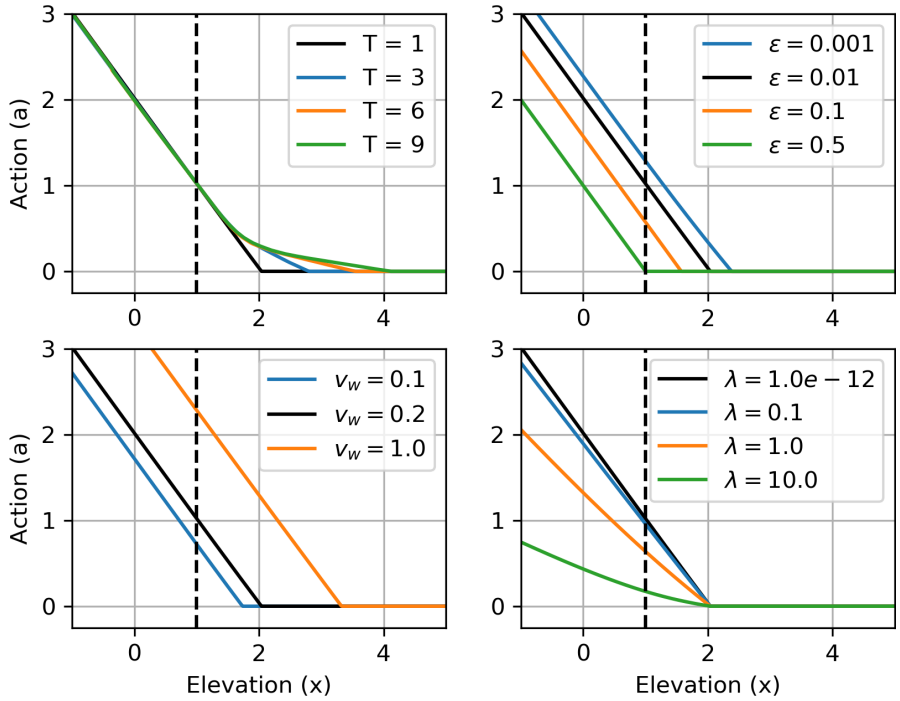

Figure 6: Slices of the control law for $m_{w,t} = 0, \mathcal{S} = (1, \infty)$, varied around reference setting $T = 1, \epsilon = 0.01, v_w = 0.2, \lambda = 10^{-12}$ (black curves). Dashed vertical lines indicate the minimal safe elevation.

The top-left diagram shows that with growing lookahead horizon T, the agent starts intervening at higher elevation. With this anticipatory effect the agent prepares for events in the more distant future. The top-right diagram also shows that the agent intervenes at higher elevation with decreasing ϵ. When violation of the constraint grows less desirable, the agent must intervene earlier in order to assure that sufficient probability mass is present in the safe region. Also note that no further action is proposed beyond an intervention threshold. Once the agent is sufficiently elevated, no corrections are

proposed until the agent wanders (or is forced) below the intervention threshold. The bottom-left figure shows a similar effect for growing wind velocity variance v_w. When the system grows more stochastic, chance constraint abidance is ensured by intervening at higher elevations. Finally, the bottom-right figure illustrates what happens when the chance constraint is combined with a Gaussian prior constraint on control. Increasing the control prior precision λ penalizes immediate correction. For low precisions (low penalty on control magnitude), the slope of the control law below the intervention threshold is equal to 1, and compensation is immediate. Control grows more robust with growing precision, at the cost of prolonged chance constraint violation.

4.3 Comparison Against a Goal-Driven Agent

In order to illustrate the difference in behavior between a chance- and a goal-driven ActInf agent, we compare the results of Fig. 6 with an ActInf agent where the chance constraint is replaced by a goal prior. We use the graphical model definition of Fig. 4 and define the auxiliary node function as a fixed prior $f_{x,k+1}(x_{k+1}) = \mathcal{N}(x_{k+1}|m_x, \vartheta_x)$ for all $t \leq k \leq t + T - 1$. We choose $m_x = 2$, and the variance $\vartheta_x = 0.18478$ such that the overflow of the safe region $1 - \int_{\mathcal{S}} f_{x,k+1}(x_{k+1})\,\mathrm{d}x_{k+1} \approx 0.01$ resembles the situation for $\epsilon = 0.01$. The message passing schedule then follows the definition of Fig. 5, where $\mu^{(i)}_{⑥}(x_{k+1})$ is no longer computed by Alg. 1 and propagates the fixed goal prior instead. Fig. 7 shows the resulting control law for $m_{w,t} = 0, T = 1, v_w = 0.2$ and varying λ.

The results of Fig. 7 show that the control for the goal-driven agent grows more robust with increasing λ – similar to the control law for the chance-driven agent (Fig. 6,

bottom right). For the smallest λ, the control law for the prior-driven agent resembles the corresponding control law for the chance-driven agent (dotted curve) only for elevations $x < 2$. For elevations $x > 2$, the goal-driven agent proposes downward corrections, while the chance-driven agent proposes no corrections. This comparison illustrates how a chance-driven agent avoids unnecessary interventions.

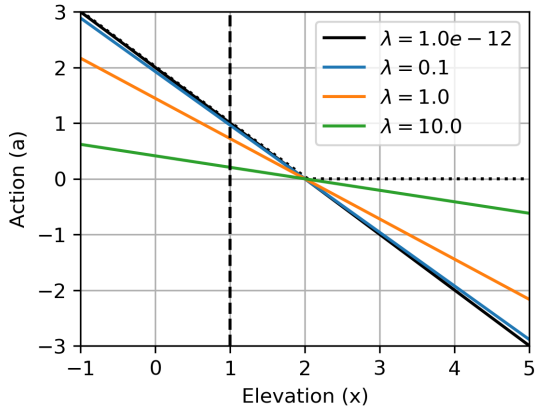

Figure 7: Slices of the control law for a goal-driven agent with $m_{w,t} = 0, T = 1, m_x = 2, \vartheta_x = 0.18478, v_w = 0.2$ with varying λ. The dashed vertical line indicates the minimal safe elevation. The black dotted curve represents the reference result ($\lambda = 10^{-12}$) for the chance-driven agent (Fig. 6, black curves).

4.4 Simulation Results

In this section we study an active inference agent in interaction with a simulated environment. The action-perception loop is based on (van de Laar and de Vries, 2019) and consists of four steps at every time t:

1. *Observe* the current agent elevation;

2. *Infer* a policy from the current elevation and the future expected wind velocities by chance-constrained message passing;

3. *Act* by selecting the first (current) action from the inferred policy;

4. *Execute* the selected action in the system and advance the time index by one.

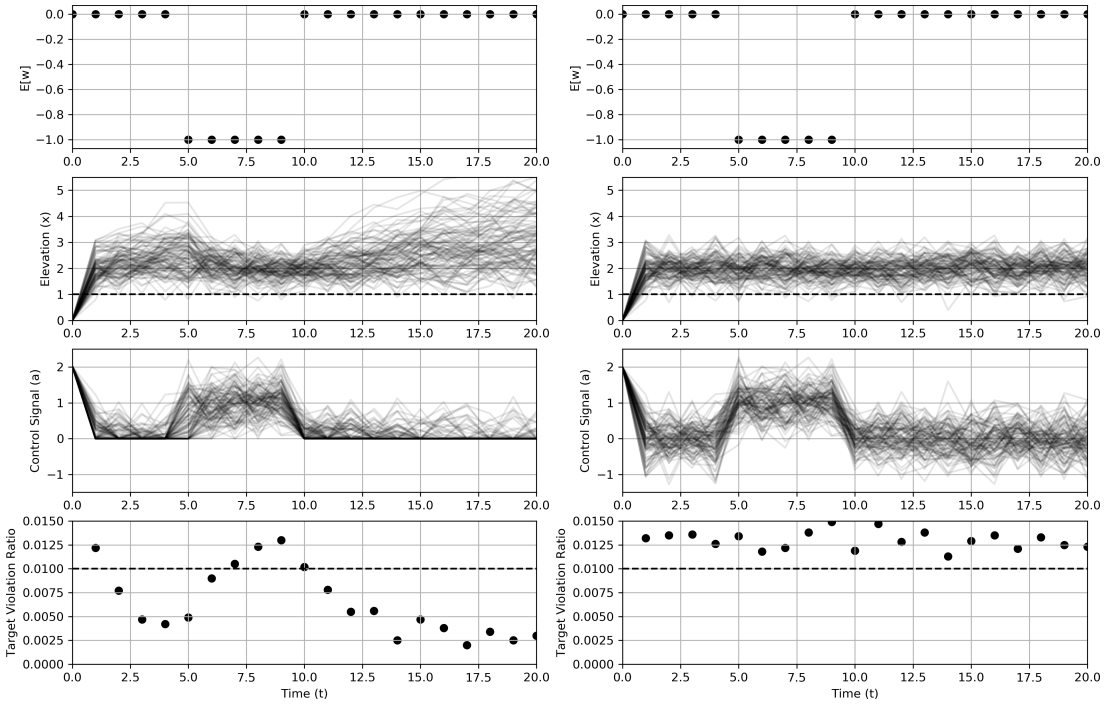

Figure 8: Results for ten thousand simulations with varying wind strength over time, and $T = 1, v_w = 0.2, \lambda = 10^{-12}$, for a chance-driven agent ($\epsilon = 0.01$, left), and a goal-driven agent ($m_x = 2, \vartheta_x = 0.18478$, right).

The results for ten thousand independent runs are plotted in Fig. 8 for a chance-driven agent (left) and a goal-driven agent (right). The first row of diagrams plots the expected wind velocity over time, which is identical for each run. The sampled wind velocity trajectories w_t do vary per run, under influence of the wind velocity variance v_w. For $5 \leq t < 10$ a downward draft attempts to push the drone below the minimal safe

elevation (dashed). The second row plots the drone elevation trajectory for a randomly selected subset of runs. Corresponding actions are plotted in the third row. The fourth row evaluates the relative number of runs that violate the safe-zone over time.

It can be seen that both agents undertake corrective actions in order to compensate for the downward wind. However, while the chance-driven agent (left) only proposes upward corrections below the intervention threshold, the goal-driven agent (right) proposes additional downward corrections above the threshold. Furthermore, it can be seen that the maximal empirical violation for the chance-constrained agent mostly remains below the chance constraint target violation probability of $\epsilon = 0.01$ (dashed), while the goal-driven agent systematically overshoots the target violation probability, i.e. violates the chance constraint. Compared to the chance-driven agent, the maximal empirical violations for the goal-driven agent are also larger. This effect can be explained in terms of the constrained beliefs. Namely, the chance-driven agent constrains the posterior beliefs, while the goal-driven agent imposes prior constraints on the model. Prior constraints may still be violated by the corresponding posterior beliefs, leading to more pronounced empirical violations.

5 Conclusions

In this paper, we formulated chance-constrained optimization of the Bethe free energy in terms of message passing on a factor graph. We showed that, in the factor graph representation of the generative model, chance constraints can be imposed by auxiliary factors that force (a specified portion of) the probability mass of the chance-constrained

beliefs inside a designated safe-zone. Message passing on the augmented graph, with the auxiliary factor-nodes included in the graph, then automatically balances the imposed chance constraints with additional (prior) constraints on the generative model. Chance constraints can thus be interpreted as modular click-on extensions to the generative model, similar to conventional factor-nodes (Loeliger et al., 2004), and can thus be used to complement message-passing formulations on generative neural models (Friston et al., 2017; van de Laar et al., 2018).

However, because the analytical result for the chance-constrained update includes an inherent discontinuity, direct application of this rule may still lead to message updates that grow prohibitively complex. To remedy this, we proposed an algorithm that approximates the resulting message with a Gaussian form. This algorithm offers a tractable formulation of chance-constrained message passing. The proposed message passing interpretation of chance constraints then vastly enhances the modularity and flexibility of chance-constrained inference, and can accelerate the search for workable models (Blei, 2014).

We demonstrated chance-constrained message passing in the context of active inference. We compared the simulated behavior of a chance-driven agent with a goal-driven agent, where the chance constraints are replaced by traditional prior beliefs on future outcomes. The results illustrate how the goal-driven agent continually proposes corrections, whereas the chance-driven agent seizes interventions above a threshold. Chance-constrained ActInf may thus avoid unnecessary interventions and reduce the cost of control.

The results for the chance-driven agent showed that, in the absence of additional

prior constraints, the empirical chance constraint violation ratio mostly remains below the pre-set target violation probability. An added prior constraint on controls robustifies control at the cost of prolonged chance constraint violation. Chance-constrained active inference thus weights all imposed constraints on the generative model, allowing e.g., for a trade-off between robust control and empirical chance constraint violation.

Acknowledgments

This work was supported, in part, by GN Hearing A/S and the Swedish Research Council (under Grants 2015-04011 and 2018-03701).

Appendix

A Calculus of Variations

The calculus of variations offers a principled method for optimizing functionals (a function of a function that returns a scalar). We follow (Engel and Dreizler, 2013) and consider the impact of a variation in a function $q(x), x \in \mathcal{X}$, on a functional $L[q]$. We define an infinitesimal variation of q by

$$\delta q \triangleq \beta\phi\,,$$

where $\beta \to 0$, and $\phi(x)$ is a continuous and differentiable "test" function.

The functional derivative $\delta L/\delta q$ relates a variation in q to a change in L, by (Parr,

1980):

$$\left.\frac{\mathrm{d}L[q+\beta\phi]}{\mathrm{d}\beta}\right|_{\beta=0} = \int \frac{\delta L}{\delta q}(x)\,\phi(x)\,\mathrm{d}x\,. \tag{20}$$

The procedure then becomes to apply the operations on the l.h.s. to L, and bring it into the form of the r.h.s., which allows us to identify the functional derivative $\delta L/\delta q$. The stationary points q^* are then obtained by setting $\delta L/\delta q \stackrel{!}{=} 0$ and solving for q.

B Proofs

B.1 Proof of Lemma 1

Application of (20) to (7) as a functional of q_b, yields

$$\left.\frac{\mathrm{d}L[q_b+\beta\phi_b]}{\mathrm{d}\beta}\right|_{\beta=0} = \int \phi_b(x_b)\left[\log\frac{q_b(x_b)}{f_b(x_b)} + 1 + \gamma_b - \sum_{i\in\mathcal{V}(b)}\zeta_{ib}(x_i)\right]\mathrm{d}x_b\,.$$

Identifying the functional derivative $\delta L[q_b]/\delta q_b$ and setting it to zero, we obtain

$$q_b^*(x_b) = f_b(x_b)\exp\left[\sum_{i\in\mathcal{V}(b)}\zeta_{ib}(x_i) - \gamma_b - 1\right]. \tag{21}$$

We now define $\mu(x_j) = \exp\zeta(x_j)$ and apply the normalization constraint, which recovers (8). □

B.2 Proof of Lemma 2

Application of (20) to (7) as a functional of q_j, yields

$$\left.\frac{\mathrm{d}L[q_j+\beta\phi_j]}{\mathrm{d}\beta}\right|_{\beta=0} = \int \phi_j(x_j)\Bigg[-(d_j-1)+\gamma_j$$
$$-(d_j-1)\log q_j(x_j) + \sum_{a\in\mathcal{F}(j)}\zeta_{ja}(x_j) + \eta_j g_j(x_j)\Bigg]\mathrm{d}x_j\,.$$

Identifying the functional derivative $\delta L[q_j]/\delta q_j$ and setting it to zero, yields

$$q_j^*(x_j) = \exp\left[\frac{1}{d_j - 1}\left(1 - d_j + \gamma_j + \sum_{a \in \mathcal{F}(j)} \zeta_{ja}(x_j) + \eta_j g_j(x_j)\right)\right], \tag{22}$$

which is the first expression for q_j^*.

We can obtain a second expression for q_j^* by applying the marginalization constraint to the result of Lemma 1. Substituting (8) in (5a),

$$\begin{aligned} q_j^*(x_j) &= \int q_b^*(x_b)\,\mathrm{d}x_{b\setminus j} \\ &= \frac{1}{Z_b}\mu_{jb}(x_j)\overbrace{\int f_b(x_b) \prod_{\substack{i \in \mathcal{V}(b) \\ i \neq j}} \mu_{ib}(x_i)\,\mathrm{d}x_{b\setminus j}}^{\mu_{bj}(x_j)}, \end{aligned} \tag{23}$$

where we identified a new quantity $\mu_{bj}(x_j)$ (note the reverse indexing).

Interestingly, the marginalization result of (23) not only holds for the specific factor b, but for all factors that neighbor j. Therefore, by symmetry, we can iterate the relation of (23) for all $c \in \mathcal{F}(j)$:

$$\prod_{c \in \mathcal{F}(j)} q_j^*(x_j) = \prod_{c \in \mathcal{F}(j)} \frac{1}{Z_c}\mu_{jc}(x_j)\mu_{cj}(x_j)\,.$$

We choose to exclude b itself from the iteration on both sides, and obtain

$$\prod_{\substack{c \in \mathcal{F}(j) \\ c \neq b}} q_j^*(x_j) = \prod_{\substack{c \in \mathcal{F}(j) \\ c \neq b}} \frac{1}{Z_c}\mu_{jc}(x_j)\mu_{cj}(x_j)\,. \tag{24}$$

We substitute (22) in the l.h.s. of (24), and note that the product on the l.h.s. now has $d_j - 1$ terms, and that neither of these terms depend on c. This allows us to remove the $d_j - 1$ terms from the exponent of (22), which yields

$$\exp(1 - d_j + \gamma_j + \eta_j g_j(x_j)) \prod_{a \in \mathcal{F}(j)} \mu_{ja}(x_j) = \prod_{\substack{c \in \mathcal{F}(j) \\ c \neq b}} \frac{1}{Z_c}\mu_{jc}(x_j)\mu_{cj}(x_j)\,.$$

Canceling duplicate terms and simplifying, we obtain an expression for μ_{jb} as identified in (23):

$$\mu_{jb}(x_j) \propto \exp(-\eta_j g_j(x_j)) \prod_{\substack{a \in \mathcal{F}(j) \\ a \neq b}} \mu_{aj}(x_j)\,. \tag{25}$$

Finally, substituting (25) back in (23) and re-normalizing, we recover (9). □

B.3 Proof of Theorem 1

We start from (9), and use the definitions of (11) to obtain

$$\int_{\mathcal{S}_j} q_j^*(x_j;\eta_j)\,\mathrm{d}x_j = \frac{\Phi_j^{(0)}\exp(-\eta_j)}{\Phi_j^{(0)}\exp(-\eta_j) - \Phi_j^{(0)} + 1}\,,$$

which leads to

$$\exp(-\eta_j^*) = \frac{(1-\epsilon)(1-\Phi_j^{(0)})}{\epsilon\Phi_j^{(0)}}\,.$$

We have now identified the η_j multiplier. Substituting this result back in (9) recovers (10), which expresses the *corrected* belief in terms of the *uncorrected* belief. □

B.4 Proof of Theorem 2

From (9), we express the uncorrected belief in terms of the messages

$$q_j^{(0)}(x_j) = \frac{1}{Z_j^{(0)}} \prod_{a \in \mathcal{F}(j)} \mu_{aj}(x_j)\,, \tag{26}$$

with $Z_j^{(0)} = Z_j(\eta_j = 0)$.

We now construct the augmented graph $\mathcal{G}' = (\mathcal{F}', \mathcal{V}, \mathcal{E}')$ according to (13), and define a message

$$\mu_{gj}(x_j) = f_g(x_j)\,, \tag{27}$$

with $f_g(x_j)$ as defined by (14).

Substituting (26) and (27) in (10) then yields the corrected belief in terms of the messages

$$q_j^*(x_j; \eta_j = \eta_j^*) = \frac{1}{Z_j^{(0)}} \mu_{gj}(x_j) \mu_{jg}(x_j)\,, \tag{28}$$

with

$$\mu_{jg}(x_j) = \prod_{\substack{a \in \mathcal{F}'(j) \\ a \neq g}} \mu_{aj}(x_j)\,. \tag{29}$$

The results of (27), (28) and (29) can be interpreted as belief propagation (2), (3) on the augmented graph $\mathcal{G}'$. □

References

Baltieri, M. and Buckley, C. L. (2018). The modularity of action and perception revisited using control theory and active inference. In *Artificial life conference proceedings*, pages 121–128. MIT Press.

Baltieri, M. and Buckley, C. L. (2019). Active Inference: Computational Models of Motor Control without Efference Copy. In *2019 Conf. on Cognitive Computational Neuroscience*.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Blackmore, L., Ono, M., and Williams, B. C. (2011). Chance-constrained optimal path planning with obstacles. *IEEE Transactions on Robotics*, 27(6):1080–1094.

Blei, D. M. (2014). Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1:203–232.

Borrelli, F., Bemporad, A., and Morari, M. (2017). *Predictive control for linear and hybrid systems*. Cambridge University Press.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Cox, M. and de Vries, B. (2018). Robust expectation propagation in factor graphs involving both continuous and binary variables. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 2583–2587. IEEE.

Cox, M., van de Laar, T. W., and de Vries, B. (2019). A factor graph approach to automated design of Bayesian signal processing algorithms. *International Journal of Approximate Reasoning*, 104:185–204.

Dauwels, J. (2007). On Variational Message Passing on Factor Graphs. In *IEEE Inter. Symp. on Information Theory*, pages 2546–2550.

de Vries, B. and Friston, K. J. (2017). A factor graph description of deep temporal active inference. *Frontiers in computational neuroscience*, 11:95.

Engel, E. and Dreizler, R. M. (2013). *Density functional theory*. Springer.

Fountas, Z., Sajid, N., Mediano, P. A., and Friston, K. (2020). Deep active inference agents using monte-carlo methods. *arXiv preprint arXiv:2006.04176*.

Friston, K., Rigoli, F., Ognibene, D., Mathys, C., Fitzgerald, T., and Pezzulo, G. (2015). Active inference and epistemic value. *Cognitive Neuroscience*, 6(4):187–214.

Friston, K. J. (2010). The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138.

Friston, K. J., Kilner, J., and Harrison, L. (2006). A free energy principle for the brain. *Journal of Physiology, Paris*, 100(1-3):70–87.

Friston, K. J., Parr, T., and de Vries, B. (2017). The graphical brain: belief propagation and active inference. *Network Neuroscience*, 1(4):381–414.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. (2014). Generative adversarial nets. In *NIPS*.

Heskes, T. (2003). Stable fixed points of loopy belief propagation are local minima of the bethe free energy. In *Advances in neural information processing systems*, pages 359–366.

Hoffmann, C. and Rostalski, P. (2017). Linear Optimal Control on Factor Graphs - a Message Passing Perspective. In *20th IFAC World Congress*, Toulouse, France.

Imohiosen, A., Watson, J., and Peters, J. (2020). Active inference or control as inference? a unifying view. In *1st International Workshop on Active Inference*.

Korl, S. (2005). *A factor graph approach to signal modelling, system identification and filtering*. ETH Zurich.

Loeliger, H.-A., Dauwels, J., Koch, V. M., and Korl, S. (2004). Signal processing with factor graphs: examples. In *First International Symposium on Control, Communications and Signal Processing, 2004.*, pages 571–574. IEEE.

Markovic, D., Stojic, H., Schwoebel, S., and Kiebel, S. J. (2021). An empirical evaluation of active inference in multi-armed bandits. *arXiv preprint arXiv:2101.08699*.

Mesbah, A. (2016). Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems Magazine*, 36(6):30–44.

Millidge, B., Tschantz, A., Seth, A. K., and Buckley, C. L. (2020). On the relationship between active inference and control as inference. In *1st International Workshop on Active Inference*.

Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369.

Parr, R. G. (1980). Density functional theory of atoms and molecules. In *Horizons of Quantum Chemistry*, pages 5–15. Springer.

Parr, T. and Friston, K. J. (2019). Generalised free energy and active inference. *Biological cybernetics*, 113(5):495–513.

Pearl, J. (1982). Reverend bayes on inference engines: A distributed hierarchical approach. In *Proc. of the Second AAAI Conference on Artificial Intelligence*, AAAI'82, page 133–136.

Ramstead, M. J. D., Badcock, P. B., and Friston, K. J. (2018). Answering Schrödinger's question: A free-energy formulation. *Physics of Life Reviews*.

Recht, B. (2019). A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2:253–279.

Sajid, N., Ball, P. J., Parr, T., and Friston, K. J. (2021). Active inference: demystified and compared. *Neural Computation*, 33(3):674–712.

Sallans, B. and Hinton, G. E. (2001). Using free energies to represent Q-values in a multiagent reinforcement learning task. In *Adv. in neural information process. systems*, pages 1075–1081.

Schwöbel, S., Kiebel, S., and Markovic, D. (2018). Active Inference, Belief Propagation, and the Bethe Approximation. *Neural Computation*, 30(9):2530–2567.

Tschantz, A., Millidge, B., Seth, A. K., and Buckley, C. L. (2020). Reinforcement learning through active inference. *arXiv preprint arXiv:2002.12636*.

Ueltzhöffer, K. (2018). Deep Active Inference. *Biological Cybernetics*, 112(6):547–573.

van de Laar, T. W. (2019). *Automated design of Bayesian signal processing algorithms*. Eindhoven University of Technology.

van de Laar, T. W., Cox, M., Şenöz, İ., Bocharov, I., and de Vries, B. (2018). Forneylab: a toolbox for biologically plausible free energy minimization in dynamic neural models. In *Conference on Complex Systems*.

van de Laar, T. W. and de Vries, B. (2019). Simulating Active Inference Processes by Message Passing. *Frontiers in Robotics and AI*, 6:20.

van de Laar, T. W., Özçelikkale, A., and Wymeersch, H. (2019). Application of the free energy principle to estimation and control. *arXiv preprint arXiv:1910.09823*.

Winn, J. and Bishop, C. M. (2005). Variational message passing. *Journal of Machine Learning Research*, 6(Apr):661–694.

Yedidia, J. S., Freeman, W., and Weiss, Y. (2005). Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312.

Yedidia, J. S., Freeman, W. T., Weiss, Y., et al. (2000). Generalized belief propagation. In *NIPS*, volume 13, pages 689–695.

Zhang, D., Wang, W., Fettweis, G., and Gao, X. (2017). Unifying message passing algorithms under the framework of constrained Bethe free energy minimization. *arXiv preprint arXiv:1703.10932*.