Goal Babbling with Unknown Ranges: A Direction-Sampling Approach

Matthias Rolf

Abstract—Goal babbling is a recent concept for the efficient bootstrapping of sensorimotor coordination that is inspired by infants’ early goal-directed movement attempts. Several studies have shown its superior performance compared to random motor babbling. Yet, previous implementations of goal babbling require knowledge of a set of achievable goals in advance. This paper introduces an approach to goal babbling that can bootstrap coordination skills without pre-specifying, or even representing, a set of goals. On the contrary, it can discover the ranges of achievable goals autonomously. This capability is demonstrated in a challenging task with up to 50 degrees of freedom, in which the discovery of possible outcomes is shown to be desperately intractable with random motor babbling.

Index Terms—Goal Babbling, Goal Selection, Inverse Models

I. INTRODUCTION

An efficient bootstrapping of coordination skills is important for both humans and developing robots. Many sensorimotor coordination tasks can be understood by a simple causality between actions $q$ in some potentially high-dimensional action space $Q$, and their observable outcomes $x$ in some observation space $X$. The causal relation between both sizes can be described by an implicit forward function $f$ that associates each action with its unique outcome (see Fig. 1). This very abstract notion can be used to describe reaching skills (e.g. with actions being robot joint angles and effector positions as outcomes), but also many other tasks like facial expression generation [1], playing golf [2], quadruped walking [3], dynamic throwing movements [4], or vocal articulation [5].

In all these scenarios, the agent is asked to achieve various goals, or desired outcomes $x^*$ that are situated inside $X$. The agent can not achieve this outcome directly, but has to learn an appropriate action $q$ that causes $x^*$. Goal babbling [6] is a recent concept that describes the learning of such skills based on consistent goal-directed exploration. It is inspired by early goal-directed movement attempts that have already been observed in newborns [7]. Several recent studies have shown the success, speed, and scalability of this approach in robotic coordination tasks [8], [9], [10], [11], [3], which also supports the significance of neonates’ early goal-directed movements.

If goal babbling is performed in the absence of external goals, the agent can internally choose goals and try to achieve them in order to bootstrap and practice a skill. Thereby the choice of which goals are tried to be achieved is crucial: all the present studies required to pre-specify a set of goals $X^* \subset X$ that the agent tries to achieve. It is desirable that the agent learns to achieve any possible outcome, i.e. any value $x$ in the observation space that can be generated by at least one action. If some value $x$ is not achievable, it would be a waste of precious resource to try and fail to achieve it. Denoting the workspace of achievable outcomes as

$$F = f(Q) = \{x \in X : \exists q \in Q : f(q) = x\} \, ,$$

it is desirable to choose $X^* \approx F$ as good as possible. In the case of reaching this workspace corresponds to the set of reachable hand positions. It is generally important for both humans and robots to know the ranges of this reachable area. For goal babbling it is important to choose the goals in a useful manner. Even with a fully learned (or analytically known) skill it is important to know whether an action can succeed or not. Yet, the ranges of $F$ are typically not known in advance, rather complicated, and tedious to configure manually. Even robots with known forward functions usually do not allow for a closed-form description of the workspace. Hence, the discovery of workspace ranges is an important learning problem.

A. Related Work

Answering the question what is achievable is tightly connected to the question how it is achievable. One can only certainly know that some outcome $x$ can be achieved if one knows a corresponding action $q$ with $f(q) = x$. Standard robotics setups typically provide ways to answer the “how” based on analytic knowledge which allows to determine the ranges of the workspace experimentally: First, a designer chooses a superset of possible outcomes. For instance a robot with revolute joints has a known maximum length, such that a cube with that edge-length certainly subsumes all possible

Fig. 1. An agent can execute actions in some high-dimensional action space, but is interested in achieving corresponding outcomes in an observation space. Describing which outcomes are possible at all is an important problem both for analytic robots, as well as efficient learning paradigms such as goal babbling.
outcomes. Then, analytic controllers [12] or stochastic search algorithms [13] can be used to find actions that reach for positions in that superset. Positions for which no appropriate actions are found are considered to be not reachable.

If an agent does not have such analytic knowledge it has to learn both “how” and “what”. An apparent straightforward approach is to perform random motor babbling [14], [15], [16]: the agent performs random actions and eventually discovers all possible outcomes. Such exhaustive exploration can not answer the “how” question in high-dimensional domains. Experiments in this paper will show that motor babbling also fails to answer the “what” question. In general it seems intractable to consider solely the action space in order to systematically guide the discovery inside the observation space.

In contrast to motor babbling, which starts exploration in the action space, goal babbling starts with goals in the lower dimensional observation space. Goal babbling has been shown to answer the “how” question very efficiently also in high dimensional action spaces [8]. Since it triggers exploration in the observation space, it appears to be a good candidate for an efficient discovery of this space as well. Yet, previous algorithms rely on prior knowledge about possible outcomes, instead of discovering them: at least a superset of achievable outcomes must be known, which are then considered as goals. The idea of this approach is to structure already early exploration along goals, which makes the prior need for such knowledge rather critical. In practice, it is often very difficult to design [17]. In some cases this problem can be alleviated by a clever choice of the coordinate system in which the outcomes are measured. Jamone et al. used a stereo-vision coordinate system for reaching [18]. In their setup the reachable workspace was nearly a cube in that coordinate system, which was easy to identify and contained only few positions that were not actually reachable. Intrinsic motivation approaches on top of goal babbling can also alleviate the problem. Competence-progress models [3] have been shown to work even if $X^\ast$ is erroneously chosen substantially larger than $F$: the approach can ultimately identify positions that are not achievable because they do not result in learning progress. Still, $X^\ast$ has to be explicitly designed before learning can start, and positions that are not contained in $X^\ast$ will not be considered, even if they are reachable.

**B. Overview**

This paper tackles the problem to apply efficient goal babbling without assuming any previously specified goals, which supersedes the tedious and rather implausible pre-structuring necessary for previous algorithms. Instead, the novel approach samples goals iteratively along continuous paths as long as the agent is able to approximately follow the desired movement direction. Thereby, the agent eventually discovers the entire workspace without explicitly representing it, and learns an inverse model to reach for any achievable outcome. The approach to iteratively sample goals along certain directions is introduced in Sec. II. This idea is exemplary implemented on the basis of the algorithm in [8], while being fully compatible with other implementations of goal babbling. The basic algorithm for goal babbling is introduced in Sec. III. Experiments in Sec. IV demonstrate the algorithm’s ability to identify the entire workspace and to learn a coordination skill for it even in challenging high-dimensional domains. In contrast to that efficient discovery, it is shown that random motor babbling drastically fails to discover the workspace.

**II. DIRECTION SAMPLING OF GOALS**

The observation space in which the agent tries to achieve goals is often entirely unbounded. In the case of reaching this space can be the three-dimensional cartesian space, in which the ranges of achievable outcomes are not a priori known. How can an agent initially choose goals in an entirely unknown and unbounded space?

Fig. 2(a) illustrates the basic idea of this paper to tackle this problem: the agent starts from some initial observation $x^{home}$ that is the result of some default action $q^{home}$. This position $x^{home}$ is the only outcome that is known to be achievable in the beginning, and is the first goal $x_0^\ast$ over the course of exploration. From this starting point, the agent randomly selects a direction $\Delta x$ which it tries to follow. In the following
time-steps \( t \), goals are chosen along this direction:

\[
x^*_t = x^*_{t-1} + \frac{\epsilon \Delta x}{\| \Delta x \|} \cdot \Delta x,
\]

where \( \epsilon \Delta x \) is some step-width. The crucial question is when to stop trying to move along this direction. To illustrate the general idea, consider an agent that already knows how to achieve the goals during exploration. In that case the agent would choose actions \( q_t \) for the presented goals that result in the observation of \( f(q_t) = x_t = x^*_t \). Hence, the observed movement and the desired movement match perfectly, \textit{unless} some goal \( x^*_t \) is not achievable. A sudden deviation between \( x_t \) and \( x^*_t \) can be seen as hint that \( x^*_t \) is outside \( \mathbb{F} \), while \( x^*_{t-1} \) is inside. The straightforward strategy in this case is to go back to \( x^*_{t-1} \), select a new random direction and follow it as long as the observed movement matches it. This series of movements with random directions will ultimately result in the full coverage of the workspace \( \mathbb{F} \), as indicated in Fig. 2(a).

A similar strategy is often exploited by autonomous vacuum cleaning robots [19]: they move along straight paths until they detect a collision with a wall. Then, they select a new random direction and ultimately cover an entire room.

When an agent just learns about its sensorimotor coordination, however, two aspects are more challenging than in the vacuum cleaning scenario: Firstly, it is not directly possible to detect a collision with a boundary, which a mobile robot navigating through a room can, for instance, do with a bump sensor. Secondly, the goals can not be perfectly achieved even if they are all achievable, simply because the agent has not yet learned \textit{how} to achieve them. This difference is particularly substantial for high dimensional action spaces, in which an agent has to first find suitable commands, while a vacuum cleaning robot has a low-dimensional action space whose relation to the observation space is perfectly known. A movement pattern that is more realistic for a learning scenario is shown in Fig. 2(b): the robot selects some desired movement direction \( \Delta x \) from which it instantaneously deviates because it has not fully learned the appropriate actions. The critical question is \textit{when to stop} following \( \Delta x \) and to choose a new goal direction. Since any movement that just roughly leads into the right direction can be useful, a criterion can be defined based on the angles of desired and action movement. The agent stops going along one direction if the observed movement deviates by more than \( 90^\circ \) from the intended one. This can be easily expressed in terms of a negative scalar product:

\[
\text{Stop following } \Delta x \text{ if: } (x^*_t - x^*_{t-1})^T \cdot (x_t - x_{t-1}) < 0. \tag{3}
\]

When this condition is detected, the agent returns to \( x^*_{t-1} \) and selects a new random direction. In the beginning this will most often generate very short paths that do not cover the full workspace. As soon as the agent learns the necessary coordination, it will be able to follow paths through the entire workspace, and to sample paths similar to those with perfect coordination shown in Fig. 2(a).

### III. Online Goal Babbling

The direction sampling of goals describes a strategy to choose \textit{what} goals are tried to be achieved. The complementary question is \textit{how} they can be achieved, which is necessary to eventually identify the ranges of the achievable workspace. The direction sampling approach is generally compatible with different implementations of goal babbling [9], [10], [11], [3] that have recently been suggested. This paper implements this idea on the basis of the algorithm in [8]. This section briefly introduces this algorithm, and explains its integration and functioning with the direction sampling of goals.

The algorithm for goal babbling learns an \textit{inverse model} \( g(x^*) \) of the worlds forward function \( f \). This to be learned inverse model suggests an action \( q = g(x^*) \) for any presented goal. If learning eventually succeeds, the model is an inverse function of \( f \) such that it can achieve all possible outcomes:

\[
f(g(x^*)) = x^* \quad \forall \ x^* \in \mathbb{F}. \tag{4}
\]

The algorithm starts with an initial motor action \( q^{\text{home}} \). The inverse estimate \( q \) has parameters \( \theta \) adaptable by learning, and is initialized in \( t = 0 \) such that it always suggests the default action \( g(x^*, \theta_0) = q^{\text{home}} \lor x^* \). Starting from this state, goals are chosen with the direction sampling method and the algorithm tries to reach for them with the inverse estimate:

\[
q_t = g(x^*_t, \theta_t) + E_t(x^*_t). \tag{5}
\]

The perturbation term \( E_t(x^*_t) \) adds exploratory noise in order to discover novel outcomes. The inverse model \( g() \) can be any parameterized function approximator. This paper uses a locally linear learner (LLM) [20]. Parameters and meta-parameters of this setup are all identical to [8]. The outcome \( x_t = f(q_t) \) of each chosen action \( q_t \) is observed and a learning step is immediately performed after an example has been generated.
Thereby the inverse model is fitted to the example \((x_t, q_t)\) by gradient-descent on the weighted action error

\[ E^C_{w_t} = w_t \cdot ||q_t - g(x_t)||^2. \]

This online learning “in the loop” with exploration constitutes a positive feedback loop between exploration and learning that allows to achieve very high learning speed [8]. Efficient movements along the continuous paths receive a higher weight

\[ w_t = ||x_t - x_{t-1}|| \cdot ||q_t - q_{t-1}||^{-1}, \]

which causes the inverse model to likewise select efficient solutions that can be effectively extrapolated in further exploration steps, and that avoid the generation of inconsistent examples that would corrupt the data-fitting [6]. Assuming that inconsistent examples are avoided, fitting the examples will eventually lead to an inverse model that can achieve the observed outcomes \(f(g(x_t)) = x_t\) [21]. Hence, if the observations cover the entire workspace, the algorithm will learn an inverse model according to Eqn. 4.

An additional, and biologically highly plausible, mechanism prevents uncontrolled drifts of the inverse estimate and selects a way to resolve the redundancy in coordination tasks (see also [22]). Instead of performing only goal-directed movements, the system returns to its “home” action \(q^{\text{home}}\) after a while. With a probability of \(p^{\text{home}} = 0.1\), the next movement after one direction according to Eqn. 2 has been sampled goes back to this state by interpolating from the last performed action \(q_{t-1}\) towards \(q^{\text{home}}\):

\[ q_t = q_{t-1} + \frac{\epsilon^Q}{||q^{\text{home}} - q_{t-1}||} \cdot (q^{\text{home}} - q_{t-1}). \quad (6) \]

Once \(q^{\text{home}}\) has been reached, the system starts from \(x_{t+1}^* = f(q^{\text{home}})\) and samples goal-directed paths again.

IV. WORKSPACE DISCOVERY IN HIGH DIMENSIONS

This section evaluates how well the proposed algorithm is able to cover unknown workspaces and to learn respective inverse models. The particular focus is on the scalability of the approach in up to 50 dimensions, and on a comparison to purely random discovery strategies.

A. Experimental Setup

The basic experimental setup is illustrated in Fig. 3: The experiments initially concern a planar robot arm with \(m = 5\) degrees of freedom and a total length of 1m. The first joint can move in a range \([-0.2\pi; 0.2\pi]\). All other joint are limited to “elbow(s)-down” positions in a range \([0.0; 0.4\pi]\), which allows the arm to bend exactly to the extent to touch its own base.

The experimental advantage of this setup is that the ground truth ranges (see blue lines in Fig. 3) of the workspace can be analytically described. For instance the lower right boundary describes a movement that starts from a stretched downward posture (lowest point) and then fully bends one joint after the

\[ q^{\text{home}} \]

where the movement is exactly in the center of the available joint space, i.e. with components \(q^{\text{home}} = (q^{\text{min},i} + q^{\text{max},i})/2\) (see blue posture in Fig. 3). Starting from that point the goal babbling algorithm is applied with the direction sampling of goals. The step-widths are chosen as \(\epsilon_X = 0.02\), which allows to cross the entire in workspace in approx. 100 steps, and \(\epsilon_Q = ||q^{\text{max}} - q^{\text{min}}||/100\) such that the entire \(m\)-dimensional joint space can likewise be crossed in 100 steps. Further parameters of the goal babbling implementation are all identical to [8].

The main experimental concern is whether learning and other. The algorithm is not aware of these ranges, but they can be utilized for experimental measures.

In the experiments the arm is initialized in a home posture \(q^{\text{home}}\) exactly in the center of the available joint space, i.e. with components \(q^{\text{home}} = (q^{\text{min},i} + q^{\text{max},i})/2\) (see blue posture in Fig. 3). Starting from that point the goal babbling algorithm is applied with the direction sampling of goals. The step-widths are chosen as \(\epsilon_X = 0.02\), which allows to cross the entire in workspace in approx. 100 steps, and \(\epsilon_Q = ||q^{\text{max}} - q^{\text{min}}||/100\) such that the entire \(m\)-dimensional joint space can likewise be crossed in 100 steps. Further parameters of the goal babbling implementation are all identical to [8].

The main experimental concern is whether learning and
exploration can cover the entire workspace $F$. Since a discrete number of generated examples can not fill the volume of that workspace, it is assumed that examples cover their locally surrounding volume up to a distance $\epsilon$. The workspace coverage $C$ then measures which portion of the workspace is already covered by the example observations $x_i$:

$$ C_t^{F} = \frac{1}{V} \int c(x, \epsilon, t) dx$$

where $V = \int_F 1dx$ is the volume of $F$. The evaluation of this measure is computationally very costly, since any collected example must be compared to any point in the continuous volume of $F$. Since a repeated evaluation is desirable in order to inspect the exploration progress, an approximation is made: only those examples are explicitly memorized that have a distance of at least $\epsilon = 0.1m$ to other memorized examples. This set of prototypes is visualized in Fig. 3 as gray topological structure [23]. Then, these prototypes are compared to positions in $F$ that are chosen from a regular grid with a distance of 0.04m between different vertices. This measure allows to compare the discovery performance of the goal babbling approach to a purely random exploration (motor babbling). For this experimental baseline motor babbling is performed by choosing random actions from $Q$, and then interpolating continuous movements paths between them with the same step-length $\epsilon_Q$ used during goal babbling.

A secondary measure concerns the learning of the coordination skill itself by means of goal babbling. The performance error expresses how accurately the inverse estimate $g$ can reach for any goal in the workspace:

$$ E_t^{X} = \frac{1}{V} \int_F \|x^* - f(g(x^*, \theta_t))\| dx^*.$$  (8)

Practically this measure is evaluated on the regular grid of goals that is also used for $C_t^{F}$. Statistics in this section are all computed across 100 independent trials.

B. Discovery and Coordination

An exemplary performance of the approach is shown in Fig. 3: In the very first timestep only the home posture $q^{home}$ is known and reachable. Initially the direction sampling only generates very short paths because desired movements can not be accurately realized. Yet, the exploration of random perturbations quickly unfolds the inverse estimate along sensitive movement directions which allows to discover new positions in the workspace. After $t = 10^4$ examples, which corresponds to 100 crossings of the workspace, most reachable areas in $F$ have been discovered. The direction sampling can now generate straight paths through almost the entire space. Sequential steps discover the entire volume. At the same time an accurate inverse model is learned: while the gray topological structure indicates positions $x$ that have been discovered, the black structure indicates how well the inverse model can reproduce the finding when trying to reach for that position. Both structures are in almost perfect congruence, which indicates that the agent has not only incidentally discovered these positions, but is also able to reach for them.

Fig. 5. Workspace discovery in $m = 50$ dimensions with random motor babbling after $t = 10^9$ examples. Even after a billion samples, random motor babbling only discovers half of the available workspace.

Statistics across different trials are shown in Fig. 4. The workspace coverage $C$ over the time is shown in Fig. 4(a), where the quantiles 50% (bold line), 25% and 75% (filled area), 10% and 90% (thin lines) are visualized. The plot shows that the approach reliably covers the workspace in all trials. However, also random motor babbling performs well. Random motor babbling has a slight advantage in the beginning since it directly exploits the full joint ranges, while the direction sampling of goals often stops very early due to the initial lack of coordination. In the end goal babbling has an advantage for the coverage measure, but both methods succeed. Fig. 4(c) shows the development of the performance error which shows that all positions can be accurately reached after a while. This measure is not applicable for motor babbling, which does not subsume a coordination skill. Random examples can not be interpolated [6] in order to reach for some intermediate goal. Goal babbling, in contrast, directly subsumes an inverse model that can perform such interpolation.

C. Harder Problems with more Dimensions

The experiment with $m = 5$ degrees of freedom demonstrates the general feasibility and success of the approach introduced in this paper. Goal babbling can be performed without the manual prior specification of a set of goals. Yet, the setup is rather low-dimensional which even permits successful discovery by means of an exhaustive random exploration. The feasibility of random motor babbling declines rapidly when the problem comprises more degrees of freedom. An example is shown in Fig. 5. In $m = 50$ dimensions, the random exploration has only discovered half of the workspace even after many millions of movements (one billion examples), while in five dimensions several thousand random movements are sufficient to discover the workspace. The random exploration essentially explores the same outcomes $x$ again and again, even though with different actions $q$. This phenomenon can be qualitatively understood with the statistical law of large numbers: if the effect of many small random variables (here random angles for different joints) is added and normalized, then the outcome
shrinls to the average with decreasing variance. In Fig. 5 this
effect is well visible: the area in the middle is very often
visited, as indicated by the yellow color (“hot”). The random
choices for different joints largely compensate each others
effects and often move the effector into this region. Even those
areas that are explored at all are mostly “cold” (blue color)
and many of them are visited only once.

The same setup with goal babbling and direction sampling
is shown in Fig. 6. Goal babbling quickly learns efficient
ways of coordination. In contrast to random movements,
different joints are actuated cooperatively along such efficient
solutions which allows for a quick discovery of new outcomes.
Without knowing the ground-truth ranges of the workspace,
the direction sampling of goals is able to cover the entire
workspace after a while, and provides the mean to learn an
inverse model for coordination (see Fig. 4(c)). Statistics of
the workspace coverage during exploration and learning are
shown in Fig. 4(b). The goal babbling approach approximately
covers the workspace after 10^6 examples, which corresponds
to 10,000 continuous crossing of the workspace. After the
same time, random motor babbling has only discovered 40% of
the workspace, and only very slowly discovers new areas.
Of course, motor babbling will succeed in the limit case,
but practically the problem is desperately intractable with
this approach. If the behavior in the plot is, speculatively,
extrapolated, one can estimate that motor babbling succeeds
earliest after 10^14 examples.

A systematic assessment of the algorithm’s scalability is
shown in Fig. 7. The dimensionality of the problem is varied
between m = 2 and m = 50 degrees of freedom. The joint
ranges are consistently [-0.2π; 0.2π] for the first joint. The
ranges for all other joints are adapted to [0; 2π/m], which
generalizes the setup for m = 5 such that the workspace F
has a coherent size and shape (compare Fig. 3 and 6). Home
positions and step-widths are set to corresponding values
q_{i}^{home} = (q_{i}^{min} + q_{i}^{max})/2 and \epsilon_{Q} = ||q_{i}^{max} - q_{i}^{min}||/100. After
an experiment runtime of t = 10^5, goal babbling with direction
sampling of goals consistently covers the workspace (see Fig.
7(a)) throughout the entire range of DOFs. In contrast, the
performance of random motor babbling drops substantially
around m = 10. This drop of performance is even visible
when random exploration is run a hundred times longer
(t = 10^6). In high dimensions, the goal babbling approach
outperforms random exploration by many orders of magnitude,
even without pre-specifying a set of goals. Fig. 7(b) shows
the number of examples needed until the performance error
drops to 10% of its initial value. It shows a mild increase
of cost, in which the discovery in m = 50 dimensions is
approx. 40 times as expensive as in m = 2 dimensions, which
is excellent compared to the exponential cost of exhaustive
random exploration. This measure was already used in [8]
in a setup with pre-specified goals. The setup could even
demonstrate an approximately constant curve across two to
50 DOFs. Yet, this setup did only involve a subset of all
achievable goals. This study investigates an entire workspace,
including very difficult to represent and discover boundary
areas. In particular the lower regions of the workspaces are
increasingly difficult in high dimensions: following the lower
boundary requires to fully move one joint after the other.
Representing a coordination skill for such movements requires
to represent very strong curvatures when many joints are
involved. Since a representation of such movements is difficult,
also their discovery is difficult because finding some outcome
always requires to know “how” to achieve it. Yet, the increase
of cost is only very mild for goal babbling, and even in
50 dimensions useful results can be obtained in a practically
feasible time.

V. DISCUSSION

This paper has introduced the first formulation of goal
babbling that can learn coordination skills without any prior
specification of a set of goals. It thereby improves the plausi-
bility of the approach by reducing pre-structuring, and prac-
tically supersedes a design step that can be very challenging
and tedious for instance in scenarios with biomorphic robots
[11],[17]. Instead, the approach incrementally explores goals
along randomly chosen directions, which it follows as long
as an angle of less than 90° between desired and actual
movement can be realized. Initially this results in a rather
local exploration around the starting point due to an initial
lack of skill. Due to the rapid learning of an inverse model,
however, the learner can quickly generate further movements,
and eventually covers the entire workspace without having any
representation of it. This allows both for an identification of the
workspace itself (“what” is possible), as well as a successful
coordination therein (“how” it is possible).

The scalability results are particularly encouraging: while
the mere workspace discovery with random motor babbling
drastically fails in the high-dimensional setups, goal babbling provides useful results in a practically feasible time. Assuming that continuous movements are sampled on a real robot with 10 examples per second, useful results emerge after few hours of real time (e.g. $t = 10^7$ corresponds to less than three hours). An exhaustive search, implemented by random motor babbling, was shown to be significantly less successful in high dimensions even if a billion examples is generated, which would correspond to 3 years of real world exploration. The extrapolation that random motor babbling might succeed to discover the workspace after $t = 10^{14}$ would correspond to millions of years.

It is worth to highlight the simplicity and generality of the direction sampling approach: both the formulation of paths along randomly chosen directions as well as their stopping criterion are very easy to implement. While we can only speculate about the underlying mechanisms in biological learning processes for such achievements, it is clearly plausible that nervous systems are capable of such simple calculations. At the same time it is general enough to be compatible with other implementations of goal babbling, for instance for the learning of forward models [9] or associative memories [3] for coordination.

### References