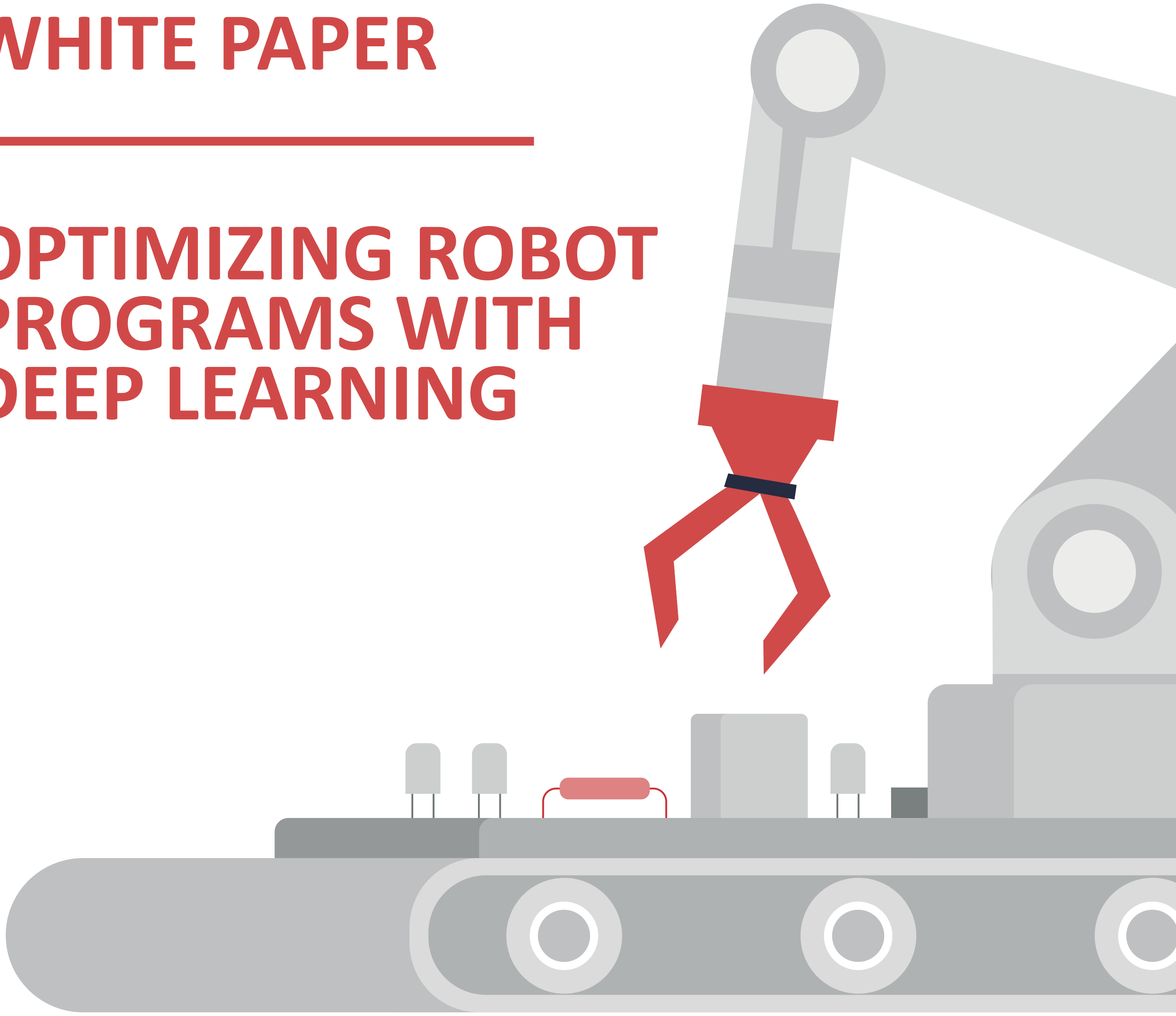
ARTIMINDS

# WHITE PAPER

# OPTIMIZING ROBOT PROGRAMS WITH DEEP LEARNING



# INTRODUCTION

Due to their flexibility, modern industrial robots have become core technology components of state-of-the-art production facilities in nearly every industry. Their large number of degrees of freedom and the availability of intuitive software tools for programming, integrating and monitoring them have made them particularly effective for high-mix, low-volume production as well as sensor-adaptive applications. Both contexts are challenging to address with traditional automation technology: Achieving the required degree of flexibility would require the expensive development of custom hardware. This flexibility is offered at much lower cost by industrial robots and their surrounding sensor and software ecosystem.

While industrial robots allow to commission, program, deploy, monitor and adapt flexible production cells much more quickly and economically than traditional automation, those savings are largely due to reduced requirements of custom hardware and high potential for standardization. Their inherent flexibility means that parts of the costs are now shifted to the programming and deployment stages of the robot cell lifecycle – where adaptive force control or vision-based robot skills are now realized in software. The ArtiMinds RPS allows robot programmers to quickly program robots to solve complex tasks by combining pre-parameterized robot skills, with dedicated skills for force- or vision-controlled sensor-adaptive tasks. Skill-based robot programming, particularly in conjunction with the data visualization and analysis tools provided by ArtiMinds LAR, significantly lowers the remaining overhead of programming sensor-adaptive tasks, or of adapting existing robot programs for high-mix, low-volume manufacturing.

# PARAMETER OPTIMIZATION IS A CORE BOTTLENECK IN ADVANCED ROBOT APPLICATIONS

With skill-based robot programming reducing the overhead of programming robots, the central remaining cost factor of advanced robotic automation is the adaptation of robot skill parameters during ramp-up and deployment. Advanced force-controlled robot skills such as search motions or moment-controlled insertion can be adapted via a set of parameters, which define central aspects of the underlying motion or force controllers.

Consider the task of placing sensitive THT electronics components onto a printed circuit board (PCB), where the positions of the connector holes and the length and orientation of the pins vary within certain bounds due to manufacturing tolerances.
A skill-based robot program to perform this task typically consists of a force-controlled approach skill to establish contact with the PCB, followed by a force-controlled spiral or spike search to find the precise position of the hole, as well as a force-controlled insertion motion to avoid damage to the components during insertion. The respective velocities, accelerations, PID controller parameters, but also skill-specific parameters such as the maximum contact force or the size and orientation of the spiral search motion can be parameterized.

**THE ROBOT PROGRAM PARAMETERIZATION BOTTLENECK**
• Force-controlled robot skills require careful parameter tuning for performance and reliability.
• Parameter tuning by human programmers is cost-intensive and requires expertise.

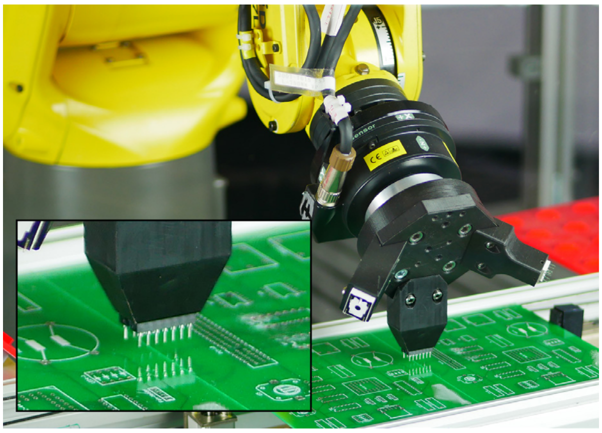
While the default parameterizations often solve the task, they should generally be fine-tuned to the particular application during ramp-up. This typically amounts to finding the parameterization which minimizes cycle time while respecting quality requirements - a challenging multicriterial optimization problem which requires significant robot and domain expertise and a possibly lengthy trial-and-error period during ramp-up, driving up costs and increasing time to production.

# AUTOMATIC PARAMETER OPTIMIZATION WITH SHADOW PROGRAM INVERSION

To further increase the cost-effectiveness of skill-based robot programming, we propose Shadow Program Inversion (SPI), a data-driven solution to automatically self-optimize robot skill parameters using machine learning. In recent years, machine learning-based approaches have been used with increasing success to increase the level of autonomy of robots in the industrial and service domains. Most notably, methods leveraging Reinforcement Learning (RL) or Learning from Demonstration (LfD) have shown promising results, teaching robots to perform complex tasks by exploring the space of possible actions in the environment (RL) or learning from human teachers (LfD). Both approaches are limited in their applicability for industrial robot programming, however:

Reinforcement Learning effectively automates the trial-and-error process by allowing the robot to repeatedly interact with the environment, which is not feasible at the ramp-up phase of a real-world production cell; RL in simulated environments requires highly accurate and compute-intensive simulators, and bridging the resulting gap between the simulation and reality is considered an unsolved problem.

The application of Learning from Demonstration to robot skill parameter optimization is challenging for many industrial applications, where the speed and precision of modern industrial robots often far surpasses the capabilities of humans.

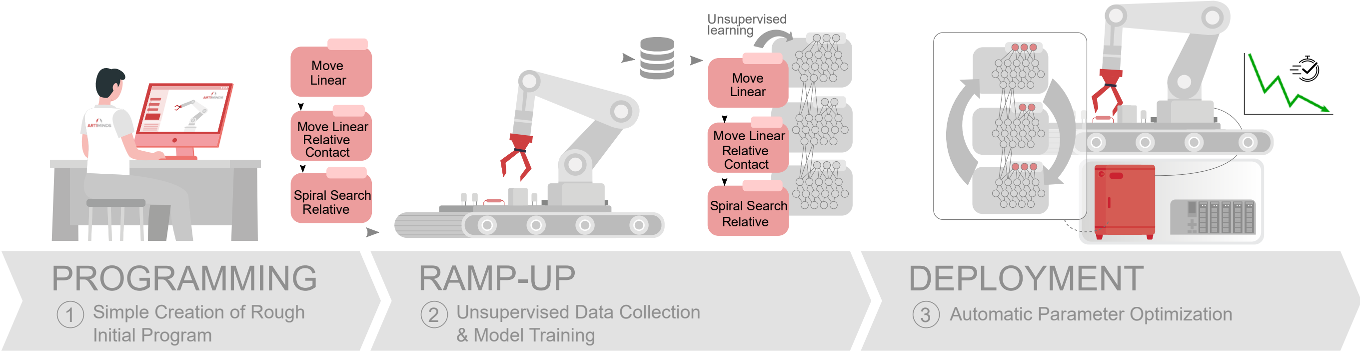


*SPI has been validated in several industrial applications such as THT assembly (left), but also in conjunction with VR and Learning from Demonstration in a household scenario (right).*

**SPI, BY CONTRAST, HAS BEEN DESIGNED TO SEAMLESSLY INTEGRATE INTO THE RAMP-UP AND DEPLOYMENT PROCESS OF ROBOTIC WORKCELLS.**

The core idea of SPI is to learn a model (called "Shadow Program") of the robot program, which predicts the expected robot trajectory (poses of the tool center point (TCP), as well as forces and torques) given the program's input parameters. We design this model to be differentiable: The model can provide an estimate of how small changes to the input parameters will affect the resulting robot trajectory. This property enables the optimization of the program's inputs via gradient descent, an efficient method for solving the type of multicriterial optimization problems encountered in the fine-tuning of robot skill parameters.

# SHADOW PROGRAM INVERSION: THE WORKFLOW




## PROGRAMMING

1. First, the robot programmer creates the robot program and provides a **ROUGH INITIAL PARAMETERIZATION**, typically based on CAD measurements and the default parameterizations provided by the RPS. For each parameter to be optimized – usually the parameters of particularly time-critical force-sensitive motions - she then defines the optimization domain, i.e. the range of permissible parameter values, to ensure safe operation of the workcell during the optimization process.

## RAMP-UP

2. SPI then **AUTONOMOUSLY COLLECTS DATA** by repeatedly executing the program on the robot, sampling a new set of program parameters at each iteration and storing the resulting measured TCP positions, forces and moments. Once enough data has been collected, SPI **TRAINS THE SHADOW PROGRAM** using this data.

## DEPLOYMENT

3. To obtain an optimal set of program parameters, the robot programmer must provide the objective function, which is to be optimized, usually a combination of cycle time minimization, some hard quality constraints reflecting the permissible tolerances and task- or workpiece-specific additional constraints, such as force or torque limits. SPI then **COMPUTES THE OPTIMAL PARAMETERIZATION** via gradient descent over the shadow program with respect to the objective function and updates the robot program with the optimized parameters.

# PROPERTIES AND ADVANTAGES

SPI HAS SEVERAL ADVANTAGES WHICH MAKE IT PARTICULARLY APPLICABLE TO REAL-WORLD ROBOT PROGRAMMING AND DEPLOYMENT WORKFLOWS:

1

**DECOUPLED LEARNING AND OPTIMIZATION.**
The core idea of SPI is to learn a model of the program, which then facilitates parameter optimization by being differentiable. This splits SPI into two distinct phases, a model learning phase and a parameter optimization phase. This split is advantageous for several reasons. First, it greatly improves the data efficiency of our approach, as the learning problem is considerably simplified, particularly compared to RL-based alternatives. Instead of having to learn what the program should do (i.e. learning an optimizer or an optimal policy in an RL manner), SPI only needs to learn what the program is doing (by simply observing repeated executions of the program with different parameters) - a much easier learning problem requiring much less data. The subsequent optimization step is pure computation – no learning required. Second, learning a model of the program instead of an optimizer can be done via unsupervised learning, requiring no human demonstrations or labelled training data. This enables the seamless integration of SPI into the ramp-up phase of the workcell lifecycle – data collection happens in the background while the workcell is undergoing regular pre-production testing.

2

**OPTIMIZATION OF ARBITRARY USER GOALS.**
SPI allows the robot programmer to specify arbitrary user goals, provided they can be expressed as a differentiable function of the robot's actions. This covers the most common objectives such as cycle time, path length or probability of failure, as well as most relevant process metrics or manufacturing constraints. Moreover, as a corollary of the split between learning and optimization, the user goals do not have to be known when training data is collected and the shadow model is trained, enabling the repeated optimization of parameters with respect to different, possibly changing objectives over the course of workcell deployment without requiring new data collection and training.

3

**JOINT OPTIMIZATION FOR ALL SKILLS IN THE PROGRAM.**
Many challenging real-world tasks require several force-controlled skills to be executed in sequence. SPI jointly optimizes the parameters of all skills in the program, automatically respecting interdependencies between skills: For instance, the target pose of an approach motion will be optimized so that the succeeding search motion has the highest probability of succeeding. Alternative approaches require complex heuristics or human intervention to achieve this.

4

**LIFELONG LEARNING AND RE-OPTIMIZATION.**
The capability of SPI to optimize program parameters is not limited to the initial programming and deployment of the workcell. It also allows for the re-optimization of program parameters in the future, by simply collecting additional data during regular production, re-training the shadow model and recomputing the optimal parameterization. This is particularly relevant for high-mix, low-volume production, as the robot program, work pieces and possibly the workcell layout will change frequently. SPI has the potential of greatly reducing the overhead of manual reparameterization after every change, making small-batch, and flexible production much more economical.

# CONCLUSION

In summary, SPI combines machine learning and gradient-based optimization to automate the optimization of robot parameters in a way that is particularly tailored to the pre-production stage of workcell deployment. We have evaluated SPI on real-world electronics and mechanical assembly tasks involving complex sequences of search and insertion strategies. We found that while RL-based alternatives or gradient-free optimization methods such as evolutionary algorithms were not applicable or ineffective in real robot cells, SPI effectively automated parameter optimization with resulting parameterizations which matched or outperformed human experts.

In addition, we demonstrated the flexibility of SPI by combining it with human demonstrations in a Virtual Reality (VR) environment and applied it in a household setting to move a glass into a sink. We could show that one single demonstration was enough to not only optimize parameters, but to generate a good parameterization from scratch, demonstrating the utility of SPI beyond industrial robot programming.

We presented SPI at the 2021 IEEE International Conference on Robotics and Automation (ICRA), and patented the technology with the German patent office. Current work is focused on reducing the data requirements of SPI even further, and on transferring SPI from the lab into real-world production settings. If you are interested in the technology and seek an automated solution to optimize your robot programs, contact us at contact@artiminds.com.


# SUMMARY

Shadow Program Inversion allows to automatically finding optimal program parameters by

1. Learning a model of the robot program using unsupervised learning
2. Using the learned model to compute optimal program parameters

SPI is applicable to any robot program and optimality criteria, is highly data-efficient, enables lifelong learning and has been tailored to integrate into the standard robot deployment workflow.



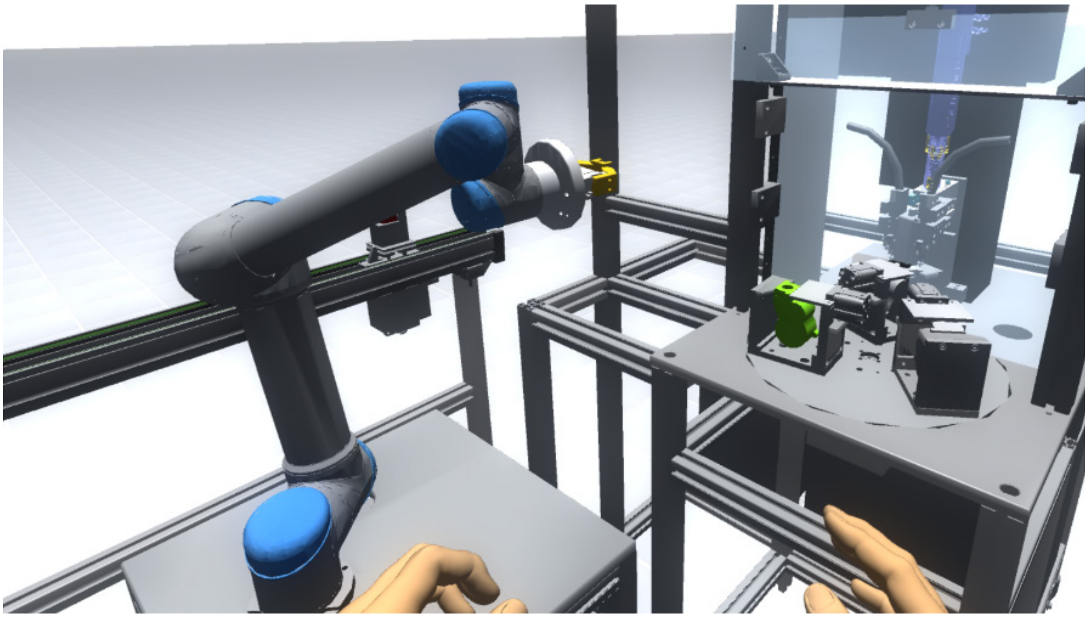
Author:
Benjamin Alt
Research Scientist
ArtiMinds Robotics GmbH

# ARTIMINDS INNOVATION LAB

ArtiMinds Robotics GmbH, with around 40 employees, is the technology leader in the field of intuitive robot programming. The in-house developed software solutions ArtiMinds Robot Programming Suite (RPS) and ArtiMinds Learning & Analytics for Robots (LAR) are used in over 20 countries and more than a dozen different industries.

The ArtiMinds Innovation Lab conducts predevelopment and applied research in advanced robotic manipulation, machine learning and robot cognition. By creating and exploring novel technologies and paradigms for interacting with, programming and deploying robots, we lay the foundations for new products and services in application domains ranging from industrial robotics over service robotics to health care.

ArtiMinds is deeply integrated in a network of industrial and industry-related academic partners, consisting of Fraunhofer institutes, universities and universities of applied sciences. In addition, ArtiMinds is a project partner of the Digital Hub Applied Artificial Intelligence Karlsruhe and a member of the innovation advisory board of the Competence Center for AI Engineering (CC-King).



# CONTACT US

ArtiMinds Robotics GmbH
Albert-Nestler-Str. 11
76131 Karlsruhe
**Phone** +49 721 96694781
**Fax** +49 721 96694708
**Email** contact@artiminds.com
**Web** www.artiminds.com
https://innolab.artiminds.com/