Formal Theory of Creativity, Fun, and Intrinsic Motivation (1990–2010)

Jürgen Schmidhuber

***Abstract*—The simple, but general formal theory of fun and intrinsic motivation and creativity (1990–2010) is based on the concept of maximizing intrinsic reward for the active creation or discovery of *novel, surprising patterns* allowing for *improved* prediction or data compression. It generalizes the traditional field of *active learning*, and is related to old, but less formal ideas in aesthetics theory and developmental psychology. It has been argued that the theory explains many essential aspects of intelligence including autonomous development, science, art, music, and humor. This overview first describes theoretically optimal (but not necessarily practical) ways of implementing the basic computational principles on exploratory, intrinsically motivated agents or robots, encouraging them to provoke event sequences exhibiting previously unknown, but learnable algorithmic regularities. Emphasis is put on the importance of limited computational resources for online prediction and compression. Discrete and continuous time formulations are given. Previous *practical, but nonoptimal* implementations (1991, 1995, and 1997–2002) are reviewed, as well as several recent variants by others (2005–2010). A simplified typology addresses current confusion concerning the precise nature of intrinsic motivation.**

***Index Terms*—Active learning, aesthetics theory, art, attention, developmental psychology, formal theory of creativity, fun, humor, limited computational resources, music, novel patterns, novelty, science, surprise, typology of intrinsic motivation.**

I. Introduction

TO SOLVE existential problems such as avoiding hunger or heat, a baby has to learn how the initially unknown environment responds to its actions. Therefore, even when there is no immediate need to satisfy thirst or other built-in primitive drives, the baby does not run idle. Instead, it actively conducts experiments: what sensory feedback do I get if I move my eyes or my fingers or my tongue just like that? Being able to predict effects of actions will later make it easier to plan control sequences leading to desirable states, such as those where heat and hunger sensors are switched off.

The growing infant quickly gets bored by things it already understands well, but also by those it does not understand at all, always searching for new effects exhibiting some yet unexplained but *easily learnable* regularity. It acquires more and more complex behaviors building on previously acquired, simpler behaviors. Eventually, it might become a physicist discovering previously unknown physical laws, or an artist creating new eye-opening artworks, or a comedian coming up with novel jokes.

For a long time I have been arguing, using various wordings, that all this behavior is driven by a very simple algorithmic mechanism that uses reinforcement learning (RL) to maximize the *fun* or *internal joy* for the discovery or creation of *novel patterns.* Both concepts are essential: *pattern* and *novelty*. A data sequence exhibits a *pattern* or regularity if it is compressible [45], that is, if there is a relatively short program that encodes it, for example, by predicting some of its components from others (irregular noise is unpredictable and *boring*). Relative to some subjective observer, a pattern is temporarily *novel* or *interesting* or *surprising* if the observer initially did not know the regularity, but is able to *learn* it. The observer's learning progress can be precisely *measured* and translated into *intrinsic reward* for a separate RL controller selecting the actions causing the data. Hence, the controller is continually motivated to create more surprising data.

Since 1990, agents were built that implement this idea. They may be viewed as simple artificial scientists or artists with an intrinsic desire to build a better model of the world and of what can be done in it. To improve their models, they acquire skills to create/discover *more* data predictable or compressible in hitherto unknown ways [67], [69]–[71], [77], [79], [85], [92]–[94], [96], [97], [99], [111]. They are intrinsically motivated to invent and conduct experiments, actively exploring their environment, always trying to learn new behaviors exhibiting previously unknown algorithmic regularities. They embody approximations of a simple, but general, formal theory of creativity and curiosity and interestingness and fun, explaining essential aspects of human or nonhuman intelligence, including selective attention, science, art, music, and humor [85], [92], [94], [96], [97]. The crucial ingredients are:

1) An adaptive world model, essentially a predictor or compressor of the continually growing history of actions/events/sensory inputs, reflecting what is currently known about how the world works.
2) A learning algorithm that continually improves the model (detecting novel, initially surprising spatio–temporal patterns that subsequently become known patterns).
3) Intrinsic rewards measuring the model's improvements (first derivative of learning progress) due to the learning algorithm (thus, measuring the *degree* of subjective surprise or fun).
4) A separate reward optimizer or reinforcement learner, which translates those rewards into action sequences or behaviors expected to optimize future reward.

Manuscript received February 15, 2010; revised April 20, 2010; accepted June 27, 2010. Date of publication July 12, 2010; date of current version September 10, 2010. This work was partially supported by the EU Project IM-CLEVER.

The author is with the Swiss AI Laboratory, Dalle Molle Institute for Artificial Intelligence, University of Lugano, Manno 6928, Switzerland (e-mail: juergen@idsia.ch).

Digital Object Identifier 10.1109/TAMD.2010.2056368

A simple example may help to see that it is really possible to learn from intrinsic reward signals *à la* item 3 that one can learn even more in places never visited before. In an environment with red and blue boxes, whenever the learning agent opens a red box, it will find an easily learnable novel geometric pattern (that is, its predictor will make progress and thus, generate intrinsic reward), while all blue boxes contain a generator of unpredictable, incompressible white noise. That is, all the RL controller has to learn is a simple policy: open the next unopened red box.

Ignoring issues of computation time, it is possible to devise mathematically optimal, *universal* RL methods for such systems [85], [92], [96], [97] (2006–2009). More about this in Section II. However, the practical implementations so far [69], [70], [77], [79], [111] were nonuniversal and made approximative assumptions. Among the many ways of combining algorithms for 1)–4), the following variants were implemented.

A) 1990: Nontraditional RL (without restrictive Markovian assumptions [72]) based on an adaptive recurrent neural network as a predictive world model [68] is used to maximize the controller's intrinsic reward, which is proportional to the model's prediction errors [67], [71].
B) 1991: Traditional RL [35], [114] is used to maximize intrinsic reward created in proportion to expected *improvements* (first derivatives) of prediction error [69], [70].
C) 1995: Traditional RL maximizes intrinsic reward created in proportion to relative entropies between the learning agent's priors and posteriors [111].
D) 1997–2002: Nontraditional RL [103] (without restrictive Markovian assumptions) learns probabilistic, hierarchical programs and skills through zero-sum intrinsic reward games of two players (called the right brain and the left brain), each trying to out-predict or surprise the other, taking into account the computational costs of learning, and learning *when* to learn and *what* to learn [77], [79].

B)–D) (1991–2002) also showed experimentally how intrinsic rewards can substantially accelerate goal-directed learning and *external* reward intake.

A. Outline

Section II will summarize the formal theory of creativity in a nutshell, laying out a mathematically rigorous but not necessarily practical framework. Section III will then discuss previous concrete implementations of the nonoptimal, but currently still more practical variants A)–D) mentioned above, and their limitations. Section IV will discuss relations to work by others, explain how the theory extends the traditional field of active learning, and how it formalizes and extends previous informal ideas of developmental psychology and aesthetics theory. Section V will offer a natural typology of computational intrinsic motivation, and Section VI will briefly explain how the theory is indeed sufficiently general to explain all kinds of creative behavior, from the discovery of new physical laws through active design of experiments, to the invention of jokes and music and works of art.

II. Formal Details of the Theory of Creativity

The theory formulates essential principles behind numerous intrinsically motivated *creative* behaviors of biological or artificial agents embedded in a possibly unknown environment. The corresponding algorithmic framework uses general RL (Section II-G, [32], and [98]) to maximize not only external reward for achieving goals such as the satisfaction of hunger and thirst, but also *intrinsic* reward for learning a better world model, by creating/discovering/learning novel patterns in the growing history of actions and sensory inputs, where the theory formally specifies what exactly is a *pattern*, what exactly is *novel* or surprising, and what exactly it means to incrementally *learn* novel skills leading to more novel patterns.

A. The Agent and its Improving Model

Let us consider a learning agent whose single life consists of discrete cycles or time steps $t = 1, 2 \ldots T$. Its complete lifetime T may or may not be known in advance. In what follows, the value of any time-varying variable Q at time t ($1 \leq t \leq T$) will be denoted by $Q(t)$, the ordered sequence of values $Q(1) \ldots Q(t)$ by $Q(\leq t)$, and the (possibly empty) sequence $Q(1) \ldots Q(t-1)$ by $Q(< t)$. At any given t, the agent receives a real-valued input $x(t)$ from the environment and executes a real-valued action $y(t)$ which may affect future inputs. At times $t < T$, its goal is to maximize future success or *utility*

$$u(t) = E_\mu \left[\sum_{\tau=t+1}^{T} r(\tau) \;\middle|\; h(\leq t) \right] \tag{1}$$

where the reward $r(t)$ is a special real-valued input (vector) at time t, $h(t)$ the ordered triple $[x(t), y(t), r(t)]$ (hence $h(\leq t)$ is the known history up to t), and $E_\mu(\cdot \mid \cdot)$ denotes the conditional expectation operator with respect to some possibly unknown distribution μ from a set $\mathcal{M}$ of possible distributions. Here, $\mathcal{M}$ reflects whatever is known about the possibly probabilistic reactions of the environment. As a very general example, $\mathcal{M}$ may contain all computable distributions [32], [45], [110]. This essentially includes all environments one could write scientific papers about. There is just one life, no need for predefined repeatable trials, no restriction to Markovian interfaces between sensors and environment [72]. Note that traditional Markovian RL [114] assumes that the world can be modeled as a Markov decision process (MDP), and that the perceptual system reveals the current state. In realistic scenarios, however, robots have to learn to memorize previous relevant inputs in form of appropriate internal representations, which motivates the work on RL in partially observable MDPs or partially observable Markov decision processes (POMDPs), e.g., [35] and [72]. The utility function implicitly takes into account the expected remaining lifespan $E_\mu(T \mid h(\leq t))$ and thus, the possibility to extend the lifespan through appropriate actions [83], [98]. Note that mathematical analysis is *not* simplified by discounting future rewards like in traditional RL theory [114]—one should avoid such distortions of real rewards whenever possible.

To maximize $u(t)$, the agent may profit from an adaptive, predictive *model* p of the consequences of its possible interactions with the environment. At any time t ($1 \leq t < T$), the model $p(t)$ will depend on the observed history so far, $h(\leq t)$. It may be viewed as the current explanation or description of $h(\leq t)$, and may help to predict future events, including rewards. Let $C(p, h)$ denote some given model p's quality or performance

evaluated on a given history h. Natural quality measures will be discussed in Section II-B.

To encourage the agent to actively create data leading to easily learnable improvements of p [70], [71], [79], [85], [92], [94], [96], [97], [99], [111], the reward signal $r(t)$ is simply split into two scalar real-valued components: $r(t) = g(r_{\rm ext}(t), r_{\rm int}(t))$, where g maps pairs of real values to real values, e.g., $g(a,b) = a + b$. Here, $r_{\rm ext}(t)$ denotes traditional *external* reward provided by the environment, such as negative reward in response to bumping against a wall, or positive reward in response to reaching some teacher-given goal state. The formal theory of creativity, however, is especially interested in $r_{\rm int}(t)$, the *intrinsic* reward, which is provided whenever the model's quality improves—for *purely creative* agents $r_{\rm ext}(t) = 0$ for all valid t.

The current *intrinsic* reward, *creativity* reward, *curiosity* reward, *aesthetic reward*, or *fun* $r_{\rm int}(t)$ of the action selector is the current *surprise* or *novelty* measured by the *improvements* of the world model p at time t.

Formally, the intrinsic reward in response to the model's progress (due to some application-dependent model improvement algorithm) between times t and $t+1$ is

$$r_{\rm int}(t+1) = f[C(p(t), h(\leq t+1)), C(p(t+1), h(\leq t+1))] \quad (2)$$

where f maps pairs of real values to real values. Various alternative progress measures are possible; most obvious is $f(a,b) = a - b$. This corresponds to a discrete time version of maximizing the first derivative of the model's quality. Note that both the old and the new model have to be tested on the same data, namely, the history so far. So progress between times t and $t+1$ is defined based on two models of $h(\leq t+1)$, where the old one is trained only on $h(\leq t)$ and the new one also gets to see $h(t \leq t+1)$. This is like $p(t)$ predicting data of time $t+1$, then observing it, then learning something, then becoming a measurably better model $p(t+1)$.

The above description of the agent's motivation conceptually separates the goal (finding or creating data that can be modeled better or faster than before) from the means of achieving the goal. Let the controller's RL mechanism figure out how to translate such rewards into action sequences that allow the given world model improvement algorithm to find and exploit previously unknown types of regularities. It is the task of the RL algorithm to trade off long-term versus short-term intrinsic rewards of this kind, taking into account all costs of action sequences [70], [71], [79], [85], [92], [94], [96], [97], [99], [111]. The universal RL methods of Section II-G, as well as recurrent neural network (RNN)-based RL (Section III-A) and success-story algorithm (SSA)-based RL (Section III-D) can in principle learn useful internal states containing memories of relevant previous events; less powerful RL methods (Sections III-B, III-C) cannot.

B. How to Measure Model Quality Under Time Constraints

In theory, $C(p, h(\leq t))$ should take the entire history of actions and perceptions into account [85], like the following performance measure C_{xry}

$$C_{xry}(p, h(\leq t)) = \sum_{\tau=1}^{t} \| \operatorname{pred}(p, x(\tau)) - x(\tau) \|^2 + \| \operatorname{pred}(p, r(\tau)) - r(\tau) \|^2 + \| \operatorname{pred}(p, y(\tau)) - y(\tau) \|^2 \quad (3)$$

where $\operatorname{pred}(\mathrm{p}, \mathrm{q})$ is p's prediction of event q from earlier parts of the history [85].

C_{xry} ignores the danger of overfitting through a p that just stores the entire history without compactly representing its regularities, if any. The principle of minimum description length (MDL) [37], [45], [62], [110], [115], [116], however, also takes into account the description size of p, viewing p as a compressor program of the data $h(\leq t)$. This program p should be able to deal with any prefix of the growing history, computing an output starting with $h(\leq t)$ for any time t ($1 \leq t < T$). (A program that wants to halt after t steps can easily be fixed/augmented by the trivial method that simply stores any raw additional data coming in after the halt.)

$C_l(p, h(\leq t))$ denotes p's compression performance on $h(\leq t)$: the number of bits needed to specify both the predictor and the deviations of the sensory history from its predictions, in the sense of loss-free compression. The smaller C_l, the more regularity and lawfulness in the observations so far.

For example, suppose p uses a small predictor that correctly predicts many $x(\tau)$ for $1 \leq \tau \leq t$. This can be used to encode $x(\leq t)$ compactly. Given the predictor, only the wrongly predicted $x(\tau)$ plus information about the corresponding time steps τ are necessary to reconstruct input history $x(\leq t)$, e.g., [73]. Similarly, a predictor that learns a probability distribution on the possible next events, given previous events, can be used to efficiently encode observations with high (respectively low) predicted probability by few (respectively many) bits (Section III-C; [31], [101]), thus achieving a compressed history representation.

Alternatively, p could also make use of a 3-D world model or simulation. The corresponding MDL-based quality measure $C_{3D}(p, h(\leq t))$ is the number of bits needed to specify all polygons and surface textures in the 3-D simulation, plus the number of bits needed to encode deviations of $h(\leq t)$ from the predictions of the simulation. Improving the 3-D model by adding or removing polygons may reduce the total number of bits required.

The ultimate limit for $C_l(p, h(\leq t))$ would be $K^*(h(\leq t))$, a variant of the Kolmogorov complexity of $h(\leq t)$, namely, the length of the shortest program (for the given hardware) that computes an output starting with $h(\leq t)$ [37], [45], [80], [110]. Here there is no need not worry about the fact that $K^*(h(\leq t))$ in general cannot be computed exactly, only approximated from above (indeed, for most practical predictors the approximation will be crude). This just means that some patterns will be hard to detect by the limited predictor of choice, that is, the reward maximizer will get discouraged from spending too much effort on creating those patterns.

$C_l(p, h(\leq t))$ does not take into account the time $\tau(p, h(\leq t))$ spent by p on computing $h(\leq t)$. A runtime-dependent performance measure inspired by concepts of optimal universal search [43], [81], [82], [85], [96], [99] is

$$C_{l\tau}(p, h(\leq t)) = C_l(p, h(\leq t)) + \log\, \tau(p, h(\leq t)). \quad (4)$$

Here, compression by one bit is worth as much as runtime reduction by a factor of 1/2. From an asymptotic optimality-oriented point of view this is one of the best ways of trading off storage and computation time [43], [81], [82].

In practical applications (Section III), the predictor/compressor of the continually growing data typically will have

to calculate its output online, that is, it will be able to use only a constant number of computational instructions per second to predict/compress new data. The goal of the possibly much slower learning algorithm must then be to improve the compressor such that it keeps operating online within those time limits, while compressing/predicting better than before. The costs of computing $C_{xry}(p, h(\leq t))$ and $C_l(p, h(\leq t))$, and similar performance measures are linear in t, assuming p consumes equal amounts of computation time for each single prediction. Therefore, online evaluations of learning progress on the full history so far generally cannot take place as frequently as the continually ongoing online predictions.

At least some of the learning and its progress evaluations may take place during occasional "sleep" phases [85], [96]. But practical implementations so far have looked only at parts of the history for efficiency reasons: The systems described in Sections III-A–III-D [70], [71], [79], [111] used online settings (one prediction per time step, and constant computational effort per prediction), nonuniversal adaptive compressors or predictors, and approximative evaluations of learning progress, each consuming only constant time despite the continual growth of the history.

C. Feasibility of Loss-Free Compression With Examples

Any set of raw data, such as the history of some observer's previous actions and sensations and rewards including suspected noise, exhibits a pattern or regularity if there exists an algorithm that is significantly shorter than the raw data, but is able to encode it without loss of information [37], [45], [109], [110]. Random noise is irregular and arbitrary and incompressible, but random-dot stereograms (e.g., a single foreground square against a more distant background) are compressible since parts of the data are just copied from others. Videos are regular as most single frames are very similar to the previous one. By encoding only the deviations, movie compression algorithms can save lots of storage space. Complex-looking fractal images are regular, as they usually look a lot like their details, being computable by very short programs that reuse the same code over and over again for different image parts. The entire universe is regular and apparently rather benign [74], [78], [88], [90]: every photon behaves the same way; gravity is the same on Jupiter and Mars, mountains usually do not move overnight, but tend to remain where they are, etc.

Many data analysis techniques are natural by-products of loss-free compression. For example, data set compression is possible if the data can be separated into clusters of numerous close neighbors and few *outliers*. *Abstraction* is another typical by-product. For example, if the predictor/compressor uses a neural net, the latter will create feature hierarchies, higher layer units typically corresponding to more abstract features, fine-grained where necessary. Any good compressor will identify shared regularities among different already existing internal data structures, to shrink the storage space needed for the whole. *Consciousness* may be viewed as a by-product of this [96], [97], since there is one thing that is involved in all actions and sensory inputs of the agent, namely, the agent itself. To efficiently encode the entire data history, it will profit from creating some sort of internal *symbol* or code (e.g., a neural activity pattern) representing itself. Whenever this representation is actively used, say, by activating the corresponding neurons through new incoming sensory inputs or otherwise, the agent could be called *self-aware* or *conscious* [96], [97].

True, any loss-free compression method will require space that grows without bound over time. But this is *not* a fundamental practical obstacle. Soon storage for 100 years of high resolution video of will be cheap. If you can store the data, do not throw it away. The data are *holy* as it is the only basis of all that can be known about the world [96], [97]. Attempts at predicting/compressing the raw data (by finding regularities/abstractions) should take place in a *separate*, typically smaller part of the storage.

Even humans may store much of the incoming sensory data. A human lifetime rarely lasts much longer than 3×10^9 seconds. The human brain has roughly 10^{10} neurons, each with 10^4 synapses on average. Assuming that only half of the brain's capacity is used for storing raw data, and that each synapse can store at most 6 bits, there is still enough capacity to encode the lifelong sensory input stream with a rate of at least 10^5 bits/s, comparable to the demands of a movie with reasonable resolution, but possibly at a much higher rate, assuming that human compressors are much smarter than those of cameras.

D. Optimal Predictors Versus Optimal Compressors

For the theoretically inclined: There is a deep connection between optimal prediction and optimal compression. Consider Solomonoff's theoretically optimal, universal way of predicting the future [32], [45], [109], [110]. Given an observation sequence $q(\leq t)$, the Bayes formula is used to predict the probability of the next possible $q(t+1)$. The only assumption is that there exists a computer program that can take any $q(\leq t)$ as an input and compute its *a priori* probability according to the μ prior. (This assumption is extremely general, essentially including all environments one can write scientific papers about, as mentioned above.) In general this program is unknown, hence a mixture prior is used instead to predict

$$\xi(q(\leq t)) = \sum_i w_i \mu_i(q(\leq t)) \tag{5}$$

a weighted sum of *all* distributions $\mu_i \in \mathcal{M}, i = 1, 2 \ldots$ where the sum of the constant positive weights satisfies $\sum_i w_i \leq 1$. This is indeed the best one can possibly do, in a very general sense [32], [110]. The drawback of the scheme is its incomputability, since $\mathcal{M}$ contains infinitely many distributions. One may increase the theoretical power of the scheme by augmenting $\mathcal{M}$ by certain nonenumerable but limit-computable distributions [80], or restrict it such that it becomes computable, e.g., by assuming the world is computed by some unknown, but deterministic computer program sampled from the Speed Prior [81], which assigns low probability to environments that are hard to compute by any method.

Remarkably, under very general conditions both universal inductive inference [45], [109], [110] and the compression-oriented MDL approach [37], [45], [62], [115], [116] converge to the correct predictions in the limit [56]. It should be mentioned, however, that the former converges faster.

As far as discoveries of regularity and compressibility are concerned, it does not make an essential difference whether we force the system to predict the entire history of inputs and actions, or just parts thereof, or whether we allow it to focus on

internal computable abstractions thereof, like the system discussed in Section III-D. Partial compressibility of selected data covered by the system's limited focus of attention implies compressibility of the whole, even if most of it is random noise.

E. Discrete Asynchronous Framework for Maximizing Creativity Reward

Let $p(t)$ denote the agent's current compressor program at time t, $s(t)$ its current controller, and **DO**.

Controller: At any time t ($1 \leq t < T$) do

1) Let $s(t)$ use (parts of) history $h(\leq t)$ to select and execute $y(t+1)$.
2) Observe $x(t+1)$.
3) Check if there is nonzero creativity reward $r_{\rm int}(t+1)$ provided by the asynchronously running improvement algorithm of the compressor/predictor (see below). If not, set $r_{\rm int}(t+1) = 0$.
4) Let the controller's RL algorithm use $h(\leq t+1)$ including $r_{\rm int}(t+1)$ (and possibly also the latest available compressed version of the observed data—see below) to obtain a new controller $s(t+1)$, in line with objective (1). Note that some actions may actually trigger learning algorithms that compute changes of the compressor and the controller's policy, such as in Section III-D [79]. That is, the computational cost of learning can be taken into account by the reward optimizer, and the decision when and what to learn can be learned as well [79].

Compressor/Predictor: Set $p_{\rm new}$ equal to the initial data compressor/predictor. Starting at time 1, repeat forever until interrupted by death at time T.

1) Set $p_{\rm old} = p_{\rm new}$; get current time step t and set $h_{\rm old} = h(\leq t)$.
2) Evaluate $p_{\rm old}$ on $h_{\rm old}$, to obtain performance measure $C(p_{\rm old}, h_{\rm old})$. This may take many time steps.
3) Let some (possibly application-dependent) compressor improvement algorithm (such as a learning algorithm for an adaptive neural network predictor, possibly triggered by a controller action) use $h_{\rm old}$ to obtain a hopefully better compressor $p_{\rm new}$ (such as a neural net with the same size and the same constant computational effort per prediction, but with improved predictive power and therefore improved compression performance [101]). Although this may take many time steps (and could be partially performed offline during "sleep" [85], [96]), $p_{\rm new}$ may not be optimal, due to limitations of the learning algorithm, e.g., local maxima. (To inform the controller about beginnings of compressor evaluation processes etc., augment its input by unique representations of such events.)
4) Evaluate $p_{\rm new}$ on $h_{\rm old}$, to obtain $C(p_{\rm new}, h_{\rm old})$. This may take many time steps.
5) Get current time step τ and generate creativity reward

$$r_{\rm int}(\tau) = f[C(p_{\rm old}, h_{\rm old}), C(p_{\rm new}, h_{\rm old})] \tag{6}$$

for example, $f(a,b) = a - b$. [Here, the τ replaces the $t+1$ of (2)].

This asynchronuous scheme [85], [92], [96] may cause long temporal delays between controller actions and corresponding creativity rewards, and may impose a heavy burden on the controller's RL algorithm whose task is to assign credit to past actions. Nevertheless, Section II-G will discuss RL algorithms for this purpose which are theoretically optimal in various senses [85], [92], [96], [97].

F. Continuous Time Formulation

In continuous time, let $O(t)$ denote the state of subjective observer O at time t. The subjective simplicity or compressibility or regularity or beauty $B(D, O(t))$ of a sequence of observations and/or actions D is the negative number of bits required to encode D, given $O(t)$'s current limited prior knowledge and limited compression/prediction method. The observer-dependent and time-dependent subjective *interestingness* or *surprise* or *aesthetic value* $I(D, O(t))$ is

$$I(D, O(t)) \sim \frac{\partial B(D, O(t))}{\partial t} \tag{7}$$

the *first derivative* of subjective simplicity: as O improves its compression algorithm, formerly apparently random data parts become subjectively more regular and beautiful, requiring fewer and fewer bits for their encoding. Given its limited compression improver, at time t_0 the creativity goal of $O(t_0)$ is to select actions that will maximize

$$E\left[\int_{t=t_0}^{T} g[r_{\rm int}(t), r_{\rm ext}(t)] \partial t\right] \tag{8}$$

where E is an expectation operator [compare (1)]; T is death; $r_{\rm int}(t) = I(H(\leq t), O(t))$ is the momentary *fun* or *intrinsic reward* for compression progress through discovery of a novel pattern somewhere in $H(\leq t)$ (the history of actions and sensations until t); $r_{\rm ext}(t)$ is the current external reward if there is any; g is the function weighing external versus intrinsic rewards, e.g., $g(a,b) = a + b$ [99].

Note that there are at least two ways of having fun: execute a learning algorithm that improves the compression of the already known data (in online settings: without increasing computational needs of the compressor/predictor), or execute actions that generate more data, then learn to compress/understand the new data better.

G. Optimal Creativity, Given the Predictor's Limitations

The previous sections discussed how to measure compressor/predictor improvements and how to translate them into intrinsic reward signals, but did not say much about the RL method used to maximize expected future reward. The chosen predictor/compressor class typically will have certain computational limitations. In the absence of any external rewards, one may define *optimal pure curiosity behavior* relative to these limitations: At discrete time step t this behavior would select the action that maximizes

$$u(t) = E_\mu\left[\sum_{\tau=t+1}^{T} r_{\rm int}(\tau) \;\middle|\; h(\leq t)\right]. \tag{9}$$

Since the true, world-governing probability distribution μ is unknown, the resulting task of the controller's RL algorithm may be a formidable one. As the system is revisiting previously incompressible parts of the environment, some of those will tend to become more subjectively compressible, and while the corresponding curiosity rewards may first go up, they will eventually decrease once the new regularity has been learned. A good RL algorithm must somehow detect and then *predict* this decrease, and act accordingly. Traditional RL algorithms [35], however,

do not provide any theoretical guarantee of optimality for such situations.

Is there a best possible, universal RL algorithm that comes as close as any other computable one to maximizing objective (9)? Indeed, there is. Its drawback, however, is that it is not computable in finite time. Nevertheless, it serves as a reference point for defining what is achievable at best, that is, what is *optimal* creativity. Readers who are not interested in the corresponding theory may skip the remainder of this section and jump immediately to the practical implementations of Section III. For the others, the next paragraphs will outline how the universal approaches work. Optimal inductive inference as defined in Section II-D can be extended by formally including the effects of executed actions, to define an optimal action selector maximizing future expected reward. At any time t, Hutter's theoretically optimal (yet uncomputable) RL algorithm AIXI [32] uses such an extended version of Solomonoff's scheme to select those action sequences that promise maximal future reward up to some horizon T (e.g., twice the lifetime so far), given the current data $h(\leq t)$. That is, in cycle $t+1$, AIXI selects as its next action the first action of an action sequence maximizing ξ-predicted reward up to the given horizon, appropriately generalizing (5). AIXI uses observations optimally [32]: the Bayes-optimal policy p^{ξ} based on the mixture ξ is self-optimizing in the sense that its average utility value converges asymptotically for all $\mu \in \mathcal{M}$ to the optimal value achieved by the Bayes-optimal policy p^{μ} which knows μ in advance. The necessary and sufficient condition is that $\mathcal{M}$ admits self-optimizing policies. The policy p^{ξ} is also Pareto-optimal in the sense that there is no other policy yielding higher or equal value in *all* environments $\nu \in \mathcal{M}$ and a strictly higher value in at least one [32].

AIXI as above needs unlimited computation time. Its computable variant AIXI(t, l) [32] has asymptotically optimal runtime but may suffer from a huge constant slowdown. To take the consumed computation time into account in a general, optimal way, one may use the recent Gödel machine s [83], [84], [86], [98] instead. They represent the first class of mathematically rigorous, fully self-referential, self-improving, general, optimally efficient problem solvers, and are applicable to the problem embodied by objective (9). The initial software $\mathcal{S}$ of such a Gödel machine contains an initial problem solver, e.g., some typically suboptimal RL method [35]. It also contains an asymptotically optimal initial proof searcher, typically based on an online variant of Levin's *Universal Search* [43], which is used to run and test *proof techniques*. Proof techniques are programs written in a universal language implemented on the Gödel machine within $\mathcal{S}$. They are in principle able to compute proofs concerning the system's own future performance, based on an axiomatic system $\mathcal{A}$ encoded in $\mathcal{S}$. $\mathcal{A}$ describes the formal *utility* function, in the present case (9), the hardware properties, axioms of arithmetic and probability theory and data manipulation etc, and $\mathcal{S}$ itself, which is possible without introducing circularity [98]. Inspired by Kurt Gödel's celebrated self-referential formulas (1931), the Gödel machine rewrites any part of its own code (including the proof searcher) through a self-generated executable program as soon as its *Universal Search* variant has found a proof that the rewrite is *useful* according to objective (9). According to the Global Optimality Theorem [83], [84], [86], [98], such a self-rewrite is globally optimal—no local maxima possible—since the self-referential code first had to prove that it is not useful to continue the search for alternative self-rewrites. If there is no provably useful optimal way of rewriting $\mathcal{S}$ at all, then humans will not find one either. But if there is one, then $\mathcal{S}$ itself can find and exploit it. Unlike the previous nonself-referential methods based on hardwired proof searchers [32], Gödel machines not only boast an optimal *order* of complexity, but can optimally reduce (through self-changes) any slowdowns hidden by the $O()$-notation, provided the utility of such speed-ups is provable [87], [89], [91].

1) Limitations of the "Universal" Approaches: The methods above are optimal in various ways, some of them not only computable, but even optimally time-efficient in the asymptotic limit. Nevertheless, they leave open an essential remaining practical question: If the agent can execute only a fixed number of computational instructions per unit time interval (say, 10 trillion elementary operations per second), what is the best way of using them to get as close as possible to the theoretical limits of universal AIs? Especially when external rewards are very rare, as is the case in many realistic environments? As long as there is no good answer this question, one has to resort to approximations and heuristics when it comes to practical applications. The next section reviews what has been achieved so far along these lines, discussing our implementations of IM-based agents from the 1990s; quite a few aspects of these concrete systems are still of relevance today.

III. Previous Implementations of Intrinsically Motivated Agents: Pros and Cons

The above mathematically rigorous framework for optimal curiosity and creativity (2006–2010) was established *after* first approximations thereof were implemented (1991, 1995, and 1997–2002). Sections III-A–III-D will discuss advantages and limitations of online learning systems described in the original publications on artificial intrinsic motivation [70], [71], [77], [111], which already can be viewed as example implementations of a compression progress drive or prediction progress drive that encourages the discovery or creation of surprising patterns. Some elements of this earlier work are believed to remain essential for creating systems that are both theoretically sound and *practical*.

A. Intrinsic Reward for Prediction Error (1990)

Early work [67], [71] describes a predictor based on an adaptive world model implemented as a RNN (in principle a rather powerful computational device, even by today's machine learning standards), predicting sensory inputs including reward signals from the entire previous history of actions and inputs. A second RNN (the controller) uses the world model and gradient descent to search for a control policy or program maximizing the sum of future expected rewards according to the model. Some of the rewards are intrinsic curiosity rewards, which are proportional to the predictor's errors. So the same mechanism that is used for normal goal-directed learning is used for implementing creativity and curiosity and boredom—there is no need for a separate system aiming at improving the world model.

This first description of a general, curious, world-exploring RL agent implicitly and optimistically assumes that the predictor will indeed improve by motivating the controller to go to places where the prediction error is high. One drawback of the prediction error-based approach is that it encourages the

controller to focus its search on those parts of the environment where there will always be high prediction errors due to noise or randomness, or due to computational limitations of the predictor. This may *prevent* learning progress instead of promoting it, and motivates the next subsection, whose basic ideas could be combined with the RL method of [67], [71], but this has not been done yet.

Another potential drawback is the nature of the particular RNN-based RL method. Although the latter has the potential to learn internal memories of previous relevant sensory inputs, and thus is not limited to Markovian interfaces between agent and environment [72], like all gradient-based methods it may suffer from local minima, as well as from potential problems of online learning, since gradients for the recurrent RL controller are computed with the help of the dynamically changing, online learning recurrent predictive world model. Apart from this limitation, the RNN of back then were less powerful than today's long short-term memory (LSTM) RNN [28], [100], which yielded state of the art performance in challenging applications such as connected handwriting recognition [24], and should be used instead.

B. Intrinsic Reward for World Model Improvements (1991)

Follow-up work [69], [70] points out that one should not focus on the errors of the predictor, but on its improvements. The basic principle can be formulated as follows: *Learn a mapping from actions (or action sequences) to the expectation of future performance improvement of the world model. Encourage action sequences where this expectation is high.* This is essentially the central principle of Section II-A.

Two implementations were described: The first models the reliability of the predictions of the adaptive predictor by a separate, so-called confidence network. At any given time, reinforcement for the model-building control system is created in proportion to the current *change* or *first derivative* of the reliability of the adaptive predictor. The "curiosity goal" of the control system (it might have additional "prewired" external goals) is to maximize the expectation of the cumulative sum of future positive or negative changes in prediction reliability.

The second implementation replaces the confidence network by a network H which at every time step is trained to predict the current *change* of the model network's output due to the model's learning algorithm. That is, H will learn to approximate the expected *first derivative* of the model's prediction error, given the inputs. The *absolute value* of H's output is taken as the intrinsic reward, thus rewarding learning progress.

While the neural predictor of the implementations is computationally less powerful than the recurrent one of Section III-A [71], there is a novelty, namely, an explicit (neural) adaptive model of the predictor's improvements, measured in terms of mean squared error (MSE). This model essentially learns to predict the predictor's changes (the prediction derivatives). For example, although noise is unpredictable and leads to wildly varying target signals for the predictor, in the long run these signals do not change the adaptive predictor's parameters much, and the predictor of predictor changes is able to learn this. A variant of the standard RL algorithm Q-learning [114] is fed with curiosity reward signals proportional to the expected long-term predictor changes; thus the agent is intrinsically motivated to make novel patterns within the given limitations. In fact, one may say that the system tries to maximize an approximation of the (discounted) sum of the expected first derivatives of the data's subjective predictability, thus also maximizing an approximation of the (discounted) sum of the expected changes of the data's subjective compressibility (the surprise or novelty).

Both variants avoid the theoretically desirable but impractical regular evaluations of the predictor on the entire history so far, as discussed in Section II-B. Instead they monitor the recent effects of learning on the learning mechanism (a neural network in this case). Experiments illustrate the advantages of this type of directed, curious exploration over traditional random exploration.

One RL method-specific drawback is given by the limitations of standard Markovian RL [72], which assumes the current input tells the agent everything it needs to know, and does not work well in realistic scenarios where it has to learn to memorize previous relevant inputs to select optimal actions. For general robots scenarios more powerful RL methods are necessary, such as those mentioned in Section III-A and other parts of the present paper.

Any RL algorithm has to deal with the fact that intrinsic rewards vanish where the predictor becomes perfect. In the simple toy world [69], [70] this is not a problem, since the agent continually updates its Q-values based on recent experience. But since the learning rate is chosen heuristically (as usual in RL applications), this approach lacks the theoretical justification of the general framework of Section II.

For probabilistic worlds there are prediction error measures that are more principled than MSE. This motivates research described next.

C. Intrinsic Reward Depending on the Relative Entropy Between Agent's Prior and Posterior (1995)

Follow-up work (1995) describes an information theory-oriented variant of the approach in nondeterministic worlds [111]. Here, the curiosity reward is proportional to the predictor's surprise/information gain [15], measured as the Kullback–Leibler distance [39] between the learning predictor's subjective probability distributions on possible next events before and after new observations—the relative entropy between its prior and posterior, essentially another measure of learning progress. Again, experiments show the advantages of this type of curious exploration over conventional random exploration.

Since this implementation also uses a traditional RL method [114] instead of a more general one, the discussion of RL method-specific drawbacks in previous subsections remains valid here as well.

Note the connection to Section II: the concepts of Huffman coding [31] and relative entropy between prior and posterior immediately translate into a measure of learning progress reflecting the number of saved bits—a measure of improved data compression.

Note also, however, a drawback of this naive probabilistic approach to data compression: it is unable to discover more general types of *algorithmic* compressibility [45] as discussed in Section II. For example, the decimal expansion of π looks

random and incompressible but is not: there is a very short algorithm computing all of π, yet any finite sequence of digits will occur in π's expansion as frequently as expected if π were truly random, that is, no simple statistical learner will outperform random guessing at predicting the next digit from a limited time window of previous digits. More general *program* search techniques are necessary to extract the underlying algorithmic regularity. This motivates the universal approach discussed in Section II, but also the research on a more general practical implementation described next.

D. Learning Programs and Skills Through Zero Sum Intrinsic Reward Games (1997–2002)

The universal variants of the principle of novel pattern creation of Section II focused on theoretically optimal ways of measuring learning progress and fun, as well as mathematically optimal ways of selecting action sequences or experiments within the framework of artificial creativity [85], [92], [96], [97]. These variants take the entire lifelong history of actions and observations into account, and make minimal assumptions about the nature of the environment, such as: the (unknown) probabilities of possible event histories are at least enumerable. The resulting systems exhibit "mathematically optimal curiosity and creativity" and provide a yardstick against which all less universal intrinsically motivated systems can be measured. However, most of them ignore important issues of time constraints in online settings. For example, in practical applications one cannot frequently measure predictor improvements by testing predictor performance on the entire history so far. The costs of learning and testing have to be taken into account. This insight drove the research discussed next.

To address the computational costs of learning, and the costs of measuring learning progress, computationally powerful controllers and predictors [77], [79] were implemented as two very general, coevolving, symmetric, opposing modules called the *right brain* and the *left brain*, both able to construct self-modifying probabilistic programs written in a universal programming language (1997–2002). An internal storage for temporary computational results of the programs is viewed as part of the changing environment. Each module can suggest experiments in the form of probabilistic algorithms to be executed, and make predictions about their effects, *betting intrinsic reward* on their outcomes. The opposing module may accept such a bet in a zero-sum game by making a contrary prediction, or reject it. In case of acceptance, the winner is determined by executing the algorithmic experiment and checking its outcome; the intrinsic reward eventually gets transferred from the surprised loser to the confirmed winner. Both modules try to maximize their intrinsic reward using a rather general RL algorithm (the SSA [103]) designed for complex stochastic policies—alternative RL algorithms could be plugged in as well. Thus both modules are motivated to discover *truly novel* algorithmic patterns, where the dynamically changing subjective baseline for novelty is given by what the opponent already knows about the (external or internal) world's repetitive patterns. Since the execution of any computational or physical action costs something (as it will reduce the cumulative reward per time ratio), both modules are motivated to focus on those parts of the dynamic world that currently make learning progress *easy*, to minimize the costs of identifying promising experiments and executing them. The system learns a partly hierarchical structure of more and more complex skills or programs necessary to solve the growing sequence of self-generated tasks, reusing previously acquired simpler skills where this is beneficial. Experimental studies [79] exhibit several sequential stages of emergent developmental sequences, with and without external reward.

Many ingredients of this system may be just what one needs to build *practical yet sound* curious and creative systems that never stop expanding their knowledge about what can be done in a given world, although future reimplementations should probably use alternative reward optimizers that are more general and powerful than SSA [103], such as variants of the optimal ordered problem solver [82].

E. Improving Real Reward Intake (1991–2010)

The references above demonstrated in several experiments that the presence of intrinsic reward or curiosity reward can actually speed up the collection of *external* reward.

However, the previous papers also pointed out that it is always possible to design environments where the bias towards regularities introduced through artificial curiosity can lead to worse performance—curiosity can indeed kill the cat.

IV. Relation to Work by Others

A. Beyond Traditional Information Theory

How does the notion of surprise in the theory of creativity differ from the notion of surprise in traditional information theory? Consider two extreme examples of uninteresting, unsurprising, boring data: A vision-based agent that always stays in the dark will experience an extremely compressible, soon totally predictable history of unchanging visual inputs. In front of a screen full of white noise conveying a lot of information and "novelty" and "surprise" in the traditional sense of Boltzmann and Shannon [106], however, it will experience highly unpredictable and fundamentally incompressible data. As pointed out since the early 1990s, according to the theory of creativity, in both cases the data is not *surprising*, but *boring* [79], [92] as it does not allow for further compression progress—there is no novel pattern. Therefore the traditional notion of surprise is rejected. Neither the arbitrary nor the fully predictable is *truly* novel or surprising. Only data with still *unknown* algorithmic regularities are [70], [71], [79], [85], [92], [96], [97], [111], for example, a previously unknown song containing a subjectively novel harmonic pattern. That's why one really has to measure the *progress of the learning predictor* to compute the degree of surprise. (Compare Section IV-E2 for a related discussion on what's aesthetically pleasing.)

B. Beyond Traditional Active Learning

How does the theory generalize the traditional field of **active learning**, e.g., [15]? To optimize a function may require expensive data evaluations. Original active learning is limited to supervised classification tasks, e.g., [2], [12], [15], [33], [47], [55], and [105], asking which data points to evaluate next to maximize information gain, typically (but not necessarily) using one

step look-ahead, assuming all data point evaluations are equally costly. The objective (to improve classification error) is given externally; there is no explicit intrinsic reward in the sense discussed in the present paper. The more general framework of creativity theory also takes the following formally into account.

1) Reinforcement learning agents embedded in an environment where there may be arbitrary delays between experimental actions and corresponding information gains, e.g., [70] and [111].
2) The highly environment-dependent costs of obtaining or creating, not just individual data points, but data *sequences* of *a priori* unknown size.
3) Arbitrary algorithmic or statistical dependencies in sequences of actions and sensory inputs, e.g., [79] and [85].
4) The computational cost of learning new skills, e.g., [79].

While others recently have started to study active RL as well, e.g., Brafman and Tennenholtz (R-MAX Algorithm [10]), Li *et al.*(KWIK-framework [44]), and Strehl *et al.* [112], our more general systems measure and maximize *algorithmic* [37], [45], [80], [110] novelty (learnable, but previously unknown compressibility or predictability) of self-generated spatio–temporal patterns in the history of data and actions [85], [92], [96], [97].

C. Relation to Hand-Crafted Interestingness

Lenat's discovery system EURISKO [41], [42] has a preprogrammed interestingness measure which was observed to become more an more inappropriate ("stagnation" problem) as EURISKO created new concepts from old ones with the help of human intervention. Unsupervised systems based on creativity theory, however, continually redefine what's interesting based on what's currently easy to learn, in addition to what's already known.

D. Related Implementations Since 2005

In 2005, Baldi and Itti demonstrated experimentally that our method of 1995 (Section III-C, [111]) explains certain patterns of human visual attention better than certain previous approaches [34]. Their web site http://ilab.usc.edu/surprise/ (retrieved on March 17, 2010) points out that the approaches of Section III-C [111] and [34] are formally identical.

Klyubin *et al.*'s seemingly related approach to intrinsic motivation [36] of 2005 tries to maximize *empowerment* by maximizing the information an agent could potentially "inject" into its future sensory inputs via a sequence of actions. Unlike our 1995 method (Section III-C, [111]), this approach does not maximize information *gain*; in fact, the authors assume a good world model is already given, or at least learned before *empowerment* is measured (Polani, personal communication, 2010). For example, using one step look-ahead in a deterministic and well-modeled world, their agent will prefer states where the execution of alternative actions will make a lot of difference in the immediate sensory inputs, according to the already reliable world model. Generally speaking, however, it might prefer actions leading to high-entropy, random inputs over others—compare Section III-A.

In 2005, Singh *et al.* [107] also used intrinsic rewards proportional to prediction errors as in Section III-A [71], employing a different type of reward maximizer based on the option framework which can be used to specify subgoals. As pointed out earlier, it is useful to make the conceptual distinction between the objective and the means of reaching the objective: The latter is shared by the approaches of [107] and of Section III-A, the reward maximizer is different.

In related work, Schembri *et al.* address the problem of learning to compose skills, assuming different skills are learned by different RL modules. They speed up skill learning by rewarding a top level, module-selecting RL agent in proportion to the TD error of the selected module [63]—compare Section III-B.

Other researchers in the nascent field of developmental robotics [9], [20], [26], [27], [38], [51], [52], [57], [64], [113] and intrinsic reward also took up the basic idea, for example, Oudeyer *et al.* [53]. They call their method "intelligent adaptive curiosity" (IAC), reminiscent of our original 1991 paper on "adaptive curiosity" (AC) [69] (Section III-B). Like AC, IAC motivates the agent to go where it can expect learning progress with a high derivative. Oudeyer *et al.* write that IAC is "intelligent" because it *"keeps, as a side effect, the robot away both from situations which are too predictable and from situations which are too unpredictable."* That is what the original AC does (Section III-B). However, IAC is less general than AC in the following sense: IAC is restricted to one-step look-ahead, and does not allow for delayed intrinsic rewards. That is, even a small short-term intrinsic reward will be more attractive to IAC than many huge long-term rewards. Nonetheless, an interesting aspect of IAC's greedy reward maximizer, is that it splits the state space into regions, reminiscent of algorithms by Doya [14] and Moore [49]; this might make learning more robust in certain situations.

Oudeyer *et al.*'s Section III-A Group 1 on "Error Maximization" [53] covers some of the topics discussed in the first paper on this subject: [71] (our Section III-A). Their Section III-B Group 2 on "Progress Maximization" addresses issues discussed in the first papers on this subject: [69], [70], [111] (our Section III-B and Section III-C). Referring to [70] in their Section III-C Group 3 on "Similarity-Based Progress Maximization," Oudeyer *et al.* [53] write:

> *"Schmidhuber...provided initial implementations of artificial curiosity, but [was] not concerned with the emergent development sequence and with the increase of the complexity of their machines...They were only concerned in how far artificial curiosity can speed up the acquisition of knowledge."*

However, emergent development sequences with and without external rewards (and several sequential stages) were studied in follow-up papers (1997–2002) [77], [79] (Section III-D) containing action frequency plots similar to those of Oudeyer *et al.* (2007). These papers also address many other issues such as continuous states (within the limits of floating point precision), whose importance is emphasized by Oudeyer *et al.*, who also write:

"Another limit of this work resides within the particular formula that is used to evaluate the learning progress associated with a candidate situation, which consists of making the difference between the error in the anticipation of this situation before it has been experienced and the error in the anticipation of exactly the same situation after it has been experienced. On the one hand, this can only work for a learning machine with a low learning rate, as pointed out by the author, and will not work with, for example, one-shot learning of memory-based methods. On the other hand, considering the state of the learning machine just before and just after one single experience can possibly be sensitive to stochastic fluctuations."

However, the 1991 AC system of Section III-B is in fact precisely designed to deal with stochastic fluctuations: in states where the next input is random and unpredictable, the learning predictor's targets will fluctuate stochastically, and the system will notice this, as there is no measurable learning progress (just small predictor changes that cancel each other). And the general 2006 systems [85] (Section II) do not have any problems of the criticized type as long as the predictor's performance is always measured on the entire history so far. Oudeyer *et al.* [53] also write:

"The question of whether hierarchical structures can simply self-organize without being explicitly programmed remains open,"

apparently being unaware of previous work on hierarchical RL systems that can discover their own subgoals [1], [60], [61], [79], [102], [118].

Friston *et al.* [19] (2010) also propose an approach which, in many ways, seems similar to ours, based on free energy minimization and predictive coding. Predictive coding is a special case of compression, e.g., [101], and free energy is another approximative measure of algorithmic compressibility/algorithmic information [45]; the latter concept is more general though. As Friston *et al.* write: *"Under simplifying assumptions free energy is just the amount of prediction error",* like in the 1991 paper [71] discussed in Section III-A. Under slightly less simplifying assumptions it is the Kullback–Leibler divergence between probabilistic world model and probabilistic world, like in the 1995 paper [111] (which looks at the learning model before and after new observations; see Section III-C). Despite these similarities, however, what Friston *et al.* do is to select actions that *minimize* free energy. In other words, their agents like to visit highly predictable states. As the authors write:

"Perception tries to suppress prediction error by adjusting expectations to furnish better predictions of signals, while action tries to fulfil these predictions by changing those signals...In summary, under active inference, perception tries to explain away prediction errors by changing predictions, while action tries to explain them away by changing the signals being predicted."

Hence, although Friston *et al.*'s approach shares buzzwords with the methods of Sections III-A–III-C, (active data selection, reinforcement learning, relative entropy, Kullback–Leibler divergence), they do *not* describe a system intrinsically motivated to learn new, previously unknown things; instead their agents really want to stabilize and make everything predictable. Friston *et al.* are well aware of potential objections: *"At this point, most (astute) people say: but that means I should retire to a dark room and cover my ears."* This pretty much sums up the expected criticism. In contrast, the theory of creativity has no problem whatsoever with dark rooms—the latter get boring as soon as they are predictable; then there is no learning progress no more, that is, the first derivative of predictability/compressibility is zero, that is, the intrinsic reward is zero, that is, the reward-maximizing agent is motivated to leave the room to find or make additional rewarding, nonrandom, learnable, novel patterns.

Recent related work in the field of evolutionary computation aims at increasing diversity within populations of individuals [21], [23], [40]. This can be done by measuring the "novelty" of their behaviors [21], [23] using compression distance [11], based on the idea that compressing the concatenation of similar behaviors is cheaper than compressing them separately.

E. Previous, Less Formal Work in Aesthetics Theory and Psychology

Two millennia ago, Cicero already called curiosity a "passion for learning." In the recent millennium's final century, art theorists and developmental psychologists extended this view. In its final decade, the concept eventually became sufficiently formal to permit the computer implementations discussed in Section III.

1) Developmental Psychology: In the 1950s Berlyne and other psychologists revisited the idea of curiosity as the motivation for exploratory behavior [5], [6], emphasizing the importance of novelty [5] and nonhomeostatic drives [25]. Piaget [54] explained explorative learning behavior of children through his concepts of assimilation (new inputs are embedded in old schemas—this may be viewed as a type of compression) and accommodation (adapting an old schema to a new input—this may be viewed as a type of compression improvement). All those ideas were informal, without providing details necessary to permit the construction of artificially curious agents.

2) Aesthetics Theory: The closely related field of aesthetics theory [4], [7], [16], [18], [48], [50] emerged even earlier in the 1930s. Why are some objects, such as works of art, more interesting or aesthetically rewarding than others? Why are humans somehow intrinsically motivated to observe them, even when they seem totally unrelated to solving typical problems such as hunger, and even when the action of observation requires a serious effort, such as spending hours to get to the museum? Some of the previous attempts at explaining aesthetic experience in the context of information theory [4], [7], [16], [18], [48], [50] tried to quantify the intrinsic aesthetic reward through the idea of an *"ideal"* ratio between expected and unexpected information conveyed by some aesthetic object (its *"order"* versus its *"complexity"*). For example, using certain measures based on information theory [106], Bense [4] argued for an ideal ratio

of $1/e \sim 37\%$. Generally speaking, however, these approaches also were not detailed and formal enough to construct artificial, intrinsically motivated, creative agents.

The theory of fun and creativity does not have to postulate an objective ideal ratio of this kind. Instead, and unlike some of the previous works that already emphasized the significance of the subjective observer [16]–[18], its dynamic formal measure of interestingness reflects the *change* in the number of bits required to encode an object, and explicitly takes into account the subjective observer's prior knowledge, as well as its limited compression *improvement* algorithm. Hence, the value of an aesthetic experience is not defined by the observed object *per se*, but by the algorithmic compression *progress* (or prediction *progress*) of the subjective, learning observer.

Why did not early pioneers of aesthetic information theory put forward similar views? Perhaps because back then the fields of algorithmic information theory and adaptive compression through machine learning were still in their infancy?

V. Simple Typology of Intrinsic Motivation

After pointing out problems of a previous typology [52], this section will provide a natural one without those problems, addressing current confusion as to what exactly should be called intrinsic reward, clarifying that this concept is orthogonal to: 1) secondary reward in RL economies; 2); internal reward for speeding up RL; 3) internal rewards for subgoals in hierarchical RL; and 4) evolution of reward functions, since all of the above are driven by external reward.

A. Problems With a Previous Typology

A recently published classification of computational intrinsic motivation [52] mentions a fraction of the relevant literature since 1990, and classifies it in a way that may introduce unnecessary complexity, hiding the fact that the basic principles of intrinsic motivation are general and simple. The proposed classes of [52] are: 1) knowledge-based models of intrinsic motivation; 1a) information theoretic and distributional models; 1b) predictive models; 1c) learning progress; 2) competence-based models of intrinsic motivation; 2a) maximizing incompetence; 2b) maximizing competence; 3) morphological models of intrinsic motivation; 3a) synchronicity motivation; and 3b) stability and variance motivation.

Closer inspection reveals that 1a) is a special case of 1b) (the probabilistic predictors/models of information theory are special types of predictors), and many instances of 1a) (such as maximizing information gain) are simultaneously special cases of 1c) (learning progress). So it does not seem to make sense to have 1a), 1b), and 1c) on the same level. It should be mentioned, however, that the authors originally intended to present at least 1c) as a special case of 1b)—misleading section labels were erroneously inserted by the editors (Oudeyer, personal communication, 2010).

In their section on "morphological models," the authors seem to make again a conceptual distinction between statistical/information-theoretic predictors and other predictors of the earlier section "knowledge-based models." Statistical knowledge, however, predicts probability distributions on possible events, instead of single, deterministic events, which are a special case. Likewise, synchronicity and stability [3a), 3b)] are special cases of predictability (and therefore compressibility). For example, given two synchronous event streams, one can trivially predict the timing of the first from the timing of the second.

The authors of [52] originally intended to present 2a) and 2b) as examples of 2), not as subcategories (Oudeyer, personal communication, 2010). Nevertheless, there is no obvious essential difference between 2) and 1), as most instances of 2) and 1) are again special cases of models that try to improve prediction mismatches (or, more generally, compressibility). To see this, note that a general predictor or compressor will try to predict/compress all accessible data including sensory inputs, reinforcement signals, executed action sequences, e.g., Section II [85]. To test behavioral competence, one must somehow compare predicted and actual outcome of some action sequence (e.g., execute robot behavior – does the final state match a predicted subgoal representation?). To test knowledge, one must do the same (e.g., move eyes here – do the properties of the resulting sensory input match the prediction?). Here is a quote from [52]: *"A second major computational approach to intrinsic motivation is based on measures of competence that an agent has for achieving self-determined results or goals. Interestingly, this approach has not yet been studied in the computational literature."* However, this is precisely what was done in several implementations of the 1990s discussed in Sections III-B–III-D [70], [77], [79], [111]. These systems had goals that included self-determined goals, namely, to execute action sequences yielding data that allowed their predictive models to improve; their RL methods simply measured competence by the amount of intrinsic reward they obtained. In particular, the system of Section III-D ([77], [79]) could design general algorithmic experiments (programs) including all kinds of computable predictions. This encompasses all kinds of computable competence tests and knowledge tests.

B. Alternative Natural Typology

Here, a conceptually simpler typology is proposed. It essentially just reflects the scheme from the introduction, and does not suffer from the problems above.

By definition, intrinsic reward is something that is independent of external reward, although it may sometimes help to accelerate the latter as discussed in Section III-E ([70], [79], [111]). So far, most if not all intrinsically motivated computational systems had the following:

1) a more or less limited adaptive predictor/compressor/model of the history of sensory inputs, internal states, reinforcement signals, and actions;
2) some sort of real-valued intrinsic reward indicative of the learning progress of 1);
3) a more or less limited reinforcement learner able to maximize future expected reward.

Hence, the typology just needs to classify previous systems with respect to properties and limitations of their specific instances of (1–3). In the spirit of MDL, we describe a compact model (in this case: a typology) of the data (in this case: various approaches to IM) by identifying what the majority of the previous IM approaches have in common.

1) Includes many subtypes characterized by the answers to the following questions.

a) What exactly can the predictor predict (or the compressor compress)?
   i) All sensory inputs as in Section III-A [71]? A preprocessed subset of the sensory inputs? For example, features indicating synchronicity of certain processes [52]? The latter may be of interest for certain limited types of IM-based learning.
   ii) Reinforcement signals as in Section III-A [71]? (Even traditional RL agents without IM do this.)
   iii) Controller actions as in Section II [79], [85], [92], [96], [97]? Then even in absence of sensory feedback, curious and creative agents will be motivated to learn new motor patterns, such as previously unknown dances.
   iv) Results of internal computations through sequences of internal actions as in Section III-D [79]? This will motivate a curious agent to create novel patterns not only in the space of sensory inputs but also in the space of abstract input transformations, such as earlier learned mappings from images of cars to an internal symbol "car". The agent will also be motivated to create purely "mental" novel patterns independent of external inputs, such as number sequences obeying previously unknown mathematical laws (corresponding to mathematical discoveries).
   v) Some combination of the above? All of the above as in Section III-D [79]? The latter should be the default for artificial general intelligences (AGIs).

b) Is the predictor deterministic as in Section III-A [71], or does it predict probability distributions on possible events as in Section III-C [111]?

c) How are the predictor and its learning algorithm implemented?
   i) Is the predictor actually a continually changing, growing 3-D model or simulation of the agent in the environment, used to predict future visual or tactile inputs, given agent actions (Section II-B)?
   ii) Is it a traditional machine learning model? A feedforward neural network mapping pairs of actions and observations to predictions of the next observation as in Section III-B [70]? A recurrent neural network that is in principle able to deal with event histories of arbitrary size as in Section III-A [71]? A Gaussian Process? A Support Vector Machine? A Hidden Markov Model? Etc.

2) includes many subtypes characterized by the answers to the following questions.

a) Is the entire history used to evaluate the predictor's performance as in Section II [85], [92], [96], [97] (in theory the correct thing to do, but sometimes impractical)? Or only recent data, e.g., the one acquired at the present time step as in Section III-B [70], or in a limited time window of recent inputs? (If so, a performance decline on earlier parts of the history may go unnoticed.)

b) Which measure is used to indicate learning progress and create intrinsic reward?
   i) Mean squared prediction error or similar measures as in Section III-A [3], [36], [71], [107]? This may fail whenever high prediction errors do not imply expected prediction progress, e.g., in noisy environments, but also when the limitations of the predictor's learning algorithm prevent learning progress even in deterministic worlds.
   ii) Improvements (first derivatives) of prediction error as in Section III-B [52], [70]? This properly deals with both noisy/non-deterministic worlds and the computational limitations of the predictor/compressor.
   iii) The information-theoretic Kullback–Leibler divergence (a.k.a. relative entropy) [39] between belief distributions before and after learning steps, as in Section III-C [34], [111]? A well-founded approach, at least under the assumption that all potential statistical dependencies between inputs can indeed be modeled by the given probabilistic model, which in previous implementations (Section III-C) was limited to singular events [34], [111] as opposed to arbitrary event sequences, for efficiency reasons.
   iv) MDL-based measures [62], [109], [110], [115], [116] comparing the number of bits required to encode the observation history before and after learning steps, as in Section II [85], [92], [96], [97]? Unlike the methods above, this approach automatically punishes unnecessarily complex predictors/compressors that overfit the data, and can easily deal with long event sequences instead of simple 1 step events. For example, if the predictor uses a 3-D world model or simulation, the MDL approach will ask (Section II-B): how many bits are currently needed to specify all polygons in the simulation, and how many bits are needed to encode deviations of the sensory history from the predictions of the 3-D simulation? Adding or removing polygons may reduce the total number of bits (and decrease future prediction errors).

c) Is the computational effort of the predictor and its learning algorithm taken into account when measuring its performance, as in Section II-B [77], [79], [85]? The only implementation of this

(Section III-C; [77], [79]) still lacks theoretical optimality guarantees.

d) Which are the relative weights of external and intrinsic reward? This is of importance as long as the latter does not vanish in environments where after some time *nothing new* can be learned any more.

3) includes many subtypes characterized by the answers to the following questions.

a) Which is the action repertoire of the controller?

i) Can it produce only external motor actions, as in Section III-B [70], [111]?

ii) Can it also manipulate an internal mental state through internal actions as in Section III-D ([77], [79]), thus being able to deal not only with raw sensory inputs but also with internal abstractions thereof, and to create/discover novel purely mathematical patterns, like certain theoreticians who sometimes do not care much about the external world?

iii) Can it trigger learning processes by itself, by executing appropriate actions as in Section III-D ([77], [79])? This is important for learning when to learn and what to learn, trading off the costs of learning versus the expected benefits in terms of intrinsic and extrinsic rewards.

b) Which are the perceptive abilities of the controller?

i) Can it choose at any time to see any element of the entire history [85] of all sensory inputs, rewards, executed actions, internal states? Or only a subset thereof, possibly a recent one, as in Section III-B [70]? The former should be the default for AGIs.

ii) Does it have access to the parameters and internal state of the predictor, like in Section III-D [79]? Or just a subset thereof? Such introspective abilities are important to predict future intrinsic rewards which depend on the already existing knowledge encoded in the predictor.

c) Which optimizer of expected intrinsic and extrinsic reward is used?

i) A traditional Q-learner [117] able to deal with delayed rewards as long as the environment is fully observable, like in Section III-B? A more limited 1-step look-ahead learner [52] that will break down in presence of delayed intrinsic rewards? A more sophisticated RL algorithm for delayed rewards in partially observable environments [35], [72], like in Section III-A? A hierarchical, subgoal-learning RL algorithm [1], [60], [61], [102], [118] or perhaps other hierarchical methods that do not learn to create subgoals by themselves [3], [13], [107]?

ii) An action planner using a 3-D simulation of the world to generate reward-promising trajectories (see MDL example in Section II-B)?

iii) An evolutionary algorithm [22], [29], [59], [104] applied to recurrent neural networks [22] or other devices that compute action sequences?

iv) One of the recent universal, mathematically optimal RL algorithms [32], [98], like in Section II-G? Variants of universal search [43] or its incremental extension, the Optimal Ordered Problem Solver [82]?

v) Something else? Obviously lots of alternative search methods can be plugged in here.

d) How does the system deal with problems of online learning?

i) Action sequences producing patterns that used to be novel do not get rewarded any more once the patterns are known. Can the practical reward optimizer reliably deal with this problem of vanishing rewards, like the theoretically optimal systems of Section II-G?

ii) Can the reward optimizer actually use the continually improving predictive world model to improve or speed up the search for a better policy? This is automatically done by the above-mentioned action planner using a continually improving 3-D world simulation, and also by the RNN-based world model of the system in Section III-A [71]. Does the changing model cause problems of online learning? Are those problems dealt with in a heuristic way (e.g., small learning rates), or in a theoretically sound way as in Section II-G?

Each node or leaf of the typology above can be further expanded, thus becoming the root of additional straightforward refinements. But let us now address some of the recent confusion surrounding the concept of intrinsic motivation, and clarify what it is *not*.

C. Secondary Reward as an Orthogonal Issue

Reward propagation procedures of traditional RL such as Q-learning [117] or RL economies and bucket brigade systems [30], [65], [66], [120] may be viewed as translating *rare* external rewards for achieving some goal into *frequent* internal rewards for earlier actions setting the stage. Should one call these internal "secondary" rewards intrinsic rewards? Of course not. They are just internal by-products of the method used to maximize *external* reward, which remains the only measure of overall success.

D. Speeding Up RL as an Orthogonal Issue

Many methods have been proposed to speed up traditional RL. Some Q-learning accelerators simply update pairs of actions and states with currently quickly changing Q-values more frequently than others (that is, Q-values with high first derivatives are favored). Others postpone updates until needed [119]. Again, one should resist the temptation to confuse such types of secondary reward modulation with intrinsic reward, because the only thing important to such methods is the *external* reward. (Otherwise one would also have to call intrinsic reward many of the things that could be invented by any (possibly universal [32], [98]) RL method whose only goal is to maximize expected *external* reward.)

E. Subgoal Learning as an Orthogonal Issue

Some goal-seeking RL systems search a space of possible subgoal combinations, internally rewarding subsystems whose policies learn to achieve those subgoals [1], [61], [102], [118]. Essentially, they seek useful reward functions for the subsystems. External reward (for reaching a final goal) is used to measure the quality of subgoal combinations: good subgoals survive, others are discarded. Again the internal reward for the subsystems should not be called intrinsic reward, as it is totally driven and justified by *external* reward.

F. Evolution of Reward Functions as an Orthogonal Issue

Essentially, the same argument holds for very similar methods that search a space of reward functions until they find one that helps a given RL method to achieve more reward more quickly, e.g., [46] and [108]. Such methods are like the subgoal evolvers [118] of Section V-E which also evolve or search for useful reward functions. The results of this search should not be called intrinsic reward functions, since once more the only thing that counts here is the *external* reward; the rest is just implementation details of the external reward maximizer.

But did not humans evolve to have such an intrinsic reward function? Sure, they did, but now it is there, and now it is independent of external reward, otherwise it would not be intrinsic reward, by definition. Scientific papers on intrinsic reward should start from there. It is a different issue to analyze how and why evolution or another search process *invented* intrinsic rewards to facilitate satisfaction of *external* goals (such as survival).

VI. How the Theory Explains Art, Science, and Humor

How does the prediction progress drive/compression progress drive explain **humor**? Consider the following statement: *Biological organisms are driven by the "Four Big F's": Feeding, Fighting, Fleeing, Mating.* Some subjective observers who read this for the first time think it is funny. Why? As the eyes are sequentially scanning the text the brain receives a complex visual input stream. The latter is subjectively partially compressible as it relates to the observer's previous knowledge about letters and words and their semantics. That is, given the reader's current knowledge and current compressor, the raw data can be encoded by fewer bits than required to store random data of the same size. But the punch line after the last comma is unexpected for those who expected another "F." Initially, this failed expectation results in suboptimal data compression—storage of expected events does not cost anything, but deviations from predictions require extra bits to encode them. The compressor, however, does not stay the same forever: within a short time interval its learning algorithm kicks in and improves the performance on the data seen so far, by discovering the nonrandom, nonarbitrary and therefore, compressible pattern relating the punch line to previous text and to the observer's previous elaborate, predictive knowledge about the "Four Big F's." This prior knowledge helps to compress the whole history including the punch line a bit better than before, which momentarily saves a few bits of storage, that is, there is quick learning progress, that is, fun. The number of saved bits (or a similar measure of learning progress) becomes the observer's intrinsic reward, possibly strong enough to motivate him to read on in search for more reward through additional yet unknown patterns.

While most previous attempts at explaining humor (e.g., [58]) also focus on the element of surprise, they lack the essential concept of *novel pattern detection* measured by compression *progress* due to learning. This progress is zero whenever the unexpected is just random noise, and thus no fun at all. Applications of the new theory of humor can be found in recent videos [95].

How does the theory informally explain the motivation to create or perceive **art and music** [75], [76], [85], [92], [94], [96], [97]? For example, why are some melodies more interesting or aesthetically rewarding than others? Not the one the listener (composer) just heard (played) twenty times in a row. It became too subjectively predictable in the process. Nor the weird one with completely unfamiliar rhythm and tonality. It seems too irregular and contains too much arbitrariness and subjective noise. The observer (creator) of the data is interested in melodies that are unfamiliar enough to contain somewhat unexpected harmonies or beats etc., but familiar enough to allow for quickly recognizing the presence of a new learnable regularity or compressibility in the sound stream: a novel pattern. Sure, it will get boring over time, but not yet. All of this perfectly fits the principle: The current compressor of the observer or data creator tries to compress his history of acoustic and other inputs where possible. The action selector tries to find history-influencing actions such that the continually growing historic data allows for improving the compressor's performance. The interesting or aesthetically rewarding musical and other subsequences are precisely those with previously unknown yet learnable types of regularities, because they lead to compressor improvements. The boring patterns are those that are either already perfectly known or arbitrary or random, or whose structure seems too hard to understand.

Similar statements not only hold for other dynamic art including film and dance (take into account the compressibility of action sequences), but also for "static" art such as painting and sculpture, created through action sequences of the artist, and perceived as dynamic spatio–temporal patterns through active attention shifts of the observer. When not occupied with optimizing *external* reward, artists and observers of art are just following their compression progress drive.

The previous computer programs discussed in Section III already incorporated (approximations of) the basic creativity prin-

ciple, but do they really deserve to be viewed as rudimentary artists and scientists? The patterns they create are novel with respect to their own limited predictors and prior knowledge, but not necessarily relative to the knowledge of sophisticated adults. The main difference to human artists/scientists, however, may be only quantitative by nature, not qualitative.

- The unknown learning algorithms of human predictors/compressors are presumably still better suited to real world data. Recall, however, that there already exist *universal,* mathematically optimal (but not necessarily practically feasible) prediction and compression algorithms (Section II-D; [32], [98]), and that ongoing research is continually producing better and better *practical* prediction and compression methods, waiting to be plugged into the creativity framework.
- Humans may have superior reinforcement learning algorithms for maximizing rewards generated through compression improvements achieved by their predictors. Recall, however, that there already exist *universal,* mathematically *optimal* (but not necessarily practically feasible) reward optimizing algorithms (Section II-G; [32], [98]), and that ongoing research is continually producing better and better *practical* reinforcement learning methods, also waiting to be plugged into the creativity principle.
- Renowned human artists and scientists have had decades of training experiences involving a multitude of high-dimensional sensory inputs and motoric outputs, while our systems so far only had a few hours with very low-dimensional experiences in limited artificial worlds. This quantitative gap, however, will narrow as IM researchers are scaling up their systems.
- Human brains still have vastly more raw computational power and storage capacity than the best artificial computers. Note, however, that this statement is unlikely to remain true for more than a few decades – currently each decade brings a hardware speed-up factor of roughly 100–1000.

Current computational limitations of artificial artists do not prevent us from already using the basic principle in human-computer interaction to create art appreciable by humans—see example applications in references [76], [85], [92], [94], [96], and [97].

How does the theory explain the nature of **inductive sciences such as physics**? If the history of the entire universe were computable, and there is no evidence against this possibility [88], then its simplest explanation would be the shortest program that computes it. Unfortunately, there is no general way of finding the shortest program computing any given data [45]. Therefore, physicists have traditionally proceeded incrementally, analyzing just a small aspect of the world at any given time, trying to find simple laws that allow for describing their limited observations better than the best previously known law, essentially trying to find a program that compresses the observed data better than the best previously known program. An unusually large compression breakthrough deserves the name *discovery*. For example, Newton's law of gravity can be formulated as a short piece of code which allows for substantially compressing many observation sequences involving falling apples and other objects. Although its predictive power is limited—for example, it does not explain quantum fluctuations of apple atoms—it still allows for greatly reducing the number of bits required to encode the data stream, by assigning short codes to events that are predictable with high probability [31] under the assumption that the law holds. Einstein's general relativity theory yields additional compression progress as it compactly explains many previously unexplained deviations from Newton's predictions. Most physicists believe there is still room for further advances, and this is what is driving them to invent new experiments unveiling novel, previously unpublished patterns [94], [96], [97]. When not occupied with optimizing *external* reward, physicists are also just following their compression progress drive!

VII. Concluding Remarks and Outlook

To build a creative agent that never stops generating nontrivial and novel and surprising data, we need two learning modules: 1) an adaptive predictor or compressor or model of the growing data history as the agent is interacting with its environment; and 2) a general reinforcement learner. The *learning progress* of 1) is the *fun* or intrinsic reward of 2). That is, 2) is motivated to invent things that 1) does not yet know, but can easily learn.

While purely curious and creative behaviors aim at maximizing expected fun or surprise through the creation of novel patterns, the relevance of all behaviors with respect to prewired or *external* goals is measured by (delayed) external reward. Recent work has led to the first RL machines that are universal and optimal in various very general senses [32], [81], [98]—see Section II-G. Such machines can in theory find out by themselves whether curiosity and creativity are useful or useless in a given environment, and learn to behave accordingly. In realistic settings, however, external rewards are extremely rare, and one cannot expect quick progress of this type, not even by optimal machines. But typically one can learn lots of useful behaviors even in absence of external rewards: unsupervised behaviors that just lead to predictable or compressible results and thus reflect the regularities in the environment, e.g., repeatable patterns in the world's reactions to certain action sequences. In this paper the assumption is that a bias towards exploring previously unknown environmental regularities is *a priori* good in the real world as we know it, and should be inserted into practical AGIs, whose goal-directed learning will profit from this bias, in the sense that behaviors leading to external reward can often be quickly composed/derived from previously learned, purely curiosity-driven behaviors. We did not worry about the undeniable possibility that curiosity and creativity can actually be harmful and "kill the cat," assuming the environment is "benign enough." Based on experience with the real world it may be argued that this assumption is realistic. The resulting explorative bias greatly facilitates the search for goal-directed behaviors in environments where the acquisition of external reward has indeed a lot to do with easily learnable environmental regularities.

It may be possible to formally quantify this bias towards novel patterns in form of a mixture-based prior [32], [45], [81], [110], a weighted sum of probability distributions on sequences of actions and resulting inputs, and derive precise conditions for improved expected external reward intake. Intrinsic reward may be viewed as analogous to a *regularizer* in supervised

learning, where the prior distribution on possible hypotheses greatly influences the most probable interpretation of the data in a Bayesian framework [8] (for example, the well-known synapse decay term of neural networks is a consequence of a Gaussian prior with zero mean for each synapse). Note, however, that there is a difference to traditional regularizers with *a priori* fixed relative weights (also known as hyper-parameters): intrinsic reward for learning progress eventually vanishes in environments where after some time *nothing new* can be learned any more; that is, the intrinsic reward eventually becomes negligible where the sources of external reward do not run dry as well (e.g., no daily food no more). Following Section VI, some of the AGIs based on the creativity principle will become scientists, artists, or comedians.

ACKNOWLEDGMENT

The author would like to thank B. Kuipers, H. W. Franke, M. Hutter, A. Barto, J. Lansey, M. Littman, J. Togelius, F. J. Gomez, G. Pezzulo, G. Baldassarre, M. Butz, M. Looks, and M. Ring for useful comments that helped to improve this paper, or earlier papers on this subject.

REFERENCES

[1] B. Bakker and J. Schmidhuber, F. Groen, Ed. *et al.*, "Hierarchical reinforcement learning based on subgoal discovery and subpolicy specialization," in *Proc. 8th Conf. Intell. Autonom. Syst.*, Amsterdam, The Netherlands, 2004, pp. 438–445.

[2] M. F. Balcan, A. Beygelzimer, and J. Langford, "Agnostic active learning," *J. Comput. Syst. Sci.*, vol. 75, no. 1, pp. 78–89, 2009.

[3] A. G. Barto, S. Singh, and N. Chentanez, "Intrinsically motivated learning of hierarchical collections of skills," in *Proc. Int. Conf. Develop. Learn.*, Cambridge, MA, 2004.

[4] M. Bense, *Einführung in die Informationstheoretische Ästhetik. Grundlegung und Anwendung in der Texttheorie (Introduction to Information-Theoretical Aesthetics. Foundation and Application to Text Theory)*. Berlin, Germany: Rowohlt Taschenbuch Verlag, 1969.

[5] D. E. Berlyne, "Novelty and curiosity as determinants of exploratory behavior," *Brit. J. Psychol.*, vol. 41, pp. 68–80, 1950.

[6] D. E. Berlyne, *Conflict, Arousal, and Curiosity*. New York: McGraw–Hill, 1960.

[7] G. D. Birkhoff, *Aesthetic Measure*. Cambridge, MA: Harvard Univ. Press, 1933.

[8] C. M. Bishop, *Neural Networks for Pattern Recognition*. London, U.K.: Oxford Univ. Press, 1995.

[9] D. Blank and L. Meeden, "Introduction to the special issue on developmental robotics," *Connect. Sci.*, vol. 18, no. 2, 2006.

[10] R. I. Brafman and M. Tennenholtz, "R-MAX—A general polynomial time algorithm for near-optimal reinforcement learning," *J. Mach. Learn. Res.*, vol. 3, pp. 213–231, 2002.

[11] R. Cilibrasi and P. Vitanyi, "Clustering by compression," *IEEE Trans. Inf. Theory*, vol. 51, no. 4, pp. 1523–1545, Apr. 2005.

[12] D. A. Cohn, "Neural network exploration using optimal experiment design," in *Advances in Neural Information Processing Systems*, J. Cowan, G. Tesauro, and J. Alspector, Eds. San Mateo, CA: Morgan Kaufmann, 1994, vol. 6, pp. 679–686.

[13] P. Dayan and G. Hinton, "Feudal reinforcement learning," in *Advances in Neural Information Processing Systems*, D. S. Lippman, J. E. Moody, and D. S. Touretzky, Eds. San Mateo, CA: Morgan Kaufmann, 1993, vol. 5, pp. 271–278.

[14] K. Doya, K. Samejima, K. Katagiri, and M. Kawato, "Multiple model-based reinforcement learning," *Neural Comput.*, vol. 14, no. 6, pp. 1347–1369, 2002.

[15] V. V. Fedorov, *Theory of Optimal Experiments*. New York: Academic, 1972.

[16] H. G. Frank, *Kybernetische Analysen Subjektiver Sachverhalte*. Quickborn, Germany: Verlag Schnelle, 1964.

[17] H. G. Frank and H. W. Franke, *Ästhetische Information. Estetika informacio. Eine Einführung in die Kybernetische Ästhetik*. Berlin, Germany: Kopäd Verlag, 2002.

[18] H. W. Franke, *Kybernetische Ästhetik. Phänomen Kunst*, 3rd ed. Munich, Germany: Ernst Reinhardt Verlag, 1979.

[19] K. J. Friston, J. Daunizeau, J. Kilner, and S. J. Kiebel, "Action and behavior: A free-energy formulation," *Biol. Cybern.*, vol. 102, no. 3, pp. 227–260, 2010.

[20] K. Gold and B. Scassellati, "Learning acceptable windows of contingency," *Connect. Sci.*, vol. 18, no. 2, 2006.

[21] F. J. Gomez, "Sustaining diversity using behavioral information distance," in *Proc. Conf. Genet. Evol. Comput.*, Montreal, QC, Canada, 2009, pp. 113–120.

[22] F. J. Gomez, J. Schmidhuber, and R. Miikkulainen, "Efficient nonlinear control through neuroevolution," *J. Mach. Learn. Res.*, vol. 9, pp. 937–965, 2008.

[23] F. J. Gomez, J. Togelius, and J. Schmidhuber, "Measuring and optimizing behavioral complexity," in *Proc. Int. Conf. Artif. Neural Netw.*, Lamissol, Cyprus, 2009, pp. 765–774.

[24] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for improved unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, May 2009.

[25] H. F. Harlow, M. K. Harlow, and D. R. Meyer, "Novelty and curiosity as determinants of exploratory behavior," *J. Exp. Psychol.*, vol. 41, pp. 68–80, 1950.

[26] S. Hart, "The Development of Hierarchical Knowledge in Robot Systems," Ph.D. dissertation, Dept. Comput. Sci., Univ. Massachusetts, Amherst, MA, 2009.

[27] S. Hart, S. Sen, and R. Grupen, "Intrinsically motivated hierarchical manipulation," in *Proc. IEEE Conf. Robot. Autom.*, Pasadena, CA, 2008.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.

[30] J. H. Holland, "Properties of the bucket brigade," in *Proc. Int. Conf. Genet. Algorithms*, Hillsdale, NJ, 1985.

[31] D. A. Huffman, "A method for construction of minimum-redundancy codes," *Proc. IRE*, vol. 40, pp. 1098–1101, 1952.

[32] M. Hutter, *Universal Artificial Intelligence: Sequential Decisions Based on Algorithmic Probability*. Berlin, Germany: Springer-Verlag, 2004.

[33] J. Hwang, J. Choi, S. Oh, and R. J. Marks, II, "Query-based learning applied to partially trained multilayer perceptrons," *IEEE Trans. Neural Netw.*, vol. 2, pp. 131–136, Feb. 1991.

[34] L. Itti and P. F. Baldi, "Bayesian surprise attracts human attention," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2005, vol. 19, pp. 547–554.

[35] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. AI Res.*, vol. 4, pp. 237–285, 1996.

[36] A. S. Klyubin, D. Polani, and C. L. Nehaniv, "Empowerment: A universal agent-centric measure of control," in *Proc. Int. Conf. E-Commerce*, Müchen, Germany, 2005.

[37] A. N. Kolmogorov, "Three approaches to the quantitative definition of information," *Problems Inform. Trans.*, vol. 1, pp. 1–11, 1965.

[38] B. Kuipers, P. Beeson, J. Modayil, and J. Provost, "Bootstrap learning of foundational representations," *Connect. Sci.*, vol. 18, no. 2, 2006.

[39] S. Kullback, *Statistics and Information Theory*. New York: Wiley, 1959.

[40] J. Lehman and K. O. Stanley, "Exploiting open-endedness to solve problems through the search for novelty," in *Proc. 11th Int. Conf. Artif. Life*, Cambridge, MA, 2008.

[41] D. B. Lenat, "Theory formation by heuristic search," *Mach. Learn.*, vol. 21, 1983.

[42] D. B. Lenat and J. S. Brown, "Why AM an EURISKO appear to work," *Artif. Intell.*, vol. 23, no. 3, pp. 269–294, 1984.

[43] L. A. Levin, "Universal sequential search problems," *Problems Inform. Trans.*, vol. 9, no. 3, pp. 265–266, 1973.

[44] L. Li, M. L. Littman, and T. J. Walsh, "Knows what it knows: A framework for self-aware learning," in *Proc. 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, 2008.

[45] M. Li and P. M. B. Vitányi, *An Introduction to Kolmogorov Complexity and its Applications*, 2nd ed. Berlin, Germany: Springer-Verlag, 1997.

[46] M. L. Littman and D. H. Ackley, R. K. Belew and L. Booker, Eds., "Adaptation in constant utility non-stationary environments," in *Proc. 4th Int. Conf. Genet. Algorithms*, San Mateo, CA, 1991, pp. 136–142.

[47] D. J. C. MacKay, "Information-based objective functions for active data selection," *Neural Comput.*, vol. 4, no. 2, pp. 550–604, 1992.

[48] A. Moles, *Information Theory and Esthetic Perception*. Chicago, IL: Univ. Illinois Press, 1968.

[49] A. Moore and C. Atkeson, "The parti-game algorithm for variable resolution reinforcement learning in multidimensional state-spaces," *Mach. Learn.*, vol. 21, 1995.

[50] F. Nake, *Ästhetik als Informationsverarbeitung*. Berlin, Germany: Springer-Verlag, 1974.

[51] L. Olsson, C. L. Nehaniv, and D. Polani, "From unknown sensors and actuators to actions grounded in sensorimotor perceptions," *Connect. Sci.*, vol. 18, no. 2, 2006.

[52] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? A typology of computational approaches," *Frontiers Neurorobot.*, vol. 1, 2006.

[53] P.-Y. Oudeyer, F. Kaplan, and V. F. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE Trans. Evol. Comput.*, vol. 11, pp. 265–286, Nov. 2007.

[54] J. Piaget, *The Child's Construction of Reality*. London, U.K.: Routledge Kegan Paul, 1955.

[55] M. Plutowski, G. Cottrell, and H. White, "Learning Mackey-Glass from 25 examples, plus or minus 2," in *Advances in Neural Information Processing Systems*, J. Cowan, G. Tesauro, and J. Alspector, Eds. San Mateo, CA: Morgan Kaufmann, 1994, vol. 6, pp. 1135–1142.

[56] J. Poland and M. Hutter, "Strong asymptotic assertions for discrete MDL in regression and classification," in *Proc. Annu. Mach. Learn. Conf. Belgium Netherlands*, Enschede, The Netherlands, 2005.

[57] J. Provost, B. J. Kuipers, and R. Miikkulainen, "Developing navigation behavior through self-organizing distinctive state abstraction," *Connect. Sci.*, vol. 18, no. 2, 2006.

[58] V. Raskin, *Semantic Mechanisms of Humor*. Dordrecht, Germany: Springer-Verlag, 1985.

[59] I. Rechenberg, "Evolutionsstrategie – Optimierung Technischer Systeme Nach Prinzipien der Biologischen Evolution," Ph.D. dissertation, Technische Universität Berlin, Berlin, Germany, 1971.

[60] M. B. Ring, L. Birnbaum and G. Collins, Eds., "Incremental development of complex behaviors through automatic construction of sensory-motor hierarchies," in *Proc. 8th Int. Workshop Mach. Learn.* , Evanston, IL, 1991, pp. 343–347.

[61] M. B. Ring, "Continual Learning in Reinforcement Environments," Ph.D. dissertation, Univ. Texas, Austin, TX, 1994.

[62] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[63] M. Schembri, M. Mirolli, and G. Baldassarre, Y. Demiris, B. Scassellati, and D. Mareschal, Eds., "Evolving internal reinforcers for an intrinsically motivated reinforcement-learning robot," in *Proc. 6th IEEE Int. Conf. Develop. Learn.*, London, U.K., 2007, pp. 282–287.

[64] M. Schlesinger, "Decomposing infants' object representations: A dual-route processing account," *Connect. Sci.*, vol. 18, no. 2, 2006.

[65] J. Schmidhuber, "A local learning algorithm for dynamic feedforward and recurrent networks," *Connect. Sci.*, vol. 1, no. 4, pp. 403–412, 1989.

[66] J. Schmidhuber, "The neural bucket brigade," in *Connectionism in Perspective*, R. Pfeifer, Z. Schreter, Z. Fogelman, and L. Steels, Eds. Amsterdam, The Netherlands: Elsevier, 1989, pp. 439–446.

[67] J. Schmidhuber, "Dynamische neuronale Netze und das fundamentale raumzeitliche Lernproblem," Ph.D. dissertation, Technische Universität München, München, Germany, 1990.

[68] J. Schmidhuber, "An on-line algorithm for dynamic reinforcement learning and planning in reactive environments," in *Proc. IEEE/INNS Int. Joint Conf. Neural Netw.*, San Diego, CA, 1990, vol. 2, pp. 253–258.

[69] J. Schmidhuber, Adaptive Curiosity and Adaptive Confidence Technische Universität München, Institut für Informatik, 1991, Tech. Rep. FKI-149-91.

[70] J. Schmidhuber, "Curious model-building control systems," in *Proc. Int. Joint Conf. Neural Netw.*, Singapore, 1991, vol. 2, pp. 1458–1463.

[71] J. Schmidhuber, J. A. Meyer and S. W. Wilson, Eds., "A possibility for implementing curiosity and boredom in model-building neural controllers," in *Proc. Int. Conf. Simulation Adapt. Behav.: From Animals to Animats*, 1991, pp. 222–227.

[72] J. Schmidhuber, "Reinforcement learning in Markovian and non-Markovian environments," in *Advances in Neural Information Processing Systems 3*, D. S. Lippman, J. E. Moody, and D. S. Touretzky, Eds. San Mateo, CA: Morgan Kaufmann, 1991, vol. NIPS 3, pp. 500–506.

[73] J. Schmidhuber, "Learning complex, extended sequences using the principle of history compression," *Neural Comput.*, vol. 4, no. 2, pp. 234–242, 1992.

[74] J. Schmidhuber, "A computer scientist's view of life, the universe, and everything," in *Foundations of Computer Science: Potential – Theory – Cognition*, C. Freksa, M. Jantzen, and R. Valk, Eds. Berlin, Germany: Springer-Verlag, 1997, vol. 1337, Lecture Notes in Computer Science, pp. 201–208.

[75] J. Schmidhuber, *Femmes Fractales*, 1997.

[76] J. Schmidhuber, "Low-complexity art," *Leonardo, J. Int. Soc. Arts, Sci. Technol.*, vol. 30, no. 2, pp. 97–103, 1997.

[77] J. Schmidhuber, What's Interesting? IDSIA, 1997, Technical Report IDSIA-35-97.

[78] J. Schmidhuber, Algorithmic Theories of Everything IDSIA, Manno, Lugano, Switzerland, 2000, Tech. Rep. IDSIA-20-00, quant-ph/0011122.

[79] J. Schmidhuber, "Exploring the predictable," in *Advances in Evolutionary Computing*, A. Ghosh and S. Tsuitsui, Eds. Berlin, Germany: Springer-Verlag, 2002, pp. 579–612.

[80] J. Schmidhuber, "Hierarchies of generalized Kolmogorov complexities and nonenumerable universal measures computable in the limit," *Int. J. Foundations Comput. Sci.*, vol. 13, no. 4, pp. 587–612, 2002.

[81] J. Schmidhuber, J. Kivinen and R. H. Sloan, Eds., "The Speed Prior: A new simplicity measure yielding near-optimal computable predictions," in *Proc. 15th Annu. Conf. Comput. Learn. Theory*, Sydney, Australia, 2002, pp. 216–228.

[82] J. Schmidhuber, "Optimal ordered problem solver," *Mach. Learn.*, vol. 54, pp. 211–254, 2004.

[83] J. Schmidhuber, "Completely self-referential optimal reinforcement learners," in *Artificial Neural Networks: Biological Inspirations – ICANN 2005*, W. Duch, J. Kacprzyk, E. Oja, and S. Zadrozny, Eds. Berlin, Germany: Springer-Verlag, 2005, vol. 3697, Lecture Notes in Computer Science, Plenary Talk, pp. 223–233.

[84] J. Schmidhuber, "Gödel machines: Towards a technical justification of consciousness," in *Adaptive Agents and Multi-Agent Systems III*, D. Kudenko, D. Kazakov, and E. Alonso, Eds. Berlin, Germany: Springer-Verlag, 2005, vol. 3394, Lecture Notes in Computer Science, pp. 1–23.

[85] J. Schmidhuber, "Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts," *Connect. Sci.*, vol. 18, no. 2, pp. 173–187, 2006.

[86] J. Schmidhuber, "Gödel machines: Fully self-referential optimal universal self-improvers," in *Artificial General Intelligence*, B. Goertzel and C. Pennachin, Eds. Berlin, Germany: Springer-Verlag, 2006, pp. 199–226.

[87] J. Schmidhuber, "The new AI: General & sound & relevant for physics," in *Artificial General Intelligence*, B. Goertzel and C. Pennachin, Eds. Berlin, Germany: Springer-Verlag, 2006, pp. 175–198.

[88] J. Schmidhuber, "Randomness in physics," *Nature*, vol. 439, no. 3, p. 392, 2006.

[89] J. Schmidhuber, "2006: Celebrating 75 years of AI – History and outlook: The next 25 years," in *50 Years of Artificial Intelligence*, M. Lungarella, F. Iida, J. Bongard, and R. Pfeifer, Eds. Berlin, Germany: Springer-Verlag, 2007, vol. 4850, Lecture Notes in Artificial Intelligence, pp. 29–41.

[90] J. Schmidhuber, "Alle berechenbaren Universen (All computable universes)," Transl.: German *Spektrum der Wissenschaft Spezial*, no. 3, pp. 75–79, 2007.

[91] J. Schmidhuber, "New millennium AI and the convergence of history," in *Challenges to Computational Intelligence*, W. Duch and J. Mziuk, Eds. Berlin, Germany: Springer-Verlag, 2007, vol. 63, Studies in Computational Intelligence, pp. 15–36.

[92] J. Schmidhuber, "Simple algorithmic principles of discovery, subjective beauty, selective attention, curiosity & creativity," in *Proc. 10th Int. Conf. Discovery Sci.*, Sendai, Japan, 2007, vol. 4755, Lecture Notes in Artificial Intelligence, pp. 26–38.

[93] J. Schmidhuber, "Driven by compression progress," in *Knowledge-Based Intelligent Information and Engineering Systems KES-2008*, I. Lovrek, R. J. Howlett, and L. C. Jain, Eds. Berlin, Germany: Springer-Verlag, 2008, vol. 5177, Lecture Notes in Computer Science, p. 11.

[94] J. Schmidhuber, "Art & science as by-products of the search for novel patterns, or data compressible in unknown yet learnable ways," in *Multiple Ways to Design Research. Research Cases that Reshape the Design Discipline*, M. Botta, Ed. Berlin, Germany: Springer-Verlag, 2009, pp. 98–112.

[95] J. Schmidhuber, Compression Progress: The Algorithmic Principle Behind Curiosity and Creativity (With Applications of the Theory of Humor), 40 min Video of Invited Talk at Singularity Summit 2009 New York City, 2009 [Online]. Available: http://www.vimeo.com/7441291

[96] J. Schmidhuber, "Driven by compression progress: A simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes," in *Anticipatory Behavior in Adaptive Learning Systems. From Psychological Theories to Artificial Cognitive Systems*, G. Pezzulo, M. V. Butz, O. Sigaud, and G. Baldassarre, Eds. Berlin, Germany: Springer-Verlag, 2009, vol. 5499, Lecture Notes in Computer Science, pp. 48–76.

[97] J. Schmidhuber, "Simple algorithmic theory of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes," *SICE J. Soc. Instrument Contr. Eng.*, vol. 48, no. 1, pp. 21–32, 2009.

[98] J. Schmidhuber, "Ultimate cognition à la Gödel," *Cogn. Comput.*, vol. 1, no. 2, pp. 177–193, 2009.

[99] J. Schmidhuber, "Artificial scientists & artists based on the formal theory of creativity," in *Artificial General Intelligence*, M. Hutter, Ed. *et al.* Berlin, Germany: Springer-Verlag, 2010.

[100] J. Schmidhuber, A. Graves, F. J. Gomez, and S. Hochreiter, *How to Learn Programs with Artificial Recurrent Neural Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2010, to be published.

[101] J. Schmidhuber and S. Heil, "Sequential neural text compression," *IEEE Trans. Neural Netw.*, vol. 7, pp. 142–146, Jul. 1996.

[102] J. Schmidhuber and R. Wahnsiedler, J. A. Meyer, H. L. Roitblat, and S. W. Wilson, Eds., "Planning simple trajectories using neural subgoal generators," in *Proc. 2nd Int. Conf. Simulation Adapt. Behav.*, Honolulu, HI, Dec. 7–11, 1992, pp. 196–202.

[103] J. Schmidhuber, J. Zhao, and M. Wiering, "Shifting inductive bias with success-story algorithm, adaptive Levin search, and incremental self-improvement," *Mach. Learn.*, vol. 28, pp. 105–130, 1997.

[104] H. P. Schwefel, "Numerische Optimierung von Computer-Modellen," Ph.D. dissertation, Technische Universität Berlin, Berlin, Germany, 1974/1975, Reprinted by Birkhäuser.

[105] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. 5th Annu. Workshop Comput. Learn. Theory*, New York, 1992, pp. 287–294.

[106] C. E. Shannon, "A mathematical theory of communication (Parts I and II)," *Bell Syst. Techn. J.*, vol. XXVII, pp. 379–423, 1948.

[107] S. Singh, A. G. Barto, and N. Chentanez, "Intrinsically motivated reinforcement learning," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2005, vol. 17, NIPS.

[108] S. Singh, R. L. Lewis, and A. G. Barto, N. Taatgen and H. van Rijn, Eds., "Where do rewards come from?," in *Proc. 31st Annu. Conf. Cogn. Sci. Soc.*, Austin, TX, 2009.

[109] R. J. Solomonoff, "A formal theory of inductive inference. Part I," *Inform. Contr.*, vol. 7, pp. 1–22, 1964.

[110] R. J. Solomonoff, "Complexity-based induction systems," *IEEE Trans. Inf. Theory*, vol. IT-24, pp. 422–432, 1978.

[111] J. Storck, S. Hochreiter, and J. Schmidhuber, "Reinforcement driven information acquisition in non-deterministic environments," in *Proc. Int. Conf. Artif. Neural Netw.*, Paris, France, 1995, vol. 2, pp. 159–164.

[112] A. Strehl, J. Langford, and S. Kakade, Learning from Logged Implicit Exploration Data 2010, Technical Report arXiv:1003.0120.

[113] D. Stronger and P. Stone, "Towards autonomous sensor and actuator model induction on a mobile robot," *Connect. Sci.*, vol. 18, no. 2, 2006.

[114] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[115] C. S. Wallace and D. M. Boulton, "An information theoretic measure for classification," *Comput. J.*, vol. 11, no. 2, pp. 185–194, 1968.

[116] C. S. Wallace and P. R. Freeman, "Estimation and inference by compact coding," *J. Roy. Statist. Soc., Series B*, vol. 49, no. 3, pp. 240–265, 1987.

[117] C. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3/4, pp. 279–292, 1992.

[118] M. Wiering and J. Schmidhuber, "HQ-learning," *Adapt. Behav.*, vol. 6, no. 2, pp. 219–246, 1998.

[119] M. A. Wiering and J. Schmidhuber, "Fast online $Q(\lambda)$," *Mach. Learn.*, vol. 33, no. 1, pp. 105–116, 1998.

[120] S. W. Wilson, "ZCS: A zeroth level classifier system," *Evol. Comput.*, vol. 2, pp. 1–18, 1994.

Jürgen Schmidhuber received the doctoral degree in computer science from the Technische Universitat München (TUM), München, Germany, in 1991 and the Habilitation degree in 1993, after a postdoctoral stay at the University of Colorado, Boulder.

He has been the Director of the Swiss Artificial Intelligence Lab (IDSIA) since 1995, a Professor of Artificial Intelligence at the University of Lugano, Lugano, Switzerland, since 2009, the Head of the CogBotLab at TUM since 2004, and a Professor at the Scuola Universitaria Professionale della Svizzera Italiana (SUPSI) since 2003. He helped to transform IDSIA into one of the world's top ten AI labs, according to the ranking of Business Week Magazine. In 2008, he was elected member of the European Academy of Sciences and Arts. He has published more than 200 peer-reviewed scientific papers on topics such as machine learning, mathematically optimal universal AI, artificial curiosity and creativity, artificial recurrent neural networks, adaptive robotics, algorithmic information and complexity theory, digital physics, theory of beauty, and the fine arts.