

Parallels between sensory and motor information processing

Emanuel Todorov
Department of Cognitive Science
University of California San Diego

Abstract

The computational problems solved by the sensory and motor systems appear very different: one has to do with inferring the state of the world given sensory data, the other with generating motor commands appropriate for given task goals. However recent mathematical developments summarized in this chapter show that these two problems are in many ways related. Therefore information processing in the sensory and motor systems may be more similar than previously thought – not only in terms of computations but also in terms of algorithms and neural representations. Here we explore these similarities as well as clarify some differences between the two systems.

Similarity between inference and control: an intuitive introduction

Consider a control problem where we want to achieve a certain goal at some point in time in the future – say, grasp a coffee cup within 1 sec. To achieve this goal, the motor system has to generate a sequence of muscle activations which result in joint torques which act on the musculo-skeletal plant in such a way that the fingers end up curled around the cup. Actually the motor system does not have to compute the entire sequence of muscle activations in advance. All it has to compute are the muscle activations right now, given the current state of the world (including the body) and some description of what the goal is. If the system is capable of performing this computation, then it will generate the resulting muscle activations, the clock will advance to the next point in time, and the computation will be repeated.

How can this control problem be interpreted as an inference problem? Instead of aiming for a goal in the future, imagine that the future is now and the goal has been achieved. More precisely, shift the time axis by 1 sec and create a fictive sensory measurement corresponding to the hand grasping the cup. The inference problem is now as follows: given that the fingers are around the cup and that the world was at a certain state 1 sec ago, infer the muscle activations which caused the observed state transition. As in the control problem, all that needs to be inferred are the muscle activations at a single point in time (1 sec ago); if this can be done then the clock will advance (to say 0.99 sec ago) and the computation will be repeated.

The above inference problem does not have a unique solution, because there are many sequences of muscle activations that could have caused the state transition we are trying to explain. Even at the final time the arm could be in many postures which all correspond to a successful grasp, thus the fictive measurement is incomplete. The same ill-posedness is present in the control problem and is known as motor redundancy (Bernstein 1967). Inference problems do not normally involve this kind of redundancy. Indeed the inference here is rather unusual: there is a period of time (1 sec in our example) when there are no sensory measurements, and the only available measurement at the end of the movement is incomplete. We could consider a different control problem which corresponds to a more usual inference problem involving complete sensory measurements. That control problem is one where we are given a detailed goal state at each point in time, i.e. a reference trajectory for all musculo-skeletal degrees of freedom, and have to generate muscle activations so as to force the plant to track this trajectory. When the latter control problem is mapped into an inference problem, the sequence of detailed goal states turns into a sequence of complete sensory measurements, thus eliminating redundancy. It is important to realize however that trajectory tracking represents only a small fraction of ecologically relevant behaviors (Todorov and Jordan 2002). Thus the natural control problem (which involves a large amount of redundancy) corresponds to an unnatural inference problem (where sensory data is very sparse) and vice versa. Inference is easier if complete sensory measurements are available at all times, and similarly, control is easier if detailed goal states are specified at all times.

This reasoning suggests that control is a harder problem than inference, at least in the temporal domain. Indeed inference in the absence of measurements is called prediction (except that here it is performed backwards in time), and prediction tends to be hard. On the other hand, redundancy makes it possible to be sloppy most of the time and still achieve the goal. This is because, even if the initial part of the movement somehow goes wrong, there is time later in the movement to observe what happened and take corrective action. The analog of this property in the inference domain is that long-term predictions tend to be inaccurate while short-term predictions (which correspond to motor commands close to the goal) are more accurate.

The above transformation from control to inference has been instantiated in formal models (Attias 2003). This is done by setting up a dynamic belief network (see below) which represents the states and actions at different points in time, treating the goal state as being observed, and performing Bayesian inference to find the actions. The shortcoming of this approach is that inference is performed over the product space of states and actions - which for a typical motor control problem is prohibitively large. In the rest of the chapter we will pursue a different approach, where the control problem will turn out to be equivalent to an inference problem involving only states. Actions will be defined implicitly as transitions between inferred consecutive states.

Now let us now ask the opposite question: can we start with an inference problem and transform it into a control problem? Consider the problem of estimating the current state of the world given our previous estimate and the current sensory measurement. One way to do this is to use a predictor-corrector method: combine the previous estimate with a model of one-step dynamics to obtain a prediction of the current state, and then correct the prediction so as to make it more compatible with the current measurement. The corrected estimate will achieve some trade-off between being close to the prediction and agreeing with the measurement. The corresponding control formulation is as follows. The entity being controlled (internally) is the state estimate. The control signal corresponds to the correction needed to achieve better agreement with the measurement. Suppose the control is chosen so as minimize a sum of two costs: an energy cost and an accuracy cost. The energy cost is minimal when there is no correction. The accuracy cost is minimal when the correction is complete. The control which minimizes the sum of these costs will lie somewhere in between the two extremes, thus achieving a similar trade-off as the predictor-corrector method.

We now see that, in the spatial domain, inference can be harder than control. This is because the estimator "controls" (i.e. corrects) all aspects of the estimated state. In the coffee-drinking example, the estimator may deal not only with the arm and the cup but also with the picture on the wall, the mountains we can see through the window, and many other things that have no relevance to motor actions. Note however that this implies a somewhat outdated view of perception in which all aspects of the sensory input are processed in parallel on equal footing. In reality perception may be geared towards serving the needs of the ongoing behavior - which in our example means ignoring the picture and the mountains and focusing on the arm and cup. In the latter view, inference and control have similar spatial complexity in terms of what needs to be computed (state estimate versus control signal). However the input to this computation (sensory data versus task goal) is always higher-dimensional for the sensory system.

The above transformation from estimation to control corresponds to the idea of minimum-energy filtering (Mortensen 1968), where estimation is formulated as a minimum-energy tracking problem and is solved using optimal control methods. The shortcoming of this approach is that it only yields point estimates, while a lot of evidence (see below) indicates that the brain computes probability distributions rather than point estimates. In the rest of the chapter we will pursue a different approach where the estimator computes the full Bayesian posterior.

To summarize the main points in this section, the similarities between control and estimation arise when task goals are associated with sensory measurements and control signals are associated with corrections to the state estimate. Our discussion was framed in the context of optimal control and optimal/Bayesian inference, which was not a coincidence. Indeed we will see below that optimality is the source of these similarities.

Duality of Bayesian inference and optimal control in isometric tasks

Here we provide a concrete example illustrating the duality between Bayesian inference and optimal control. Let \mathbf{u} be a vector of muscle activations, \mathbf{t} the resulting vector of joint torques, and M the matrix of moment arms which maps muscle forces (proportional to muscle activations under isometric conditions) to joint torques: $\mathbf{t} = M\mathbf{u}$. Isometric means that there is no movement. Since there are more muscles than joints, a desired torque \mathbf{t}^* can be achieved by infinitely many muscle activations \mathbf{u} . This is a manifestation of motor redundancy. In order to select one out of all possible \mathbf{u} the motor system needs some selection criterion. Suppose this criterion is to keep the sum of squared muscle activations as small as possible. Then \mathbf{u} can be found by minimizing the cost function:

$$(1) \quad \frac{1}{2} \|\mathbf{t}^* - M\mathbf{u}\|^2 + \frac{1}{2} r \|\mathbf{u}\|^2$$

The first term is an accuracy cost which is minimal when the desired torque is exactly achieved. The second term is an energy cost which is minimal when all muscle activations are zero. The parameter r determines the relative importance of the two. Quadratic cost functions are usually scaled by a factor of $1/2$ for convenience. We have shown (Todorov 2002) that the above cost function as well as more realistic versions of it predict the empirical phenomenon of cosine tuning, i.e. the fact that muscle activation varies with the cosine of the angle between the mechanical pulling direction of the muscle and the direction of end-effector force (Hoffman and Strick 1999).

We now turn to the corresponding inference problem, which involves a Gaussian prior over the elements of \mathbf{u} with mean 0 and variance $1/r$, and a fictitious measurement corresponding to goal achievement, namely $\mathbf{y} = \mathbf{t}^*$. Bayesian inference requires a generative model, i.e. a model of how the (noisy) sensory measurement was generated given the state. In this case the generative model is $\mathbf{y} = M\mathbf{u} + \boldsymbol{\varepsilon}$ where the elements of $\boldsymbol{\varepsilon}$ are Gaussian with mean 0 and variance 1. Applying Bayes rule (see below) and using the formula for a Gaussian, we obtain the posterior probability of \mathbf{u} given \mathbf{y} :

$$(2) \quad p(\mathbf{u} | \mathbf{y}) \propto p(\mathbf{y} | \mathbf{u})p(\mathbf{u}) \propto \exp\left(-\frac{1}{2} \|\mathbf{t}^* - M\mathbf{u}\|^2 - \frac{1}{2} r \|\mathbf{u}\|^2\right)$$

Thus the posterior probability in the inference problem (equation 2) coincides with the exponent of the negative cost in the control problem (equation 1), and in particular the most probable muscle activations coincide with the optimal muscle activations.

This completes our example of duality in isometric tasks. Although it is a simple example which does not involve state variables changing over time, it nevertheless illustrates a key idea used extensively later. The idea is that costs and probabilities are related by an exponential transformation. This is to be expected: costs add while

probabilities multiply, and it is the exponential transformation that turns sums into products. The same transformation shows up in other fields as well. In statistical mechanics for example, the energy of a given state and the probability of finding the system in that state at thermal equilibrium are related by the Gibbs distribution – which is the exponent of the negative energy.

We are now ready to develop a general form of duality between optimal control and optimal/Bayesian inference over time. To this end we will first review the concepts of optimality in sensory and motor processing, and note the similarities and differences between the two formalisms. This analysis will then indicate how the control problem should be phrased so as to become mathematically equivalent to Bayesian inference.

Optimality in sensory and motor processing

While all aspects of neural function have evolved to produce behavior beneficial to the organism, the evolutionary pressures on real-time sensory and motor processing may have been particularly strong and direct because of the crucial role such processing plays in getting food to the mouth, escaping predators and generally keeping the organism alive. It is then not surprising that the underlying neural mechanisms perform about as well as any information processing system subject to the same constraints could perform – in other words, near-optimally. Indeed optimality is becoming the theoretical framework of choice of studying both sensory and motor systems (Todorov 2004, Kording and Wolpert 2006, Doya et al. 2007).

In the sensory domain optimality corresponds to Bayesian inference. In the simplest setting it involves three probability distributions over the (relevant) state of world: the prior, the likelihood and the posterior. They are related according to Bayes rule:

$$(3) \quad \textit{posterior}(x) \propto \textit{likelihood}(y | x) \textit{prior}(x)$$

The prior summarizes everything we know about the state of the world before observing the measurement. The likelihood (which formalizes the generative model) is the probability of measurement y being generated when the world is in state x . The posterior summarizes everything we know after the measurement is taken into account. If there are multiple independent measurements, the right hand side of equation (3) contains the product of the corresponding likelihoods. The latter setting is used in models of cue integration, where subjects are presented with two (often incompatible) sensory cues and asked to estimate some property of the world. Such experiments have provided the simplest and perhaps most compelling evidence that perception relies on Bayesian inference (e.g. Ernst and Banks 2002). The probability distributions used in these studies are typically Gaussians.

Unlike the static nature of many cue-integration experiments, sensory processing in the real world takes place in time and requires integration of measurements obtained at different points in time. This is called recursive estimation or filtering. The basic update scheme applied at each point in time has the predictor-corrector form:

$$(4) \quad p(x) \propto l(y|x) \sum_{x_{prev}} d(x|x_{prev})p(x_{prev})$$

Here $p(x)$ is the posterior at the current state, $p(x_{prev})$ is the posterior at the previous state (which we have already computed at the previous time step), $l(y|x)$ is the likelihood function and $d(x|x_{prev})$ is the stochastic one-step dynamics of the world. When estimating the state of the body, the dynamics d will also depend on the control signal available to the sensory system in the form of an efference copy. The product of d and p , which is being summed over, is the joint probability of x and x_{prev} . The sum marginalizes out x_{prev} and yields a prediction (or prior) over x . In this way the posterior at one point in time is used to compute the prior at the next point in time. The multiplication by the likelihood l is the sensory-based correction discussed earlier.

A number of experimental findings support the notion of Bayesian inference over time (Wolpert et al. 1995, Kording and Wolpert 2004, Saunders and Knill 2004). These studies typically use arm movements, not so much for the purpose of studying the motor system but as a continuous readout of perception. Such studies demonstrate that subjects take into account multiple sources of information over time (visual and proprioceptive, along with internal predictions) and rely on that information to guide movements. As in cue integration, the probability distributions assumed here are typically Gaussian. When the dynamics are linear and all noise is Gaussian, the posterior is also Gaussian and can be computed using the Kalman filter.

There is a graphical representation of Bayesian inference problems (**Figure 1a**) known as a graphical model or a belief network (dynamic belief network when time is involved). This representation is very popular in statistics and machine learning (Pearl 1988). Belief networks help understand the mathematical models intuitively, and will also be useful later in clarifying the relationship between estimation and control. To avoid confusion keep in mind that unlike neural networks, the nodes in belief networks do not correspond to neurons and the arrows do not correspond to synaptic connections. Instead the nodes correspond to collections of random variables, whose probabilities are presumably represented by populations of neurons. The arrows strictly speaking encode conditional probabilities, but in reality they often correspond to the causal relations in the world, as illustrated in **Figure 1a**. We only show part of the network containing the states of the world at two consecutive points in time as well as the corresponding sensory measurements/inputs. Filled gray circles denote variables whose values are observed and which therefore contribute a likelihood function. Empty circles denote variables whose values are to be inferred. The forward arrows encode the

stochastic dynamics of the world, i.e. the one-step transition probability d . The downward arrows encode how sensory measurements are generated as a function of world states. This generative model may incorporate a model of optics in vision or a model of acoustics in audition, plus a model of sensory transduction in the corresponding modality. One can think of perception as a computational process which inverts the generative model in a probabilistic sense (this idea goes back to Helmholtz).

Optimality has also been applied in motor control, perhaps even more extensively than in perception. This may be because, apart from its general appeal as an organizing principle, optimality appears to be the right way to resolve redundancy. There is a wealth of experimental data (for reviews see Todorov 2004, Kording and Wolpert 2006) suggesting that the motor system generates actions that maximize task performance or utility. Optimal control models have accounted in parsimonious ways for numerous features of motor behavior on the levels of kinematics, dynamics and muscle activity. There are two general approaches: open-loop control and closed-loop control. Open-loop control pre-computes the entire sequence of motor commands from now until the goal is achieved, while closed-loop control (or feedback control) only computes the current motor command given the current state estimate, and then uses information about the next state to compute the next command. Since movements are under continuous sensory guidance, the latter type of model corresponds more closely to what the brain does. Although optimal feedback controllers are harder to construct, we now have efficient algorithms and fast computers that enable us to explore such models.

Here is how optimal feedback control works in a nutshell. Define an instantaneous cost which accumulates over time and yields a cumulative cost. The instantaneous cost is usually a sum of a control cost $r(u)$ which encourages energetic efficiency, and a state cost $q(x)$ which encourages accuracy, or more generally, getting to desirable states and avoiding undesirable states. Also define the one-step stochastic dynamics $d_u(x_{next} | x)$ which is similar to the one-step transition probability in Bayesian inference except that it now depends on the control u explicitly. The objective is to find an optimal control law, i.e. a mapping from states to controls which minimizes the expected cumulative cost. This computation is facilitated by the optimal cost-to-go function $v(x)$, defined as the cost expected to accumulate if the plant is initialized at state x and is controlled optimally thereafter. The optimal cost-to-go function plays a key role because it summarizes all relevant information about the future and allows us to compute the optimal control at the present time using greedy optimization without look-ahead. This function is the unique solution to the Bellman equation:

$$(5) \quad v(x) = \min_u \left[q(x) + r(u) + \sum_{x_{next}} d_u(x_{next} | x) v(x_{next}) \right]$$

The control u which achieves the minimum is the optimal control at the current state x . This equation is quite intuitive: it says that the optimal cost-to-go can be broken down into the instantaneous cost incurred at the current state when applying the optimal control, plus the optimal cost-to-go for the movement originating at the next state. The expectation (sum over x_{next}) is needed because the dynamics are stochastic and the next state is known only in a probabilistic sense. Equation (5) can always be solved numerically using dynamic programming, which involves computing the minimum over u and assigning it to $v(x)$ for each x and each time step. For large problems however this computation is often intractable.

One special case where equation (5) can be solved efficiently is the case of linear dynamics, Gaussian noise and quadratic costs. In such problems (known as LQG) the optimal cost-to-go is quadratic and can be computed with a method very similar to the Kalman filter. Note that a Gaussian is the exponent of a quadratic function, the product of two Gaussians is the exponent of the sum of the corresponding quadratics, and a sum of quadratics is again a quadratic (as illustrated in the isometric task example). So both the Kalman filter and the LQG optimal controller are based on manipulating quadratics; indeed the underlying equations are identical. This duality was discovered by Kalman (1960) and was the first indication that optimal estimation and optimal control are closely related. We recently showed (Todorov 2008) that Kalman's duality is special to the LQG setting and does not generalize. However there exists another form of duality that does generalize. It was developed by Mitter and Newton (2003) in continuous time and Todorov (2006, 2008) in discrete time. The two developments are technically quite different and yet yield related results. Our presentation in the next section will use the discrete-time version which is more intuitive and also turns out to be more general. The continuous-time version can be obtained as a special case, by assuming Gaussian noise and taking a certain limit.

General duality of Bayesian inference and optimal control

Comparing equations (4) and (5) we can already see a similarity between optimal control and Bayesian inference. The state cost q and the likelihood l are related in the sense that they both inject new information about x in each step of the recursive process. The optimal cost-to-go v and the posterior p are related in the sense that they both accumulate information about x over time. The one-step dynamics d are present in both control and estimation. One equation involves sums while the other involves products, but sums can be turned into products by the exponential transformation. Yet we also see a difference: while equation (4) specifies the posterior p directly via an explicit formula, equation (5) specifies the optimal cost-to-go v indirectly as the solution to an unsolved optimization problem. Neither the control cost $r(u)$ nor the dependence of d_u on u have analogs in (4), suggesting that whether or not the optimal control problem is dual to a Bayesian inference problem will depend on how we define $r(u)$ and d_u .

In order to establish a general duality, we will define the control signal as a probability distribution over possible next states. This is unusual but in retrospect natural. What controls do is affect the plant dynamics. So we can characterize them directly in terms of how they affect the plant dynamics. For a stochastic plant this characterization takes the form of a probability distribution $u(x_{next})$. Thus the one-step dynamics are simply:

$$(6) \quad d_u(x_{next} | x) = u(x_{next})$$

According to this definition the controller has the power to impose on the plant whatever dynamics it wishes. We will restrict this power somewhat, by defining the passive dynamics $d(x_{next} | x)$ in the same way as in the inference problem, and allowing u to be non-zero only if d is non-zero. The passive dynamics capture the effects of gravity, interaction forces and motor noise. The above restriction means that the control signals can cause only those state transitions which could have occurred by accident, i.e. the noise and the controls are restricted to act in the same subspace. For musculo-skeletal plants where the controls correspond to muscle activations, the noise model should be restricted to muscle space and should not be allowed to act directly on, say, arm position (which would be physically unrealistic anyway). The above restriction also means that we cannot model external perturbations acting on objects of interest. Such perturbations are often used experimentally to probe the visual feedback control laws, however they are uncommon in the real world.

Having a model of passive or uncontrolled dynamics allows us to define the control cost in a natural way. Intuitively such a cost should measure how large the control signals are. Larger control signals have larger effects on the plant dynamics, i.e. they push the plant further away from its passive dynamics. This suggests a control cost which measures the difference between the probability distributions $u(x_{next})$ and $d(x_{next} | x)$. Differences between probability distributions are most commonly measured using Kullback-Liebler (KL) divergence, thus the control cost will be defined as:

$$(7) \quad r(u) = \text{KL}(u, d) = \sum_{x_{next}} u(x_{next}) \log \frac{u(x_{next})}{d(x_{next} | x)}$$

Definitions (6) and (7) yield a family of control problems which still satisfy the Bellman equation (5) but have additional structure that can be exploited. A control problem in our family is defined by specifying the state cost $q(x)$ and passive dynamics $d(x_{next} | x)$. Once q and d are given, we can substitute (6) and (7) into (5) and observe that the minimization with respect to u can be performed analytically, due to properties of the KL divergence. We omit the derivation and summarize the results. The results are expressed most conveniently in terms of a desirability function defined as:

$$(8) \quad z(x) = \exp(-v(x))$$

When the optimal cost-to-go $v(x)$ is small the function $z(x)$ is large, thus the term "desirability". It also rhymes with probability, which is appropriate because z will turn out to behave like a probability distribution. It can now be shown that the optimal control (i.e. the optimal next-state probability distribution) at state x is:

$$(9) \quad u^*(x_{next}) \propto d(x_{next} | x) z(x_{next})$$

This form of control is illustrated in **Figure 2**. Given the current state x , we multiply the one-step passive dynamics $d(x_{next} | x)$ by the desirability $z(x_{next})$, normalize to obtain a proper probability distribution $u^*(x_{next})$, and sample the next state from it. Note that multiplication by the desirability z has the effect of shifting the passive dynamics d towards more desirable states.

We still need to compute z . This is done by substituting the optimal control (9) into the Bellman equation (5), dropping the min operator, and exponentiating so as to obtain an update for z rather than v . The resulting update is:

$$(10) \quad z(x) = \exp(-q(x)) \sum_{x_{next}} d(x_{next} | x) z(x_{next})$$

The similarity with Bayesian inference (equation 4) is now obvious: the desirability z corresponds to the posterior probability p , the exponentiated state cost $\exp(-q)$ corresponds to the likelihood l , and the one-step transition probability d plays the same role in both cases. The only difference is that z is updated backward in time while p is updated forward in time. This is because control is about the future while inference is normally about the past. However if we construct the inference problem as outlined earlier, i.e. provide fictive sensory measurements in the future, then Bayesian inference and optimal control become mathematically equivalent.

Optimal control can then be represented with the belief network shown in **Figure 1b**. This network is drawn upside-down so as to highlight an important difference between inference and control. In inference the known quantities (sensory measurements) are near the periphery, while in control the known quantities (task goals) are deep inside the CNS. Conversely, the outputs of the sensory system are deep inside the CNS while the outputs of the motor system are close to the periphery. Both diagrams are oriented so that the CNS is up and the periphery is down. One network involves "world states" and the other "plant states", however these two notions of state may actually be similar. This is because the motor system has to represent not only the state of the body (plant) but also all relevant aspects of the state of the environment, while the sensory system may not represent all aspects of the world but instead focus on those relevant to the ongoing behavior.

Thus far estimation and control were discussed separately, while in the brain they are performed simultaneously. Can we think of both sensory and motor processing as being part of the same computation? This can be done by combining the two belief networks in **Figure 1** and performing Bayesian inference on the composite network. The sensory measurements in the past would be real while those in the future would be fictive. The probability over past states would encode what we believe has already happened, while the probability over future states would encode what we believe will happen if we act optimally. The control problem was set up in such a way that having a prediction about the future is equivalent to specifying a control signal which turns this prediction into reality. Note that such a unified computational scheme would be only approximately optimal, because the controller here was designed with the assumption that the current state is known with certainty. This form of approximation tends to be quite accurate and is often used in control engineering (it is known as certainty-equivalence control). The approximation fails when the uncertainty about the state affects the optimal actions – as in tasks which involve trade-offs between exploration and goal achievement. In that case the state in the control problem can be augmented with the uncertainty about the state in the inference problem (Simpkins et al. 2008).

To summarize this section, we described a family of optimal control problems that are mathematically equivalent (i.e. dual) to Bayesian inference. The state costs and the passive dynamics in our formulation are completely general and can be defined in whatever way is necessary. The only constraints are that the control signals must act in the same subspace as the passive dynamics, and the control cost must equal the KL divergence between the controlled and passive dynamics. In the continuous-time limit this control cost reduces to the familiar quadratic energy cost. Control problems which do not satisfy the above constraints do not seem to have exact duals, yet they can often be approximated with problems which satisfy the constraints (Todorov 2006).

Intermediate representations: sensory features and motor synergies

On the system level, sensory processing performs a transformation from sensory inputs to inferred states, while motor processing performs a transformation from task goals to motor commands. However neither transformation is performed monolithically by a single brain area. Instead there are multiple brain areas involved, and most of them use neural representations which correspond to neither the input nor the output of the overall computation but to something in between. Similarly, if we analyze a typical computer program, we will notice that most of the variables declared in it are internal variables representing intermediate results. How do such intermediate representations relate to the overall computation? One way to address (or rather, avoid) this question is the Computer Science way, adapted to Neuroscience by Marr (1982). There one makes a strict distinction between the problem being solved, the algorithm for solving it, and the implementation of the algorithm in software or hardware or wetware. While this

approach has many merits, a significant drawback is that computational-level analyses tell us little about the underlying neural representations and the interactions among them. In this section we outline a somewhat different approach which enables us to relate intermediate representations to the overall computation more directly. We will first develop the idea for sensory systems and then see how it applies to motor systems.

Intermediate sensory representations are often called "features" and thought to be features of the "stimulus". But what is a stimulus? Is it the sensory input, or is it the relevant aspect of the world reflected in the input? If features are defined as functions of the sensory input, then they do not belong on the computational level and we are back to Marr's strict separation. Suppose instead that features are statements about the state of the world. For example, suppose the activity of an "edge detector" in primary visual cortex is not a statement about the presence of an edge in the retinal image, but a statement about the state of the world which caused the retinal image to contain an edge. In this view features are part of the generative model (**Figure 3a**). The sensory input is modeled as a (probabilistic) function of the features instead of the other way around. Bayesian inference can be applied to such a hierarchical generative model without modification. One prediction is that at every intermediate level of sensory processing there will be both bottom-up and top-down effects. This is because the probability of any variable in a belief network generally depends on all other variables. The beauty of this approach is that different levels of the generative model can be instantiated in different brain areas, and as long as the communication within and between areas corresponds to Bayesian inference, the entire distributed system will perform a single computation, using perhaps a single algorithm (see below) which operates in parallel on multiple representations.

Let us now apply the same idea to the motor system. The closest analog of a sensory feature in motor control is the notion of a motor synergy. It corresponds to some intermediate representation which is more abstract than the full musculo-skeletal state but more detailed than the task goal. By analogy to the sensory system, we propose that synergies are part of a hierarchical generative model – which in the case of the motor system is a (probabilistic) mapping from plant states to task goals. Synergies are often thought to be related to motor commands rather than plant states, however recall that in our formulation motor commands are implicit, and can be recovered from the probability of the future states under the optimal controls. As illustrated in **Figure 3b**, synergies can be used for both spatial and temporal abstraction. For example, a synergy might be a statement about the shape of the fingertip trajectory over a short period of time. The fact that the synergy corresponds to a period of time and not a single point in time yields temporal abstraction. The fact that the synergy corresponds to only some aspects of the state of the plant and not the entire state (e.g. it does not specify all the joint angles but only the fingertip position) yields spatial abstraction. Different forms of spatial and temporal abstraction have played an important role in designing automatic controllers for complex tasks, suggesting that the brain may also rely on such tools.

Thus intermediate representations in both sensory and motor systems can be thought of as being part of hierarchical generative models. One might ask, however, what is the point of having such representations when generative models can be built without them? For example, given the full state of the arm we can directly compute where the fingertips are, without the help of motor synergies. Similarly, we can directly compute the retinal image resulting from a given configuration of 3D objects and light sources, without relying on sensory features (this is what computer graphics does). Indeed intermediate representations are not only unnecessary to build generative models but may even complicate the construction of such models. However, the goal of both the sensory and motor systems is not so much to build generative models but rather to invert them. The inversion is the harder problem, and is also the problem that has to be solved in real time. Intermediate representations are likely to facilitate this inversion – by providing various forms of abstraction and enabling the inference algorithm to construct the final answer in manageable pieces. Thus intermediate representations may exist not for the sake of representation but because they facilitate the computation.

One might also ask, where do intermediate representations come from? In sensory systems, it has been shown that unsupervised learning applied to collections of natural sensory inputs can recover the features observed experimentally. The most notable examples come from the visual system (Olshausen and Field 1996) although the approach has also been applied successfully to the auditory system (Lewicki 2002). Unsupervised learning looks for statistical regularities in high-dimensional data. Traditional unsupervised learning methods like principal components analysis (PCA) reduce the dimensionality of the data. In contrast, the forms of unsupervised learning thought to be used by sensory systems tend to increase dimensionality, i.e. they form over-complete (and sparse) representations. This may seem counterproductive, however it resonates well with recent computational approaches where increasing dimensionality simplifies computation. Support vector machines and kernel methods in general are based on this idea (Scholkopf and Smola 2001). Liquid state machines in Neuroscience have the same flavor (Maass et al. 2002).

Unsupervised learning has also been applied in motor control to extract candidate synergies (D'Avella et al. 2003, Santello et al. 1998). However the situation here is qualitatively different. While in sensory systems unsupervised learning is applied to sensory data available to the brain during learning/development, in motor systems it is applied to movement data available to the brain only after it has mastered the motor task. If we agree that appropriate synergies must exist before successful movements can be generated in a given task, then the brain cannot learn those synergies from successful movements. In other words, the unsupervised learning methods used by motor control researchers are not a feasible model of learning by the motor system. A feasible model should learn based on information available at the time of learning. The one thing that is always available is the input – which in the case of the motor system corresponds to the task goals. Thus the analog of learning features from sensory inputs would be

learning synergies from task goals. Unfortunately the task goals are not directly accessible to an external observer, so the application of unsupervised learning as outlined here is not easy, yet we suspect it is worth pursuing.

Another insight into motor synergies that comes from the analogy with sensory systems is the number of synergies. It is widely believed that motor synergies serve the purpose of dimensionality reduction; indeed they are usually defined as the outputs of dimensionality reduction algorithms. However, as discussed earlier, dimensionality expansion rather than reduction may be more beneficial in terms of simplifying computation. Furthermore, the number of different neural activation patterns in, say, primary motor cortex, greatly exceeds the number of musculo-skeletal degrees of freedom. If we agree to think of neural activity in motor areas as representing synergies, in the same way we think of neural activity in sensory areas as representing features, the dimensionality-expansion point of view becomes unavoidable.

This view represents a significant departure from the established thinking about motor synergies, and may at first seem incompatible with the evidence that large amounts of variance (in movement kinematics or EMGs or isometric forces) can be explained by small numbers of components. How can behavioral evidence for dimensionality reduction be reconciled with intermediate representations performing dimensionality expansion? One answer comes from our work on optimal control (Todorov and Jordan 2002, Todorov 2004), where we showed that an optimally-controlled redundant system will exhibit signs of dimensionality reduction regardless of how the controller is implemented. If that is the case, and the motor system is good at approximating optimal controllers, then a lot of the dimensionality-reduction results currently taken as evidence for synergies may instead be indirect evidence for optimality. The over-complete intermediate representations which we propose to call synergies may be the mechanism that enables the motor system to perform near-optimally.

Algorithms for learning and online computation

Bayesian inference and optimal control are of interest in many fields (e.g. Statistics, Computer Science, Signal Processing, Control Engineering, Economics). Consequently many algorithms have been developed. While none of them can yet compete with biological sensory and motor systems on complex real-world problems, this repository of algorithmic knowledge is an important source of insights into what the brain might be doing. We refer the reader to (Doya et al. 2007) for an extended discussion. Here we only make a few points relevant to this chapter.

One class of Bayesian inference algorithms, known as belief propagation (Pearl 1988), are reminiscent of computation in recurrent neural networks except that the messages being exchanged are probability distributions (presumably encoded by populations of

neurons). The analog in optimal control is dynamic programming – which for the family of control problems described above is reduced to belief propagation. An important corollary of the estimation-control duality is that sensory population codes thought to represent probability distributions (Doya et al. 2007) can be equally useful in motor control. Both belief propagation and dynamic programming are global methods in the sense that they aim to compute functions over the entire state space. For large problems this is unlikely to be doable in real time. In the control domain this point is well appreciated; indeed dynamic programming is normally applied offline so as to pre-compute/learn the optimal control law. The latter is then used online to generate motor commands as a function of plant states and goal parameters (e.g. target positions). The equivalent in Bayesian inference would be to learn a direct mapping from sensory inputs to state estimates – which is not how people usually think about inference. There may be several reasons for this: (i) the input to the sensory system is so high dimensional that learning such a mapping is infeasible; (ii) inference is an easier problem than control (recall our discussion of redundancy) and so the computation is easier to perform online; (iii) the brain actually learns direct mappings from sensory inputs to estimated states, but this is not yet reflected in most Bayesian inference algorithms. One exception here is the Helmholtz machine (Dayan et al. 1995), which is a belief network augmented with a mechanism for learning the direct mapping discussed above (i.e. the inverse of the generative model).

Regardless of whether and how much of the transformation is learned, it is clear that a lot of processing takes place in real time in both the sensory and motor systems. There is a simple way to combine the advantages of learning and online computation: learn a global but approximate transformation, use it online for initialization, and then apply an online algorithm to refine the solution locally around the current state. Locally, probabilities and costs can be approximated by simplified models (e.g. Gaussians and quadratics) which afford faster computation. Such local approximation methods are available in both estimation (the extended Kalman filter) and control (iterative LQG or differential dynamic programming). In the case of control, local improvement requires unfolding the time axis up to a certain horizon. This is known as receding-horizon control. For our family of control problems it is illustrated with the belief network in **Figure 3b**. If the motor system relies on such methods, we should expect to find neurons coding the state of the plant at multiple points in time in the future. Indeed it has often been noted (e.g. Kalaska et al. 1998) that the latency between neural firing and motor behavior has a broad distribution, and on average is substantially longer than what one would expect from conduction latencies alone. Such data can tell us how much unfolding is taking place in the motor system. The answer is on the order of 200 msec for reaching movements, although more complex tasks (for which we do not have data) may require unfolding over longer time horizons.

Bayesian inference problems can also be solved using sampling methods. An example is Gibbs sampling, which works by choosing a node to be updated, re-sampling its value

from the conditional probability given the current values of its neighbors, choosing another node etc. After a "burn-in" period, the samples generated in this way match the correct Bayesian posterior. The estimation-control duality makes it possible to apply sampling algorithms to optimal control problems as well. Sampling algorithms have not been seriously considered as models of brain function, but perhaps they should be, for several reasons. First, these are the only algorithms that are actually guaranteed to solve the problem (even though it may take a long time). All other algorithms when applied to continuous state variables require function approximation, and as a result may never converge or converge to the wrong answer. Second, sampling is inherently parallel. Other algorithms can be parallelized but not to the same extent. This is an important consideration given the staggering number of neurons in the brain. Third, sampling is inherently stochastic. Implementing it in a deterministic computer requires a pseudo-random number generator. The brain has internal sources of noise (e.g. failures of synaptic transmission) which could be used as random number generators - implying that neural noise may be a feature rather than a nuisance.

In summary, a range of algorithms for Bayesian inference and optimal control have been developed in multiple fields. Furthermore, the estimation-control duality makes it possible to take estimation algorithms and apply them to control problems and vice versa. Such algorithms are very relevant to Neuroscience because they solve the same problems that the sensory and motor systems appear to be solving. Which of these algorithms resemble the ones used by the brain is not yet clear (but see Doya et al. 2007). Algorithmic issues have generally received limited attention in Neuroscience, perhaps because they are hard to address experimentally. This is in contrast with system-level computations which can be addressed using behavioral data, and neural representations which can be addressed using single neuron data. Indeed a lot is already known about both system-level computations and neural representations, in both the sensory and motor systems. This knowledge imposes strong constraints, which, in conjunction with algorithmic insights from multiple fields, may soon enable us to go after the brain's algorithms in a systematic way.

Acknowledgements

This work was supported by the US National Science Foundation.

References

- Attias, H. (2003) Planning by probabilistic inference. In *International Conference on Artificial Intelligence and Statistics*.
- Bernstein, N. (1967) *The Coordination and Regulation of Movements*. Pergamon, Oxford.
- Dayan, P., Hinton, G., Neal, R. and Zemel, R. (1995) The Helmholtz Machine. *Neural Computation*, 7, 1022-1037.
- D'Avella, A., Saltiel, P. and Bizzi, E. (2003) Combinations of muscle synergies in the construction of a natural motor behavior. *Nature Neuroscience* 6: 300-308.
- Doya, K., Ishii, S., Pouget, A. and Rao, R. (2007) *Bayesian Brain: Probabilistic Approaches to Neural Coding*. MIT Press, Cambridge MA.
- Ernst, M. and Banks, M. (2002) Humans integrate visual and haptic information in a statistically optimal fashion. *Nature* 415: 429-433.
- Hoffman, D. and Strick, P. (1999) Step-tracking movements of the wrist. IV. Muscle activity associated with movements in different directions. *J Neurophysiol.* 81: 319-333.
- Kalman, R. (1960) A new approach to linear filtering and prediction problems. *ASME Transactions J Basic Engineering* 82:35-45.
- Kalaska, J., Sergio, L. and Cisek, P. (1998) Cortical control of whole-arm motor tasks. In *Sensory Guidance of Movement* 176-201, Glickstein, M. ed., Wiley, Chichester, UK.
- Kording, K. and Wolpert, D. (2004) Bayesian integration in sensorimotor learning. *Nature* 427: 244-247.
- Kording, K. and Wolpert, D. (2006) Bayesian decision theory in sensorimotor control. *Trends in Cognitive Sciences* 10: 320-326.
- Lewicki, M. (2002) Efficient coding of natural sounds. *Nature Neuroscience* 5: 356-363.
- Maass, W., Natschlager, T. and Markram, H. (2002) Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Computation* 14: 2531-60.
- Marr, D. (1982) *Vision*. Freeman, San Francisco.

- Mitter, S. and Newton, N. (2003) A variational approach to nonlinear estimation. *SIAM J Control and Optimization* 42: 1813-1833.
- Mortensen, R. (1968) Maximum-likelihood recursive nonlinear filtering. *J Optimization Theory and Applications* 2: 386-394.
- Olshausen, B. and Field, D. (1996) Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* 381: 607-609.
- Pearl, J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francisco.
- Santello, M., Flanders, M. and Soechting, J. (1998) Postural hand synergies for tool use. *J Neuroscience* 18: 10105-10115.
- Saunders, J. and Knill, D. (2004) Visual feedback control of hand movements. *J Neuroscience* 24: 3223-3234.
- Scholkopf, B. and Smola, A. (2001) *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge MA.
- Simpkins, A., de Callafon, R. and Todorov, E. (2008) Optimal trade-off between exploration and exploitation. In *American Control Conference*.
- Todorov, E. (2002) Cosine tuning minimizes motor errors. *Neural Computation* 14: 1233-1260.
- Todorov, E. (2004) Optimality principles in sensorimotor control. *Nature Neuroscience* 7: 907-915.
- Todorov, E. (2006) Linearly-solvable Markov decision problems. In *Advances in Neural Information Processing Systems*.
- Todorov, E. (2008) General duality between optimal control and estimation. In *IEEE Conference on Decision and Control*.
- Todorov, E. and Jordan, M. (2002) Optimal feedback control as a theory of motor coordination. *Nature Neuroscience* 5: 1226-1235.
- Wolpert, D., Gharahmani, Z. and Jordan, M. (1995) An internal model for sensorimotor integration. *Science* 269: 1880-1882.

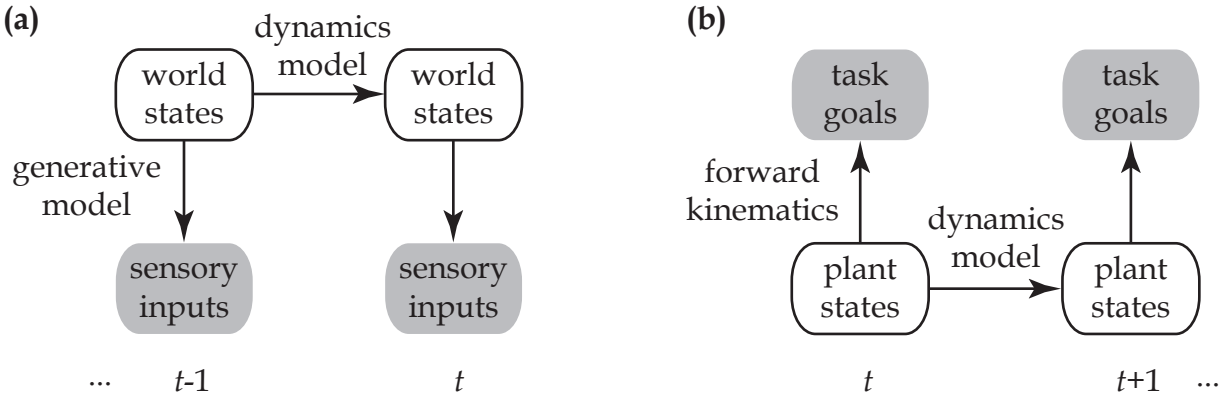


Figure 1 - belief networks for Bayesian inference and optimal control.

(a) - Shaded nodes correspond to observed quantities while open nodes correspond to random variables whose (marginal) probabilities are to be computed. The dynamics model is the probability distribution of the next state given the current state. The generative model is the probability distribution of the sensory input given the state.

(b) - The optimal control problems in this chapter are mathematically equivalent to Bayesian inference problems, thus they can be represented with belief networks. The forward kinematics play the role of a generative model and indicate whether the goal is achieved by the current state of the plant. The actual generative model specifies a probability distribution proportional to $\exp(-q(x))$ where $q(x)$ is the state cost.

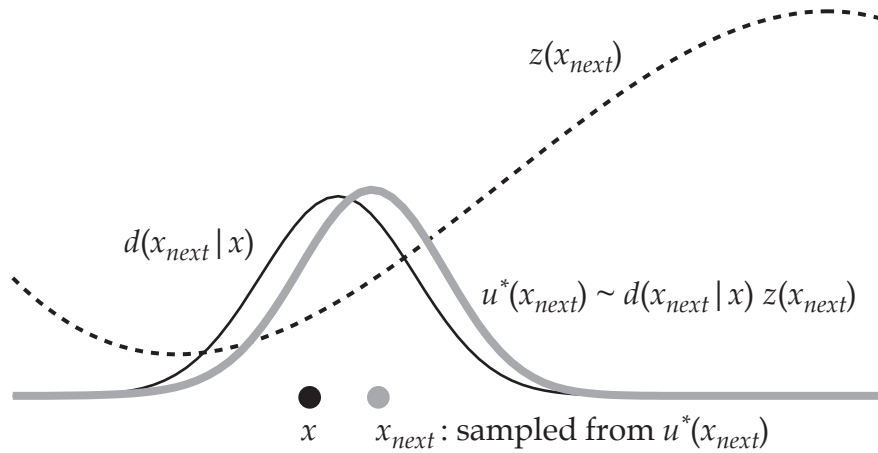


Figure 2 - optimal control with probability distributions.

The passive dynamics $d(x_{next} | x)$ is the probability distribution of the next state when the system is not controlled. The control $u(x_{next})$ is the probability distribution of the next state when the system is controlled. The optimal control $u^*(x_{next})$ is proportional to the product of the passive dynamics $d(x_{next} | x)$ and the desirability function $z(x_{next})$. Multiplying a narrow probability distribution by a smooth function has the effect of shifting the distribution along the gradient of that function. Thus the optimal control is similar in shape to the passive dynamics but is shifted towards more desirable states.

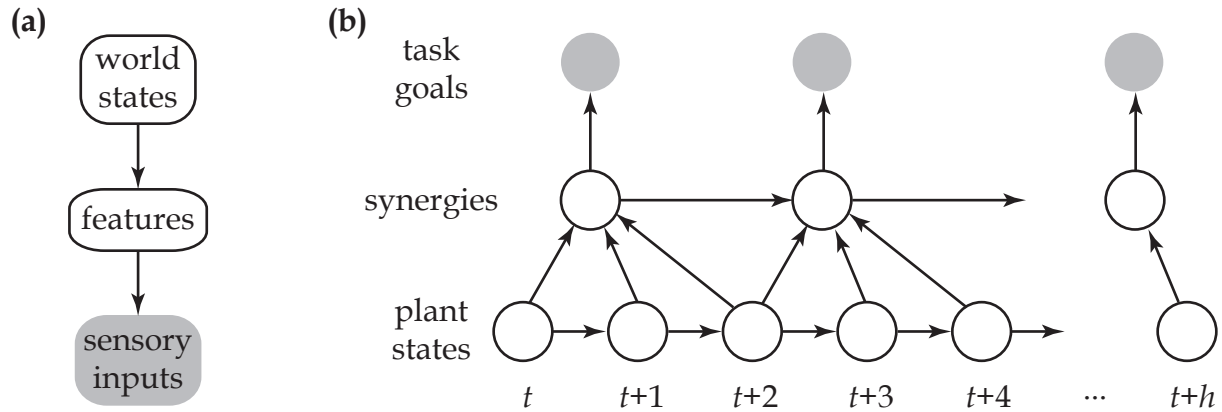


Figure 3 - belief networks for hierarchical Bayesian inference and optimal control.

(a) - By defining intermediate sensory representations (features) we can construct hierarchical generative models. The features become part of the model of how sensory inputs depend on states of the world. They are not needed to build generative models, but presumably facilitate the inversion of such models using Bayesian inference.

(b) - By defining intermediate motor representations (synergies) that depend on only some aspects of the state but over extended periods of time, we can achieve both spatial and temporal abstraction. This is done using cost functions of the form $q(h(x_t \dots x_{t+d}))$ where h are the synergy states and d is the temporal abstraction horizon.

The synergies become part of the model of how goal achievement depends on the plant state. Control is about achieving goals that are removed in time, which requires unfolding the time axis and representing multiple time steps. Limiting this unfolding to a fixed number of steps into the future is called receding horizon control. At the horizon $t+h$ we need some approximation of the desirability function $z(x_{t+h})$.