Curiosity-Driven Optimization

Tom Schaul
IDSIA
University of Lugano
Galleria 2, Manno, Switzerland
tom@idsia.ch

Yi Sun
IDSIA
University of Lugano
Galleria 2, Manno, Switzerland
yi@idsia.ch

Daan Wierstra
Brain Mind Institute
Ecole Polytechnique Fédérale de Lausanne
Lausanne, Switzerland
daan.wierstra@epfl.ch

Fausino Gomez
IDSIA
University of Lugano
Galleria 2, Manno, Switzerland
tino@idsia.ch

Jürgen Schmidhuber
IDSIA
University of Lugano
Galleria 2, Manno, Switzerland
juergen@idsia.ch

Abstract—**The principle of artificial curiosity directs active exploration towards the most informative or most *interesting* data. We show its usefulness for global black box optimization when data point evaluations are expensive. Gaussian process regression is used to model the fitness function based on all available observations so far. For each candidate point this model estimates expected fitness reduction, and yields a novel closed-form expression of expected information gain. A new type of Pareto-front algorithm continually pushes the boundary of candidates not dominated by any other known data according to both criteria, using multi-objective evolutionary search. This makes the exploration-exploitation trade-off explicit, and permits maximally informed data selection. We illustrate the robustness of our approach in a number of experimental scenarios.**

I. Introduction

Deciding where to explore next is a ubiquitous challenge in reinforcement learning and optimization. Inspired by the human drive to discover "interesting" parts of the world, one formal interpretation of *artificial curiosity* [1], [2], [3] defines momentary *interestingness* as the first derivative of the quality of an adaptive world model, where quality is measured in terms of how much the current model is able to compress the data observed so far. Something is interesting if it contains new, yet learnable regularities that can be used to *better* predict or compress the data, taking into account the limitations of the model's learning algorithm.

Here we introduce a novel variant of this notion of artificial curiosity designed for optimization problems, such as assembly-line optimization or helicopter design [4], where function evaluations are very expensive. For these problems, even a small reduction in the required number of evaluations justifies a significant investment of computational resources. Expensive global optimization is closely related to active learning [5], in that candidate points to be evaluated must be chosen with care, but the goal is different: active learning is concerned with obtaining an accurate model of the data, while in optimization modeling is secondary, and only useful inasmuch as it facilitates locating optima more efficiently. Therefore, active learning cannot be used naively for optimization. Instead, the related response surface methods [6], [7] are the standard tool for global optimization. They store all available evaluations (some possibly given in advance) and use them to model the cost function, which useful for dimensionality reduction, visualization, assessing uncertainty, and ultimately determining good points to explore [4], [8]. In addition, a statistical model of the cost function allows expert knowledge to be incorporated in the form of a Bayesian prior. Our variant of curiosity-driven exploration uses such a memory-based model to estimate the interestingness of each candidate point.

Unfortunately, the most informative points (in terms of model improvement) do not necessarily promise maximal cost reduction. We propose a novel way of handling this fundamental exploration-exploitation trade-off: to make informed data selection decisions, they are postponed until the Pareto-optimal front of candidate points with respect to both objectives is known.

Section II-A provides a brief overview of artificial curiosity and definitions of interestingness for various models. Section II-B discusses the fundamental trade-off between exploration and exploitation in optimization, and shows how data selection *after* computing the Pareto-front of a set of candidates allows for making more informed choices. Section III presents a instantiation of curiosity-driven optimization framework based on Gaussian processes, for which we derive a simple analytical expression of information gain (interestingness). Finally, section IV demonstrates the approach in a number of scenarios, and section V discusses some shortcomings and possible extensions.

II. Curiosity-driven optimization

Curiosity is the drive to actively explore the interesting regions in search space that most improve the model's predictions or explanations of what is going on in the world. Originally introduced for reinforcement learning [1], [9], the curiosity framework has been used for active learning [10], [11], to explain certain patterns of human visual attention

978-1-4244-7835-4/11/$26.00 ©2011 IEEE

better than previous approaches [12], and to explain concepts such as beauty, attention and creativity [3], [13].

A. Formalizing Interestingness

The interestingness of a new observation is the difference between the performance of an adaptive model on the observation history before and after including the new point. The goal of the active data point selection mechanism is to maximize expected cumulative future interestingness. Various proposed distance measures include: the difference in data compressibility before and after letting the learning algorithm take the new data into account [13], [3], the difference in mean squared prediction error on the observation history before and after re-training with the new point [1], [2], and the Kullback-Leibler (KL) divergence between belief distributions before and after the new observation [9]. Note that interestingness is observer-dependent and dynamic: a point that was interesting early on can become boring over time.

To permit the incorporation of a Bayesian prior, we will focus on probabilistic models and use a particular variant of the KL-based approach [9] to maximize information gain [14], [15], [16], [17], [18], [19]: The KL-divergence or relative entropy between prior and posterior (before and after seeing the new point) is invariant under any transformation of the parameter space.

Formally, let Y_{env} be the environment of interest, and y_{pre} be our current knowledge[1]. The information gain (interestingness) $\psi\left(y|y_{pre}\right)$ brought about by the observation y is defined as

$$\begin{aligned}\psi\left(y|y_{pre}\right) &= \mathbb{D}\left[p\left(Y_{env}|y_{pre};y\right)||p\left(Y_{env}|y_{pre}\right)\right]\\ &= \int p\left(y_{env}|y_{pre};y\right)\\ &\qquad \log\frac{p\left(y_{env}|y_{pre};y\right)}{p\left(y_{env}|y_{pre}\right)}dy_{env},\end{aligned}$$

where $\mathbb{D}\left[\cdot||\cdot\right]$ denotes the KL-divergence. For a set of observations y_{pre}, it is also useful to define the leave-one-out (LOO) information gain for each observation y_s w.r.t. the remaining $y_{pre\backslash s}$ as

$$\psi_{LOO}\left(y_s\right) = \psi\left(y_s|y_{pre\backslash s}\right).$$

The information gain $\psi\left(y|y_{pre}\right)$ is defined *a posteriori*, meaning that it is only defined after we see the value y. However, in most cases, we want to assess the information gain of an observation *a priori*, i.e., before seeing the value. This leads to the *expected information gain* of random variable Y, defined by

$$\begin{aligned}\Psi\left(Y|y_{pre}\right) &= \mathbb{E}\left[\psi\left(Y|y_{pre}\right)\right]\\ &= \int p\left(y|y_{pre}\right)\int p\left(y_{env}|y_{pre};y\right)\\ &\qquad\qquad \log\frac{p\left(y_{env}|y_{pre};y\right)}{p\left(y_{env}|y_{pre}\right)}dy_{env}dy\\ &= I\left(Y;Y_{env}|y_{pre}\right),\end{aligned}$$

which turns out to be the conditional mutual information between the observation and the environment.

B. Exploration-Exploitation trade-off

The previous section introduced maximal expected information gain as a possible objective for exploration. The straightforward choice of objective for exploitation would be to minimize the expected cost. However, in optimization, there is an asymmetry in utility: solutions that are better than the best currently found f_{min} largely outweigh those that are almost as good. Thus exploitation really aims at maximizing the *expected improvement* in cost with respect to f_{min}. It can be shown [6] that the expected improvement takes the following form:

$$\Delta\left(x\right) = \sigma\left(Y|y_o\right)\left(s\Phi\left(s\right)+\phi\left(s\right)\right),$$

where

$$s = \frac{f_{min} - \mathbb{E}\left[Y|y_o\right]}{\sigma\left(Y|y_o\right)},$$

while $\Phi\left(\cdot\right)$ and $\phi\left(\cdot\right)$ are the cumulative distribution and density functions of Gaussian distributions, respectively.

Optimizing conflicting objectives necessarily involves some form of trade-off, which is typically handled by using a weighted sum of both objectives, where the weights are set manually, or tuned to the problem. Combining two objectives of different scale into a single utility measure is common practice [19], but problematic [20]. In fact, if the cost landscape is ill-shaped each objective can completely dominate in some regions while being dominated in others.

Therefore we propose turning the problem around and only deciding on the trade-off after Œst evaluating both objectives for a large set of candidate points. This means finding the Pareto-front of candidates that are non-dominated w.r.t. expected improvement and expected information gain, which can be performed by any multi-objective optimization method, for example the Non-dominated Sorting Genetic Algorithm version II (NSGA-II; [21]) which is used in the experiments in section IV.

All non-dominated candidates are considered "good" solutions, and therefore each should be assigned a probability of being chosen that favors those that stand out on the Pareto-front in terms of combining both objectives. Ideally, this probability should be insensitive to quirks of the algorithm that builds the front (i.e. varying candidate densities), and to any smooth order-preserving transformation of the cost function. In addition, we can shift the focus from one objective to the other, e.g. exploitation becoming more important over time.

In the absence of an optimal way of handling this decision, we opt for the simplest solution, which consists of choosing the next point at random from the Pareto-front.

C. Algorithm

Algorithm 1 combines the components described in the previous section into a general framework for curiosity-driven optimization. In each iteration it first fits a probabilistic model M_{cost} to all known points X, and then uses M_{cost} to determine

[1] We use upper case letters for random variables, and the corresponding lower case letters for specific configurations.

Algorithm 1 Curiosity-driven Optimization

```
Given cost function $f$, models $M_{cost}$ and $M_{int}$, initial points $X$.
repeat
    Fit $M_{cost}$ to $X$.
    for all $s$ in $X$ do
        $\psi_{LOO}(s) = \mathbb{D}[M_{cost}(X) || M_{cost}(X_{-s})]$
    end for
    Fit $M_{int}$ to $\psi_{LOO}$.
    Find a set $C$ of non-dominated candidate points,
        maximizing information gain (estimated by $M_{int}$) and
        minimizing cost (estimated by $M_{cost}$)
    Choose $x^*$ from $C$.
    $X \leftarrow X \cup \{(x^*, f(x^*))\}$
until stopping criterion is met.
```

the LOO-information gain at each point. This interestingness function is then approximated using a second model, M_{int}. The Pareto-front of the candidate points is then computed using a multi-objective optimization algorithm, each model providing an estimate for one of the two objectives. Finally, a new point x^* is chosen, as described in section II-B.

The class of probabilistic models used for M_{cost} and M_{int} should be general and flexible enough to fit multi-modal and highly non-linear cost functions. Ideally, for every unknown point, such a model should be able to (efficiently) predict the expected value, the expected uncertainty associated with that prediction, and provide an analytical expression for computing information gain.

One option would be to use a mixture of Gaussians on the joint parameter-cost function space (as in [22]). However, this approach has the drawback of being sensitive to the number of Gaussians used, as well as giving poor interpolation in regions with few sampled points.

III. Curiosity-driven optimization with Gaussian Processes

In this section we present an implementation of curiosity-driven optimization which satisfies all our criteria from the previous section by using Gaussian processes to model the cost function.

A. Gaussian Processes

Gaussian processes (GP, [23]) can be seen as a probability distribution over functions, as evaluated on an arbitrary but finite number of points. Given a number of observations, a Gaussian process associates a Gaussian probability distribution to the function value for each point in the input space. Gaussian processes are capable of modeling highly complex cost landscapes through the use of appropriate covariance (kernel) functions, and are commonly used for regression and function modeling [23], [24].

Formally, we consider the Gaussian process with zero mean and the kernel function

$$k(x, x') + \sigma_n^2 \delta(x, x'),$$

where $\delta(\cdot, \cdot)$ is the Kronecker delta function. Thus, for any values y, y' at x, x', $\mathbb{E}[yy'] = k(x, x') + \sigma_n^2$. We make the assumption that the function k is smooth and local in the sense that $k(x, x') \to 0$ when $|x - x'|$ goes to infinity.

B. Gaussian Process information gain

The concept of information gain can easily be mapped onto Gaussian processes, but previous work has failed to provide a closed form expression for efficiently computing it for each candidate point [5]. Let us consider a collection of fixed reference points x_r, and view their value Y_r as the environment of interest. Our prior knowledge y_{pre} consists of all previously evaluated points x_o with value y_o. The expected information gain of the value Y at point x is thus defined by

$$\begin{aligned} \Psi_r(x|y_o) = \quad & I(Y_r; Y|y_o) \\ & = H(Y|y_o) - H(Y|Y_r, y_o), \end{aligned}$$

where $H(\cdot|\cdot)$ is the conditional entropy. A major disadvantage of this definition is that the expected information gain depends on the reference points x_r. However, we may consider the situation where the number of reference points goes to infinity. By definition, $p(Y|Y_r, y_o)$ is a Gaussian distribution, and

$$H(Y|Y_r, y_o) = \frac{1}{2} \log 2\pi e \sigma^2 (Y|Y_r, y_o).$$

Here $\sigma^2(Y|Y_r, y_o)$ is the predictive variance at x given by

$$\begin{aligned} \sigma^2(Y|Y_r, y_o) &= \sigma_n^2 + k(x, x) - k(x, x_{ro}) \\ & \qquad \left(k(x_{ro}, x_{ro}) + \sigma_n^2 I_d\right)^{-1} k(x_{ro}, x) \\ &= \sigma_n^2 + \sigma_e^2. \end{aligned}$$

with $x_{ro} = [x_r, x_o]$ and I_d being the identity matrix. We take advantage of the fact that in GP, the predictive variance depends only on the location of observations. In particular, σ_e^2 is the variance of the predicted mean $\bar{y} = \mathbb{E}[Y]$, and

$$0 \le \sigma_e^2 = \sigma^2(\bar{y}|Y_r, y_o) \le \sigma^2(\bar{y}|Y_r) = \sigma_s^2,$$

because conditioning always reduces variance for Gaussian distributions. According to [23], σ_s^2 converges to 0 when the number of reference points x_r around x goes to infinity. This indicates that σ_e^2 converges to 0 uniformly w.r.t. y_o, thus $\sigma^2(Y|Y_r, y_o)$ converges to σ_n^2 uniformly w.r.t. y_o.

When the number of observation points is sufficiently large around any given point x, we have

$$\begin{aligned} \Psi_r(x|y_o) &= H(Y|y_o) - H(Y|Y_r, y_o) \\ &\to \frac{1}{2} \log 2\pi e \sigma^2(Y|y_o) - \frac{1}{2} \log 2\pi e \sigma_n^2 \\ &= \frac{1}{2} \log \sigma^2(Y|y_o) - \frac{1}{2} \log \sigma_n^2. \end{aligned}$$

Algorithm 2 Curiosity-driven optimization with Gaussian Processes (CO-GP)

```
Given cost function $f$, kernel function, initial points $X$.
repeat
    Fit a Gaussian process $\mathcal{G}$ to $X$.
    Find a set $C$ of non-dominated candidate points $x$ maximizing $\Psi_{\mathcal{G}}(x)$ and $\Delta_{\mathcal{G}}(x)$.
    Choose $x^*$ from $C$.
    $X \leftarrow X \cup \{(x^*, f(x^*)\}$
    Optionally optimize the kernel hyperparameters w.r.t. the marginal likelihood.
until stopping criterion is met.
```

The limit no longer depends on the reference points, thus it can be used as an 'objective' measure for the expected information gain at point x:

$$\Psi\left(x|y_o\right) = \frac{1}{2}\log\sigma^2\left(Y|y_o\right) - \frac{1}{2}\log\sigma_n^2.$$

The second term is constant, therefore there is a direct connection between the expected information gain and the predictive variance given the observation, which can be computed efficiently. Note that Seo et al. [25] found the predictive variance to be a useful criterion for exploration, without realizing that it is equivalent to information gain.

C. Algorithm

Choosing a Gaussian process to model the cost function significantly simplifies the general algorithm introduced in Section II-C. First, it allows us to compute the expected information gain Ψ instead of the less robust LOO-information gain. Second, the model M_{int} is no longer necessary, as Ψ can be computed for unknown points as well. The resulting algorithm (CO-GP) is shown in Algorithm 2. The remainder of this section discusses some practical considerations.

The computational complexity of each iteration of CO-GP is dominated by one matrix inversion of $O(n^3)$, where n is the total number of evaluations. Building the Pareto-front consumes most of the computation time early on, but scales with $O(n^2)$. The computational complexity of Gaussian processes can be reduced e.g. by implementing them online and using a reduced base vector set, containing only the most informative points [26]. We have not implemented these yet, as computation time was not a major concern in our experiments.

Gaussian process regression only gives reasonable results when the kernel hyperparameters are set properly. With enough computation time available, we can periodically optimize the hyperparameters with respect to the marginal likelihood. For this we use the natural evolution strategies algorithm (NES, [27], [28]). Potentially, we could also employ diagnostic methods [8] to determine whether the model is appropriate.

At each iteration, an inner multi-objective optimization algorithm is used, in our case, NSGA-II [21]. The implementations of this and NES (above) are taken from the PyBrain machine learning library [29]. We can make use of our available information to make this step more efficient. For example, we initialize the search with the Pareto-front found in the previous iteration. Furthermore, as we want the search to roughly cover the range of known points, we adjust the scale (for step-sizes) accordingly.

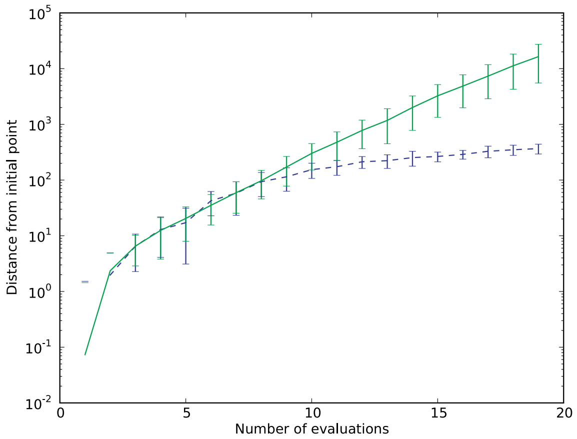

Fig. 1. Performance in linear regions. The plot shows the performance of CO-GP on a linear surface (averaged over 20 independent runs), in terms of the distance from the best point found so far to the initial point. The green (solid) and blue (dashed) curves correspond to CO-GP with hyperparameter adaptation enabled and disabled, respectively. We observe that the distance from the initial point (and thus the decrease in cost) grows exponentially if the hyperparameters are adapted, but only linearly otherwise.

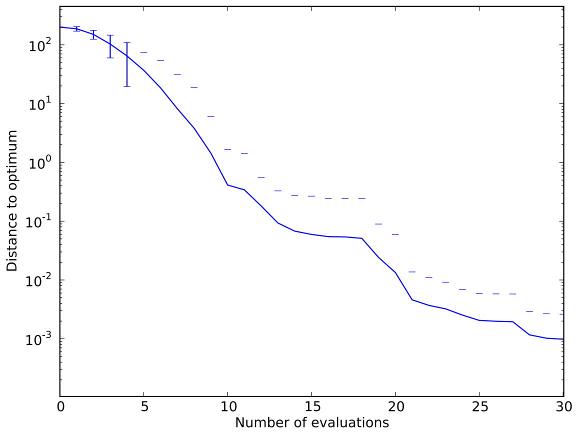

Fig. 2. Precisely locating optima. The plot shows the performance of CO-GP on a unimodal, quadratic surface (averaged over 20 independent runs), in terms of the distance from the optimum to the best point found so far. This distance decreases exponentially with the number of points.

IV. Experiments

To demonstrate the practical viability of CO-GP, we first investigate how it handles a number of common but problematic scenarios and then test it on a couple of standard benchmarks functions. Following [23], all experiments use the common

Gaussian kernel (also known as radial basis function) with noise, which is a very robust choice in practice.

A. Long slopes

Many cost function landscapes contain large linear regions. Specifically, if the scale of the region covered by the initial points is too small, almost any landscape will appear linear. An ideal algorithm should be able to exploit the linearity of such regions. In particular, it is highly desirable to have the searched region grow exponentially with the number of points. Note that many well-known algorithms, such as Estimation of Distribution Algorithms, do not have this property, and instead rely either on correct initialization or heuristics [30]. In contrast, CO-GP does have this property, as our results on the linear function show (see Figure 1).

B. Local optimization

While designed primarily for multi-modal cost landscapes, we investigated how our approach handles simple cost landscapes with a single optimum. The success criterion for this case is to have the distance to the optimum decrease exponentially with the number of points. While we cannot prove that this is the case in general, Figure 2 shows that it holds for the multi-dimensional sphere function. This indicates that CO-GP can locate optima up to a high precision, at least whenever, locally, the cost function is approximately quadratic.

C. Global optimization

Every global optimization algorithm should provide a guarantee that in the limit its chosen points will cover the search space densely, which is the only way to ensure that it will eventually find the global optimum. Optimization based on expected improvement has been shown to have this property [31]. It turns out that if we remove the information gain objective from CO-GP, the algorithms are equivalent. Therefore, as one extreme of the Pareto-front will always correspond to the point maximizing expected improvement exclusively, and that point has a non-zero probability of being chosen, CO-GP inherits the property that it always finds the global optimum in the limit.

D. Branin benchmark

The Branin function [6], [8] is a commonly used benchmark for global optimization of the form:

$$\begin{aligned} f(x_1, x_2) = \quad & a(x_2 - bx_1^2 + cx_1 - d)^2 \\ & + e(1-f)\cos(x_1) + e, \end{aligned}$$

where the standard parameter settings are $a = 1$, $b = \frac{5}{4\pi^2}$, $c = \frac{5}{\pi}$, $d = 6$, $e = 10$, $f = \frac{1}{8\pi}$. The function has three global minima (at $(-\pi, 12.275)$, $(\pi, 2.275)$ and $(9.42478, 2.475)$ with value $f(x^*) = 0.397887$), a bounded domain and a non-trivial structure. Figure 3 illustrates the behavior of CO-GP on the Branin function over the course of a single run, starting with four points on the boundary corners. The Gaussian process model produces a good fit of the true function after about 30 iterations. Locating one optimum (up to a precision of 0.1) requires only 28 ± 8 evaluations, locating all three requires 119 ± 31. The qualitative behavior of the algorithm is very intuitive, placing part of its search points spaced broadly within the domain, while the other part forms clusters of points ever closer around the optima. Although this experiment is intended as a proof of concept, not an empirical comparison to other global optimization algorithms, the quantitative results indicate that CO-GP is on par with the best results reported in the literature [6].

V. Discussion

The results in section IV demonstrate that curiosity-driven optimization properly handles a number of typical optimization challenges. Although based on the general and theoretically powerful principle of artificial curiosity, our current implementation exhibits certain weaknesses.

In particular, despite the derived closed-form expressions for the objectives, the method's computational cost is still high, limiting the application domain to (relatively) costly optimization problems. On the other hand, many real-world problems (e.g., helicopter design [4]) are precisely of this type. Nevertheless, in future research we would like to extend the method such that one can specify the amount of acceptable computational effort for the selection of each data point, and have the algorithm make the best choice under this constraint.

Another drawback of our approach is that it is greedier than the original curiosity framework: it does not necessarily maximize cumulative future expected information gain, but greedily selects the next data point that is expected to be the immediately most interesting. More sophisticated reinforcement learning algorithms will be necessary to maximize the expected *sum* of future intrinsic rewards (each reward being proportional to the information gain of the corresponding observed data).

Unless the issue of having to choose an appropriate kernel is ignored, the method is not completely parameter-free (a problem shared by all kernel-based methods). However, this may also be seen as an asset, allowing for the elegant integration of expert knowledge.

The results bode well for applying the general template (Algorithm 1) to related domains such as constrained or discrete optimization, or even mixed-integer programming. One application domain where we expect curiosity-based methods to be counter-productive, however, are problems that require minimizing cumulative evaluated regret, because those tend to require risk-averse exploration (which in practice generally translates to careful local search): here curiosity may kill the cat.

VI. Conclusion

This paper introduced a novel exploration principle for costly optimization problems derived from the principle of artificial curiosity. Using Gaussian processes as a model, we derived a closed-form expression for expected information gain. Pursuing expected cost reduction as a second objective, our algorithm builds a Pareto-front of non-dominated

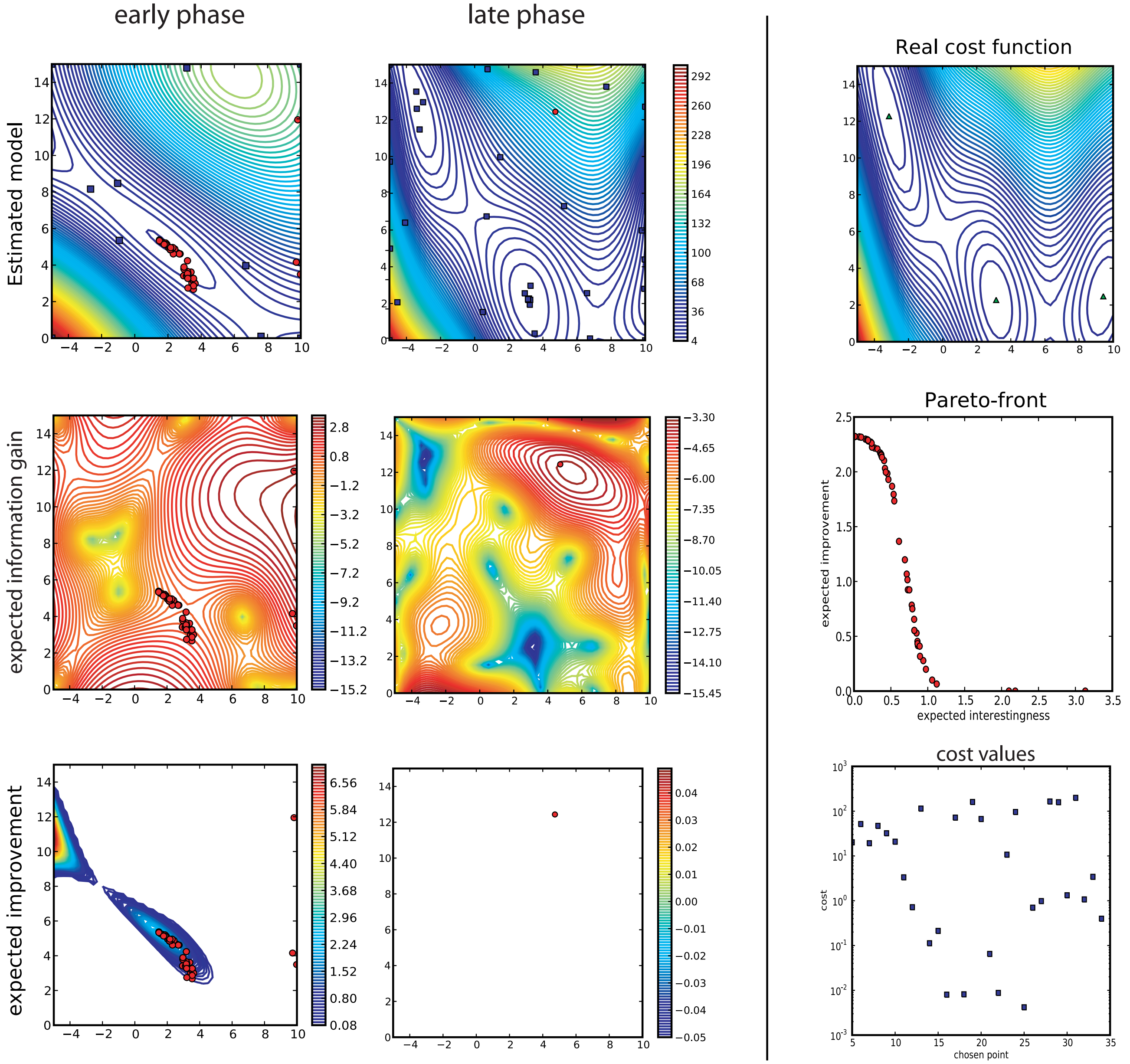

Fig. 3. Optimization on the Branin function. The plot in the top right corner shows a contour plot of the Branin function with the three global optima marked with triangles. The left column shows the estimated cost function model (top), and the two competing objectives, expected information gain (middle) and expected improvement (bottom), in an early phase after 10 points (blue squares) have been observed. The red circles are the points on the Pareto-front being considered for the next choice. The middle column shows the same information after 30 iterations. Note that in this later stage the model is very close to the true function. The plot in the middle of the right column shows the shape of the Pareto-front corresponding to the situation in the left column, and the plot on the bottom right shows the values of the cost function at the chosen points (the initial 4 corner points are not shown). In the early phase, the Pareto-front contains a continuum of points in the center that trade off improvement and information gain, plus a few isolated points with high information gain, but very low expected improvement. After 30 iterations, two of the global optima have been located precisely. The expected improvement is zero everywhere, so the Pareto-front is collapsed to the single point with highest information gain. CO-GP now performs a purely exploratory step, and will continue to do so until it leads to non-zero expected improvement (e.g. around the third optimum). On average, CO-GP requires about 120 points to locate all three optima with high accuracy.

candidate data points, and thus can handle the exploration-exploitation trade-off in hindsight. Experiments demonstrate that curiosity-driven optimization with Gaussian processes is robust and promising, opening up an exciting new direction of optimization research.

Acknowledgments

This research was funded by the SNSF grant 200021-113364/1. We thank Tobias Glasmachers and our anonymous reviewers for their helpful input.

References

[1] J. Schmidhuber, "Curious model-building control systems," *IEEE International Joint Conference on Neural Networks*, pp. 1458 – 1463, 1991.

[2] ——, "Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts," *Connection Science*, vol. 18, pp. 173–187, 2006.

[3] ——, "Driven by compression progress: A simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes," *Anticipatory Behavior in Adaptive Learning Systems, from Sensorimotor to Higher-level Cognitive Capabilities*, 2009.

[4] A. Booker, J. J. Dennis, P. Frank, and D. Serafini, "Optimization using surrogate objectives on a helicopter test example," *Computational Methods in Optimal Design and Control*, 1998.

[5] A. Krause and C. Guestrin, "Nonmyopic active learning of gaussian processes: An exploration-exploitation approach," in *Proceedings of the International Conference on Machine Learning*, 2007.

[6] D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *Journal of Global Optimization*, vol. 21, pp. 345–383, 2001.

[7] A. W. Moore and J. Schneider, "Memory-based stochastic optimization," in *Advances in Neural Information Processing Systems*, 1996.

[8] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, pp. 455–492, 1998.

[9] J. Storck, J. Hochreiter, and J. Schmidhuber, "Reinforcement-driven information acquisition in non-deterministic environments," in *International Conference on ArtiŒial Neural Networks (ICANN)*, Paris, 1995, pp. 159–164.

[10] M. P. Deisenroth, C. E. Rasmussen, and J. Peters, "Gaussian process dynamic programming," *Neurocomputing*, pp. 1508–1524, 2009.

[11] T. Pfingsten, "Bayesian active learning for sensitivity analysis," in *Machine Learning: ECML 2006*, 2006, pp. 353 – 364.

[12] L. Itti and P. Baldi, "Bayesian surprise attracts human attention," in *Advances in Neural Information Processing Systems*, Y. W. Platt, B. Schölkopf, and J., Eds. MIT Press, 2006, pp. 547—-554.

[13] J. Schmidhuber, "Simple algorithmic principles of discovery, subjective beauty, selective attention, curiosity and creativity," *Lecture Notes In ArtiŒial Intelligence*, vol. 4754, 2007.

[14] V. V. Fedorov, *Theory of optimal experiments*. New York: Academic press, 1972.

[15] J.-N. Hwang, J. J. Choi, S. Oh, and R. J. I. Marks, "Query-based learning applied to partially trained multilayer perceptrons," *IEEE Transactions on Neural Networks*, vol. 2, pp. 131–136, 1991.

[16] D. J. C. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, pp. 550–604, 1992.

[17] M. Plutowski, G. Cottrell, and H. White, "Learning mackey-glass from 25 examples, plus or minus 2," in *Advances in Neural Information Processing Systems*, 1994, pp. 1135–1142.

[18] D. A. Cohn, "Neural network exploration using optimal experiment design," in *Advances in Neural Information Processing Systems*, 1994, pp. 679–686.

[19] K. Chaloner and I. Verdinelli, "Bayesian experimental design: A review," *Statistical Science*, vol. 10, pp. 273–304, 1995.

[20] Y. Zhang, W. Xu, and J. Callan, "Exploration and exploitation in adaptive filtering based on bayesian active learning," in *International Conference on Machine Learning (ICML)*, 2003, pp. 896–903.

[21] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182 – 197, 2002.

[22] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *Journal of ArtiŒial Intelligence Research*, vol. 4, pp. 129–145, 1995.

[23] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. MIT Press, 2006.

[24] D. Büche, N. N. Schraudolph, and P. Koumoutsakos, "Accelerating Evolutionary Algorithms with Gaussian Process Fitness Function Models," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. C 35, no. 2, pp. 183–194, 2005.

[25] S. Seo, M. Wallat, T. Graepel, and K. Obermayer, "Gaussian process regression: Active data selection and test point rejection," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2000, pp. 241–246.

[26] L. Csató and M. Opper, "Sparse on-line gaussian processes," *Neural Computation*, vol. 14, 2002.

[27] D. Wierstra, T. Schaul, J. Peters, and J. Schmidhuber, "Natural Evolution Strategies," in *Proceedings of the Congress on Evolutionary Computation (CEC08), Hongkong*. IEEE Press, 2008.

[28] T. Glasmachers, T. Schaul, Y. Sun, D. Wierstra, and J. Schmidhuber, "Exponential Natural Evolution Strategies," in *Genetic and Evolutionary Computation Conference (GECCO)*, Portland, OR, 2010.

[29] T. Schaul, J. Bayer, D. Wierstra, Y. Sun, M. Felder, F. Sehnke, T. Rückstieß, and J. Schmidhuber, "PyBrain," *Journal of Machine Learning Research*, vol. 11, pp. 743–746, 2010.

[30] P. Bosman, J. Grahl, and D. Thierens, "Enhancing the performance of maximumlikelihood gaussian edas using anticipated mean shift," in *Parallel Problem Solving from Nature PPSN X*, 2008, pp. 133 – 143.

[31] M. Locatelli, "Bayesian algorithms for one-dimensional global optimization," *Journal of Global Optimization*, pp. 57–76, 1997.