

Received July 31, 2020, accepted August 24, 2020, date of publication September 4, 2020, date of current version September 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3021950

# Bacterial Memetic Algorithm Trained Fuzzy System-Based Model of Single Weld Bead Geometry

CSONGOR MÁRK HORVÁTH<sup>1,2</sup>, JÁNOS BOTZHEIM<sup>1</sup>, (Senior Member, IEEE),  
TRYGVE THOMESSEN<sup>2</sup>, AND PÉTER KORONDI<sup>1</sup>, (Senior Member, IEEE)

<sup>1</sup>Department of Mechatronics, Optics and Mechanical Engineering Informatics, Faculty of Mechanical Engineering, Budapest University of Technology and Economics, 1111 Budapest, Hungary

<sup>2</sup>PPM Robotics AS, 7038 Trondheim, Norway

Corresponding author: Csongor Márk Horváth (hcsongorm@mogi.bme.hu)

This work was supported in part by PPM Robotics AS, in part by The Norwegian Research Council through the Virtual Presence in Remote Operation of Industrial Robot Systems and the Industrial Ph.D. Program under Grant 244972/O30, in part by the CoRoWeld Project through the User-Driven Research-Based Innovation Program under Grant 245691, in part by the MultiPass2020 Project through the SkatteFUNN Program under Grant 293141, in part by the NKFIH Hungary under Grant BME NC TKP2020, and in part by the BME Artificial Intelligence of NKFIH Hungary under Grant TKP2020 IE and Grant BME IE-MI-FM TKP2020. The work of János Botzheim was supported by the János Bolyai Research Scholarship of the Hungarian Academy of Science.

**ABSTRACT** This article presents a fuzzy system-based modeling approach to estimate the weld bead geometry (WBG) from the welding process variables (WPVs) and to achieve a specific weld bead shape. The bacterial memetic algorithm (BMA) is applied to solve these problems in two different roles, as a supervised trainer, and as an optimizer. As a supervised trainer, the BMA is applied to tune two different WBG models. The bead geometry properties (BGP) model follows a traditional approach providing the WBG properties as outputs. The direct profile measurement (DPM) model describes the bead profiles points by a non-linear function realized in the form of fuzzy rules. As an optimizer, the BMA utilizes the developed fuzzy systems to find the solution sets of WPVs to acquire the desired WBG. The best performance is achieved by applying six rules in the BGP model and eleven rules in the DPM model. The results indicate that the normalized root means square error for the validation data set lies in the range of 0.40 – 1.56% for the BGP model and 4.49 – 7.52% for the DPM model. The comparative analysis suggests that the BGP model estimates the BWG in a superior manner when several WPVs are altered. The developed fuzzy systems provide a tool for interpreting the effects of the WPVs. The developed optimizer provides multiple valid set of WPVs to produce the desired WBG, thus supporting the selection of those process variables in applications.

**INDEX TERMS** Bacterial memetic algorithm, fuzzy system, machine learning, TIG welding, weld bead geometry.

## I. INTRODUCTION

Most of the automated welding systems were developed and used for mass production. Recently, the robotization of small series production gained more attention, primarily by small and medium-sized companies. The transition from manual to robotized welding requires the adaptation of the welder's expertise to the automated system. However, this knowledge is mostly not available in a quantified format as precisely as needed to program the robotic welding system or estimate the weld bead geometry (WBG).

The associate editor coordinating the review of this manuscript and approving it for publication was Min Xia<sup>1</sup>.

The Tungsten Inert Gas (TIG) welding method can produce solid and high-quality joints for a wide range of regular and more exotic types of metals [1]. Along with the welding process's non-linear characteristics, the weld bead formation depends on the chemical compositions of the base and filler materials [2], [3]. The sulfur and oxygen content could vary in a wide range between the manufacturing batches, which influence the surface tension of the melted metal, an important factor defining the shape of the solidified surface [4], [5]. Furthermore, the deposition efficiency needs to be considered besides the inconsistent evaporation of the filler metal [6].

Consequently, a model-based approach is needed to achieve the requirements for the given material composition

and environment; and support the activities during the design and the execution of the welding process.

During the early 2000's, optimization and modeling of welding processes mostly consisted of numerical [7] and statistical approaches [8]. However, in the last decade computational intelligence and machine learning became dominant [9], [10] due to the capability to solve complex and non-linear problems. Now they are providing a base for applications for future intelligent welding manufacturing [11]. Recently, WBG models and the related process controls gained attention in wire and arc additive manufacturing (WAAM) [12]–[15] and multi-pass welding (MPW) [16]–[18].

Most research studies of WBG modeling are focused on consumable electrode welding methods [12], [19]–[23] and flux activated TIG (A-TIG) without deposition [24]–[26]. Only a few study focus on TIG for stainless steel bead geometry [27]–[30], because TIG was traditionally used without deposition or on non-steel metals [31] to weld joints where the reinforcement is not utilized.

Numerical models provide deep analysis on the mechanical properties, residual stresses, distortion, and heat distribution beside other internal properties of the welded workpiece [7], [32]–[34]. However, the bead shapes and the welding process variables (WPVs) are not considered, since they are provided as inputs to the finite element modeling and adjusted manually. The WPVs are adjusted until they provide the final product's required properties, while the weld bead shapes are modeled by their simplified shape. The modeling neglects the challenges connected to the precise execution by including higher error margin and safety factors. The shape of the weld beads is an important factor in WAAM and MPW because controlled height and width are needed to ensure stable deposition [35]–[37]. Analytical models were developed to describe the bead shape [36], [38], [39], but those models were discussing the type of the fitting curve for the bead surface and its effect during overlapping.

Statistical methods are often used in the literature to provide the comparison base for different methods since their linear behavior is usually outperformed by methods capable of modeling the non-linear behavior [12], [21]. Computational intelligence (CI) techniques – such as artificial neural networks (ANN) [12], [22], [23], fuzzy inference systems (FS) [20], [24], [25], evolutionary algorithms (EA) [40], [41], and genetic programming (GP) [42] – are widely used to describe the WBG. However, due to their limitations [26], several hybrid computing techniques were developed [26], [32], including the adaptive neuro-fuzzy inference systems (ANFIS) [25], [26], [43] and the evolutionary fuzzy systems (EFS) [44], [45].

An important common feature of computational intelligence techniques is that they aim at acceptably suboptimal, usually, approximate solutions, while keeping the computational complexity at a tractable, usually low degree polynomial level. CI methods such as fuzzy systems and neural

networks are universal approximators [46], [47]. They can be transformed into each other, therefore any approach (or architecture) choice can be justified [48]–[50]. Kóczy showed that the Takagi-Sugeno-Kang (TSK) FS is asymptotically equivalent to the Mamdani FS model, and they can be transformed into each other [49].

The main advantage of choosing FS is the inference base, compared to the black-box behavior of the ANNs. Fuzzy systems provide human-like reasoning systems since their rule-based approach with a non-linear mapping of inputs offers an easily interpretable method, where the arguments leading to the conclusion can be assessed.

The FS method's main drawback is to find the optimal definition of membership functions and define the sufficient number of rules. The fewer number of fuzzy rules may help understand the system's decision-making, bringing the developed inference model closer to the person applying it in operation. Furthermore, the fuzzy inputs allow us to handle some degree of uncertainty in the input parameters, which happens in welding. In the overviewed literature, either high number (above 20 rules) [25], [26], [51] or manually defined fuzzy rules [24], [52] were applied to estimate one feature of the weld bead.

The tuning of the membership function parameters can be done by applying training algorithms [53], which allow the fuzzy systems to learn from data. In many different fields, the evolutionary algorithms are applied to design the fuzzy systems [44], [45], [54].

Several evolutionary optimization algorithms were developed, with the ability to solve and quasi-optimize problems with non-linear and discontinuous characteristics [55], [56]. The bacterial evolutionary algorithm (BEA) [57] is one of the possibilities that mimic bacterial rather than eukaryotic evolution among microbes. Each bacterium represents a solution to the original problem. In their mutation and the gene transfer operations, bacteria share chunks of their genes instead of performing neat crossover in chromosomes, which is the characteristic of the eukaryotes. The main disadvantage of the classical evolutionary algorithms is the low convergence speed, thus long running time. However, combining them with gradient-based local search methods can utilize both methods' advantages in the optimization process. This hybridization leads to memetic algorithms [58].

Bacterial Memetic Algorithm (BMA) [54] is a memetic algorithm, in which the bacterial technique is used instead of the classical genetic algorithm, and the Levenberg-Marquardt (LM) method [59], [60] is applied as local search. The BMA provides a competitive performance during optimization [61], [62] and supervised machine learning [62] against genetic algorithm (GA) and particle swarm optimization (PSO) and their memetic versions. It has already been applied in several combinatorial optimization problems [61], [63], in continuous optimization tasks [64], and in supervised machine learning tasks such as fuzzy rule base

extraction [54] and training fuzzy neural networks [65]. However, the capabilities of BMA were not explored in welding or related technology yet.

## II. PROBLEM DEFINITION

Precise positioning of weld beads is required to achieve a desirable weld joint in welding, which is influenced by the weld bead shape during the deposition of multiple beads. The weld bead geometry is directly related to the WPVs, and the bead formation depends on the previously deposited beads' shapes.

In small series productions, the tuning of the process could take up a significant amount of time and be done by trial and error method following internal standards; and carrying out experiments and laboratory analysis of the test workpieces. This approach can be enhanced by a bi-directional model to describe the relationship between the process parameters and the resulting WBG to support the automated operation and the selection of the process variables.

Simulation and analytical models could also support the manufacturing process, where the WPVs and WBG are provided as inputs to the finite element modeling. The target WBG is modeled by simplified shapes and defined by manual calculations or numerical calculations to satisfy the criteria of the mechanical loads and stresses of the workpiece. The WBG and WPVs are adjusted until they provide the required properties of the final product. Mishra and DebRoy [32] showed that a specific WBG could be produced by multiple combinations of the WPVs, even in the range qualified by the *Welding Procedure Specification*. A provided list of WPVs could allow to select the best combination fitting to the desired workpiece properties.

However, numerical modeling neglects the challenges connected to the precise execution by including higher error margin and safety factors. Proposing a model to estimate the bead geometry provides refined geometrical information to further evaluation of the mechanical properties and residual stresses in the workpiece. Therefore, such a model does not compete with the numerical models but complements them to define the WPVs and WBG.

As the industrial application aspect of the welding process is considered, three control parameters were selected in our approach, namely arc current ( $I$ ), torch travel speed ( $v_t$ ), and wire feed rate ( $v_f$ ). Arc voltage ( $U$ ) was also recorded and included in the model as input since it had small fluctuation and a significant effect on the heat input. These four parameters will be referred to as Welding Process Variables (WPVs). The examined parameters of the WBG were chosen as width ( $w$ ), height ( $h$ ), and cross-sectional area ( $A_B$ ) and later referred to as Weld Bead Profile Properties (BPPs).

To describe the weld bead profiles, the second-order polynomial (parabola) fitting was chosen [36]. The parabola curves are reconstructed from the  $w$ ,  $h$  parameters using the following equation:

$$y = -4 \frac{h}{w^2} x^2 + h, \quad (1)$$

where  $x$  is the abscissa- and  $y$  is the ordinate of the profile points of the weld bead in the coordinate system in the cross section, according to the definition shown in Fig. 2. The parabola defined in Equation (1) does not contain the first order  $x$  component since the coefficient is zero because the origin of the coordinate system is in the focal point of the parabola.

The theoretical  $A_B^d$  cross-sectional area of the weld beads can be estimated according to Equation (2) by calculating the amount of the deposited weld metal:

$$A_B^d = \eta_d \frac{\pi D_w^2}{4} \cdot \frac{v_f}{60v_t}, \quad (2)$$

where  $\eta_d$  represents the deposition efficiency, and  $D_w$  is the diameter of the feeding wire. The deposition efficiency is estimated by comparing  $A_B^d$  calculated value with the measured  $A_B^m$  value using the area under the curve on the measured profile (Eq. (3)).

$$A_B^m = \int_{-w/2}^{w/2} f(w, h) dx = \frac{2wh}{3} \quad (3)$$

During our preliminary study, we found that the value of  $\eta_d$  is 0.955 with a standard deviation of 0.063 for the whole sample population, which is similar to a generally accepted (above 0.9) value [6]. Furthermore, the deposition efficiency decreases with the increase of arc current, which is interpreted as the result of a higher vaporization rate of the feed metal. However, the sign and the degree of the deviations are not consistent, causing error propagation in multi-pass welding as the layered beads added on top of each other. On a given workpiece, the error is usually reduced by on-site adjustment of the torch position, and the welding parameters are set to as constant across several layers. Thus, we expect that an accurate model utilizing the measurement data can increase the accuracy of estimation of the WBG.

Furthermore, the review conducted by Vasudevan concluded that FSs are the best soft computing methods for control and monitoring of welding processes, and (GA) evolutionary algorithm-based models for WPV optimization [27]. This provided a motivation to utilize our method since it combines the advantages of both approaches.

The effectiveness of the BMA is emphasized by adapting it to our application in two different ways – developing a bead shape model and finding multiple optimal WPVs to achieve the target bead geometry. This method is unique since the BMA was not utilized in welding before or implemented for one problem with two roles (supervised trainer and optimizer). Furthermore, for the two separate tasks, two different methods are usually applied in the overviewed literature.

The specific aims of our study are to (i) provide a sufficient estimation model of the WBG by adopting the BMA to the welding application, where the FS-based model consists of a low number of fuzzy rules; and (ii) provide a decision support tool for selecting WPVs to achieve specific WBGs.

### III. PROPOSED METHODS

Our proposed method provides a bead geometry model on single weld beads in a horizontal position using fuzzy systems, and an optimizer to retrieve a list of WPVs producing a specified WBG. These support the decision to chose a suitable set of WPV for simulation or execution. The developed FS provides a human-like reasoning system and a non-linear mapping of inputs to assess the arguments leading to the conclusion. The supervised training of the rule base and the optimization task are carried out by the same Bacterial Memetic Algorithm.

Our proposed method is capable of accepting not only the conventional main welding parameters but also the profile points of the weld beads directly, measured by a laser triangulation sensor allowing a broader range of applications than the traditional methods.

In the welding process modeling, BMA is used both as a supervised trainer and as an optimizer. Our method (Fig. 1) can be broken down into the following four steps:

- 1) **Welding data generation** by TIG welding on 304L stainless steel
- 2) **Welding data acquisition and preprocessing** by the data processing framework
- 3) **Welding model generation** by tuning the membership functions of the fuzzy systems by BMA in order to infer the bead profile properties from the welding process variables
- 4) **Welding design** by BMA optimization of WPV in order to achieve the desired BPP resulting in a multiple set of WPV

The empirical model development is based on the welding experiments (Step 1), whose design is covered in the Sec. III-A, and utilize the second-order polynomial (parabola) fitting. The generated data is preprocessed (Step 2) to provide the profile property information of weld bead cross sections for each trial, following the method described in Sec. III-B. The output of the preprocessing is the training patterns for the model development of the weld bead profile.

During the supervised training (Step 3, Sec. III-C), the BMA is utilized to tune the membership functions of the fuzzy systems. The bacterium's chromosome encodes the breakpoints of the trapezoidal membership functions, and the evaluation of the bacteria is carried out by the Mamdani inference model [66]. The bacterium's chromosome is modified to reduce the value of approximation error ( $E_i$  in Fig. 1), which is the difference between the estimated and the desired BPP value in the  $i$ -th training pattern. The correlation between the WPVs and the BPPs is realized in two separate models.

The first approach, the Bead Geometry Properties (BGP) model (Sec. III-D), is a more conventional method to model a direct relation between the WPVs and the BPPs in three parallel fuzzy rule bases. The second approach is the Direct Profile Measurements (DPM) model (Sec. III-E), which is the extension of the BGP model. It contains a single fuzzy rule base to reconstruct the weld bead shape from profile points;

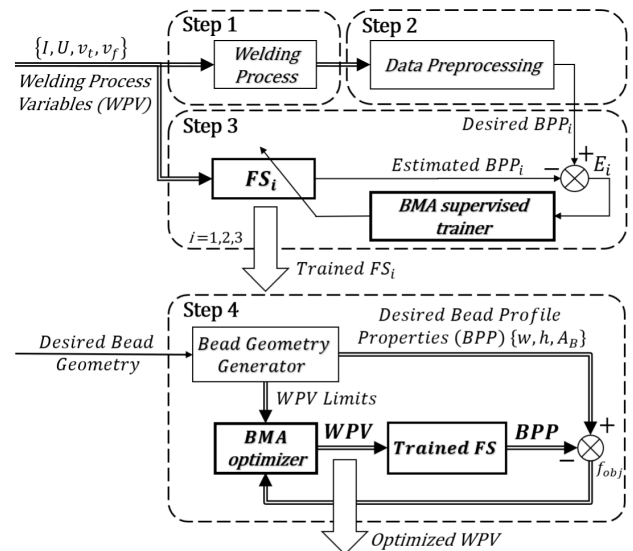


FIGURE 1. Overview of the proposed method's structure.

then, the BPPs are acquired as the result of the postprocessing.

As Step 4, the BMA is used again, now as an optimizer, incorporating the BGP model to optimize the WPVs to achieve the targeted weld bead geometry (Sec. III-F). Both the fitness of the models and the results of the optimization were validated through experiments.

The application of BMA in Step 3 and Step 4 differs in a few details, such as the encoded information in the bacteria's genes, the evaluation method, and the update process in the Levenberg-Marquardt algorithm. In the optimizer role, the bacterium encodes the WPVs, and the evaluation is based on the value of the single objective function. According to their gradient vector, the WPVs are modified with their direct effect on the objective function in the iterations between the generations. In comparison, in the supervised trainer role of the BMA, the bacteria's chromosomes encode a complete set of fuzzy rules. The parameters of the rule's membership functions are tuned in the process. During the evaluation of a bacterium, the estimated and the desired response of the fuzzy system is calculated as provided in the training patterns, giving the algorithm a multi-objective characteristics which is transformed into a single-objective optimization with uniform weights. Therefore, the gradient is represented as a Jacobi matrix instead of a gradient vector. The detailed discussion will be given of the implemented Levenberg-Marquardt algorithm in both cases in the corresponding sections.

#### A. EXPERIMENT DESIGN AND EXECUTION

A bead-on-plate welding experiment was carried out to create an estimation model of the weld bead profile (Step 1). Figure 2 shows a measured cross-section of a weld bead, incorporated visualization of the BPPs, and the expected results of the proposed models.

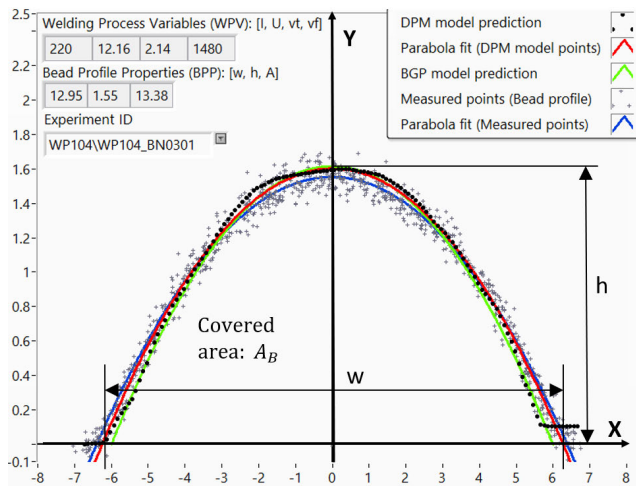


FIGURE 2. A weld bead cross-section as the result of the measurements and the models.

The base metal is 20 mm thick 304L steel with 3 mm clad welded with the same 1.2 mm diameter Böhler 13/4-IG filler material as used to produce the weld bead. The experiment was executed with a regulated constant 12 V, 2.4 mm arc gap in PA welding position, 40°C preheating temperature. An E3 tungsten electrode with rare earth mixed oxides of 3.2 mm diameter and 60° cone angle was used. Pure Argon gas with a flow rate of 12 – 14 L/min provided the shielding. The experiment was carried out in the welding cell shown in Figure 3 by a Kuka KR-30-3 and a NACHI MZ-07 industrial robot arms, using a MagicWave 5000 JOB welding power source, and a custom controller software.

TABLE 1. Definition of input process parameters and their levels.

	Level 1	Level 2	Level 3	Level 4	Level 5
Arc Current [A]	180	200	220	240	260
Torch Travel Speed [mm/s]	1.92	2.14	2.36	2.58	2.8
Wire Feed Rate [mm/min]	820	1040	1260	1480	1700

Taguchi design [67] was applied to design the experiment using an  $L_{25}(5^3)$  layout as shown in Table 1. Each trial was carried out twice, and each produced a 100 mm long weld bead. The WPVs' minimum and maximum values of the parameters were selected to comply with the *Welding Process Specification*, provided by our industrial partner. Unfortunately, not all samples provided usable data due to welding failures. Our design considered non-optimal settings; thus, high heat input was applied in the failed tests with a low feed rate and vice versa; consequently, these settings are not suitable for industrial applications. At the end of the modeling process, welding data from the  $N_{trial} = 42$  trials were included for the training, which was the result of 21 different WPV combinations (Training data sets in Table 2). A similar experiment was carried out to prepare the weld beads for

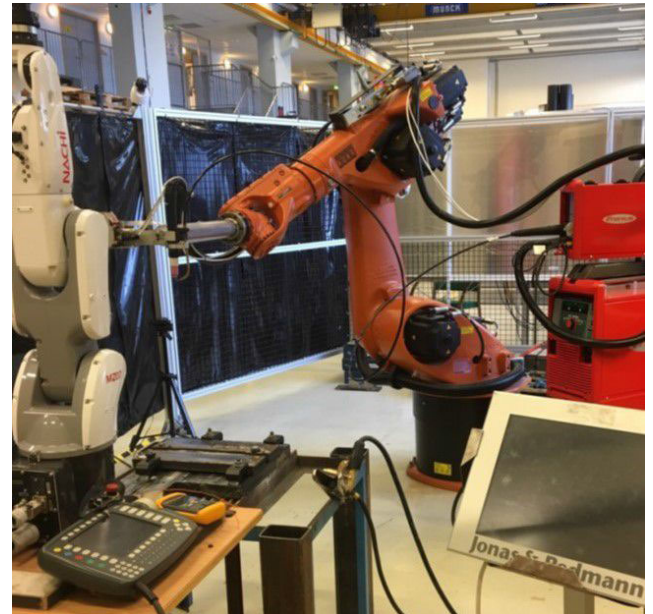


FIGURE 3. Robotic welding cell, where the experiments were carried out.

the independent validation. Seven WPVs combinations were selected arbitrarily from different regions of the parameter window, and then each welding trial was performed twice (Validation data sets in Table 2).

After each finished welding sequence, the weld beads' surface was recorded by a M2DW 160/40 Line Laser Triangulation Sensor (LTS). Each weld bead was measured by 0.1 mm increments along the weld line containing ca. 100 data points per profiles, which supplied the RSP raw scanned profiles as cross-sectional data for further data processing. The first and last fifth of the weld beads were neglected; thus, only the stabilized cross-sectional area was considered with an effective length of 60 mm.

### B. WELD BEAD PROFILE DATA ACQUISITION AND PREPROCESSING

The profile data of the weld beads were preprocessed (Step 2. in Fig 1) in a framework developed in LabVIEW™. The pseudo-code of the whole process is shown in Algorithm 1. The framework requires the measurement profile data from the LTS sensor (SMP), the WPVs from each trial, and the trajectory information of the welding robot system (RSP). As the first step, the required information was combined into weld bead profile data (CPD); thus, each cross-section of the weld beads contained aligned position information from the robot system with the corresponding actual welding parameters and measurement points.

The weld bead measurements were taken before and after the welding sections using the LTS scanner, producing the cross-sectional measurements along the weld line with a 0.1 mm incremental steps. Five consecutive CPD cross-section measurements were combined into one S profile for

**TABLE 2.** Mean values of the measurements and the predicted values of the proposed models. Dimensions of the experiment design:  $I$  arc current [A],  $U$  arc voltage [V],  $v_t$  torch travel speed [mm/s],  $v_f$  wire feed rate [mm/min]. Dimensions of the weld bead geometry:  $w$  bead width [mm],  $h$  bead height [mm], bead area [mm<sup>2</sup>].

ID	Experiment design				Measurements			BGP prediction			DPM prediction			MRA prediction		
	$I$	$U$	$v_t$	$v_f$	$w$	$h$	$A_B$	$w$	$h$	$A_B$	$w$	$h$	$A_B$	$w$	$h$	$A_B$
Training data sets:																
1	180	12	1.92	820	10.57	1.18	8.32	10.57	1.18	8.33	10.48	1.18	8.37	10.72	1.23	8.71
2	180	12	2.14	1040	10.26	1.41	9.63	10.26	1.41	9.63	10.40	1.40	9.62	10.42	1.37	9.54
3	180	12	2.36	1260	10.20	1.50	10.15	10.21	1.49	10.12	10.35	1.49	10.00	10.13	1.52	10.36
4	180	12	2.58	1480	9.59	1.68	10.76	9.59	1.68	10.74	10.03	1.65	10.76	9.84	1.66	11.19
5	180	12	2.80	1700	9.33	1.84	11.47	9.33	1.91	11.54	9.68	1.79	11.83	9.55	1.81	12.02
6	200	12	1.92	1040	11.37	1.37	10.36	11.36	1.37	10.35	11.51	1.36	10.35	11.41	1.36	10.42
7	200	12	2.14	1260	11.09	1.54	11.38	11.12	1.53	11.45	11.04	1.54	11.39	11.12	1.51	11.25
8	200	12	2.36	1480	11.11	1.60	11.86	11.04	1.60	11.96	10.99	1.60	11.79	10.83	1.65	12.06
9	200	12	2.80	820	10.28	0.86	5.91	10.07	0.86	6.03	10.20	0.86	5.80	10.00	0.87	5.30
10	220	12	1.92	1260	12.48	1.51	12.58	12.43	1.52	12.57	12.43	1.50	12.44	12.14	1.50	12.13
11	220	12	2.14	1480	12.44	1.61	13.36	12.48	1.62	13.29	12.32	1.62	13.31	11.85	1.64	12.95
12	220	12	2.36	1700	12.15	1.68	13.59	12.09	1.69	13.58	12.29	1.68	13.95	11.57	1.78	13.75
13	220	12	2.80	1040	10.92	1.04	7.53	10.93	1.01	7.59	11.34	1.02	7.55	10.73	1.01	7.02
14	240	12	1.92	1480	12.99	1.67	14.44	13.13	1.64	14.53	13.17	1.43	12.52	12.84	1.63	13.80
15	240	12	2.14	1700	12.82	1.68	14.39	12.76	1.69	14.41	12.68	1.67	14.14	12.54	1.78	14.66
16	240	12	2.58	1040	12.16	0.96	7.80	12.17	0.98	7.58	12.14	0.95	7.69	11.71	1.00	7.91
17	240	12	2.80	1260	11.71	1.04	8.09	11.70	1.06	8.11	11.84	1.05	8.06	11.42	1.15	8.73
18	260	12	2.14	820	12.87	0.87	7.46	12.83	0.87	7.44	13.01	0.91	8.30	13.02	0.84	7.96
19	260	12	2.36	1040	12.64	0.95	8.03	12.64	0.95	8.50	12.55	0.97	8.60	12.72	0.99	8.82
20	260	12	2.58	1260	12.25	1.11	9.08	12.25	1.11	9.03	12.61	0.98	8.13	12.44	1.13	9.62
21	260	12	2.80	1480	12.35	1.19	9.80	12.32	1.19	9.76	12.23	1.20	9.83	12.18	1.26	10.34
Validation data sets:																
1	200	12	2.60	900	10.13	1.00	6.77	10.16	1.00	6.78	9.77	1.09	7.19	10.33	1.01	6.69
2	220	12	2.30	1300	11.57	1.46	11.30	11.58	1.46	11.32	11.90	1.44	11.55	11.56	1.41	10.94
3	220	12	2.40	1350	11.15	1.50	11.18	11.20	1.51	11.19	11.72	1.35	10.69	11.42	1.43	10.93
4	230	12	2.20	1450	12.39	1.51	12.45	12.28	1.52	12.50	12.21	1.58	12.99	12.07	1.56	12.50
5	240	12	2.20	1700	12.08	1.85	14.88	12.24	1.84	14.63	12.40	1.69	14.01	12.45	1.76	14.42
6	240	12	2.30	1750	12.33	1.79	14.67	12.30	1.78	14.78	12.59	1.70	14.33	12.31	1.78	14.41
7	260	12	2.30	1500	12.29	1.54	12.60	12.28	1.54	12.55	12.18	1.53	13.00	12.93	1.46	12.54

each trial to reduce the noise of the LTS and the WPV measurements. This resulted in  $S = 120$  cross-sections for each trial, containing ca. 100 data points per profile. However, five percent of the cross-sections were discarded due to failed measurements caused by the material's locally high reflection.

During the data processing, each profile of each weld bead was filtered for noise, aligned to the horizontal position, and segmented to identify the weld bead. The segmentation provided the data points for the weld bead profiles and the substrate segment. In our case, the substrate segment was the flat base metal, but profile sections from other trials also appeared; however, they were discarded. The fitting method was chosen as the second-order polynomial, described in Eq. (1), to generate the BPP for the training and validation.

After the analysis, two sets of training patterns were generated in the Multiple-Input-Single-Output format. The first set was used for the BGP model tuning, consisting of three separated sets for the weld bead profile parameters ( $w, h, A_B$ ) with the input parameters of ( $U, I, v_t, v_f$ ) for each cross-section, 4795 cross-sections altogether. The other set was used for DPM model tuning, consisting of the weld bead profile points ( $Y$ ) and the input parameters of ( $X, U, I, v_t, v_f$ ) creating the 487 286 training patterns.

The overall measurement process time depends on the number of completed  $O_{pproc}$  pre-processings of the bead

profiles calculated as:

$$O_{pproc} = N_{trial} \cdot S \quad (4)$$

The validation sets were processed in the same way as the training patterns. The welding trials contained seven arbitrary WPV combinations with two parallel executions. The weld beads' length was 100 mm, and the first and last sections were discarded again. Altogether 1608 cross-sections were used with the same amount of validation patterns for the BGP model and 158 548 validation patterns for the DPM model.

### C. BACTERIAL MEMETIC ALGORITHM FOR TRAINING FUZZY SYSTEMS

In order to define the fuzzy systems for the weld bead models, we applied the BMA as a supervised trainer (Step 3. in Fig. 1) on the  $N_{pattern}$  training patterns [54]. In this case, the BMA minimizes the cumulative error between the output of the training patterns and the output of the fuzzy rule bases by adjusting the breakpoints of the fuzzy rules' trapezoids, encoded into the bacterium's chromosome [54]. The evaluation of the bacteria is carried out by the Mamdani Inference Model [66], and the result is given as the Sum of Squared Error (SSE) between the model output and the desired output of the patterns. This evaluation is used in all steps of the BMA.

The BMA consists of four main steps, first the random generation of the initial population, then the three main operations are performed in each generation. These three operations are the bacterial mutation (BM), the local search by the Levenberg-Marquardt method (LM), and the gene transfer (GT). The pseudo-code of the BMA method is shown in Algorithm 2, and the applied meta parameters are listed in Table 3. These meta-parameters are selected according to the problem's size to provide the balance between the required computational time and residual error of the estimation. The parameter settings are deduced from the preliminary experiments, based on the experience gained from the earlier application of the BMA on other problems [54], [64], [65].

The  $R_i$  rules in the fuzzy system are given in the following form:

$$R_i: \text{ IF } x_1 = A_{i,1} \text{ and } \dots \text{ and } x_n = A_{i,n} \text{ THEN } y = B_i,$$

where  $\mathbf{x} = (x_1, \dots, x_n)$  is the input vector,  $y$  is the output,  $A_i = (A_{i,1} \dots A_{i,n})$  is the antecedent parameter vector, and  $B_i$  is the consequent parts in the  $i$ -th rule.

The rule base is defined to cover the whole interpretation interval of the input variables to provide a valid inference result. The trapezoidal membership functions can be written as:

$$\mu_{A_{ij}}(x_j) = \begin{cases} \frac{x_j - a_{ij}}{b_{ij} - a_{ij}}, & \text{if } a_{ij} < x_j \leq b_{ij} \\ 1, & \text{if } b_{ij} < x_j \leq c_{ij} \\ \frac{d_{ij} - x_j}{d_{ij} - c_{ij}}, & \text{if } c_{ij} < x_j \leq d_{ij} \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

---

#### Algorithm 1 Processing of the Bead Profile Measurements

**Require:** *RawScannedProfiles* : RSP profile measurements

**Require:** *WeldingProcessVariables* : WPV parameters

**Require:** *RobotSystemParameters* : RSP parameters

**Require:**  $N_{\text{trial}}, N_{\text{profile}}[ID_{\text{trial}}]$

**Combine** RSP, WPV, and RSP into CPD combined profiles data

**for** Each trial **do**

**Combine** five CPD profiles into one  $S$  profile

**for** Each  $S$  profile **do**

        Filter measurement and correct errors

        Profile Segmentation

        Curve fitting

        Acquire profile properties

**end for**

**end for**

**Export** P1 Training patterns (Weld Bead Profile Properties)

**Export** P2 Training patterns (Profile Points of the Bead Segments)

---

where  $a_{ij} \leq b_{ij} \leq c_{ij} \leq d_{ij}$  denote the four breakpoints of the membership function of the  $i$ -th rule and the  $j$ -th input variable. The  $a_i, b_i, c_i, d_i$  values belong to the output membership function in the  $i$ -th rule. As in the original Mamdani algorithm, the *minimum* operator is used as the t-norm in the inference mechanism meaning that the degree of matching of the  $i$ -th rule in the case of an  $N_{\text{input}}$ -dimensional crisp  $\mathbf{x}$  input vector is:

$$w_i = \min_{j=1}^{N_{\text{input}}} \mu_{A_{ij}}(x_j). \quad (6)$$

The output of the fuzzy inference is defined as *maximum* aggregation. The defuzzification is calculated by the *Center of Sums* technique, which can be given in the explicit formula as shown in Equation (7), where the number of rules is  $N_{\text{rule}}$ , and  $N_{\text{input}}$  is the number of the input dimensions.

The operation of the BMA starts with the generation of the initial population, generating  $N_{\text{ind}}$  random individuals with the restriction to maintain the trapezoidal characteristics of the membership functions in the fuzzy rules. The number of rules ( $N_{\text{rule}}$ ) is defined beforehand and set to a low number to avoid the model overfitting. The total number of the created membership functions is  $N_{\text{ind}} \cdot (N_{\text{input}} + 1) \cdot N_{\text{rule}}$  where  $N_{\text{input}} = 4$  is the number of input variables and each membership function has four parameters. In the iteration phase of the algorithm, the bacterial mutation, local search, and gene transfer operations are performed until the number of generations ( $N_{\text{gen}}$ ) is reached.

Let us define  $k$  as the iteration variable, vector  $\mathbf{b}_k$  (the bacterium) contains all the parameters of the membership functions, and  $\mathbf{x}^{(p)}$  as the  $p$ -th training pattern's input vector. Here, the evaluation of the bacterium is carried out as

$$E(\mathbf{b}_k) = \|\mathbf{e}_k\|_2^2, \quad (8)$$

the 2-norm sum of squared error of the desired and model's output values, where the elements of the  $\mathbf{e}_k$  vector are calculated as

$$\mathbf{e}_k = \left[ e_k^{(p)} \right] = \left[ d^{(p)} - y_k(\mathbf{b}_k, \mathbf{x}^{(p)}) \right], \quad (9)$$

where  $d^{(p)}$  is the desired value given in the  $p$ -th training pattern, and  $y_k(\mathbf{b}_k, \mathbf{x}^{(p)})$  is the output value for the  $\mathbf{x}^{(p)}$  inputs provided by fuzzy system encoded in the  $\mathbf{b}_k$  bacteria.

---

#### Algorithm 2 Bacterial Memetic Algorithm

**Require:** *Patterns* :  $N_{\text{pattern}}$  patterns

**Require:** *Parameters* :  $N_{\text{ind}}, N_{\text{gen}}, N_{\text{rule}}, N_{\text{clone}}, l_{\text{bm}}, M_{\text{unit}}, LM_{\text{prob}}, LM_{\text{iter}}, \gamma_{\text{init}}, \tau, N_{\text{inf}}, l_{\text{gt}}, I_{\text{unit}}$

Create initial population

**for**  $gen = 1$  to  $N_{\text{gen}}$  **do**

**for**  $ind = 1$  to  $N_{\text{ind}}$  **do**

        Bacterial mutation

        Levenberg-Marquardt algorithm

**end for**

    Gene transfer in the population

**end for**

---

TABLE 3. Parameters of the BMA.

Parameter Name	DPM	BGP	Optimization
$N_{patterns}$	487286	4795	–
$N_{gen}$	50	100	500
$N_{ind}$	30	30	50
$N_{clones}$	3	3	8
$M_{unit}/M_{type}$	point	point	uniform
$l_{bm}$	1	1	1
$LM_{prob}$	0.2	0.2	0.1
$LM_{iter}$	8	8	8
$\gamma_{init}$	1.00	1.00	1.00
$\tau$	0.0001	0.0001	0.0001
$N_{inf}$	20	20	20
$l_{unit}$	rule	rule	–
$l_{gt}$	1	1	1

### 1) BACTERIAL MUTATION

The bacterial mutation is applied one by one to each bacterium (Algorithm 3). First,  $N_{clone}$  clones of the rule base are generated, which are then subjects of random changes in their genes according to the mutation unit ( $M_{unit}$ ), which can either be a breakpoint of the trapezoid, an entire trapezoid, or an entire rule. The number of modified genes during this mutation is given by the mutation segment length ( $l_{bm}$ ) parameter of the algorithm. All clones are evaluated, and the best individual transfers the mutated part into the other individuals. In the end, only the best rule base is kept. The bacterial mutation is repeated  $N_{segment}$  times, where  $N_{segment}$  depends on  $M_{unit}$  and  $l_{bm}$ , and its value is  $4 \cdot N_{rule} \cdot (N_{input} + 1)/l_{bm}$  in the case of point mutation.

In each generation, the computational cost of the bacterial mutation operator can be defined according to the  $O_{BM}$  num-

---

#### Algorithm 3 Bacterial Mutation

---

**Require:** *Parameters* :  $N_{ind}, N_{clone}, l_{bm}, M_{unit}$   
 Create  $(N_{clone} + 1)$  clones of *Bacterium*  
 Set  $N_{segment} = 4 \cdot N_{rule} \cdot (N_{input} + 1)/l_{bm}$   
**for**  $i = 1$  to  $N_{segment}$  **do**  
   Select  $l_{bm}$  yet unmutated random gene  
   **for**  $j = 1$  to  $N_{clone}$  **do** Mutate selected gene in *Clone<sub>j</sub>*  
   **end for**  
   **for**  $j = 1$  to  $N_{clone} + 1$  **do** Evaluate *Clone<sub>j</sub>* using Eq. (8)  
   **end for**  
   Select *BestClone*  
   Transfer best clone's gene to all clones  
**end for**  
 Set *Bacterium* = *BestClone*

---

ber of fitness function calls:

$$O_{BM} = N_{ind} \cdot N_{clone} \cdot N_{segment}. \quad (10)$$

### 2) LEVENBERG-MARQUARDT ALGORITHM

After the bacterial mutation step, the Levenberg-Marquardt algorithm (Algorithm 4) is used for each individual to solve the minimization problem.

The local search was carried out with a  $LM_{prob}$  local search probability for each individual, maximum  $LM_{iter}$  number of iteration steps until the  $\tau$  terminal condition is met.

Let denote  $\mathbf{J}(\mathbf{b}_k)$ , as the Jacobian matrix of  $\mathbf{b}_k$  bacterium:

$$\mathbf{J}(\mathbf{b}_k) = \left[ \frac{\partial y_k(\mathbf{b}_k, \mathbf{x}^{(p)})}{\partial \mathbf{b}_k^T} \right] \quad (11)$$

Equation (11) expresses, that each row of the  $\mathbf{J}(\mathbf{b}_k)$  matrix contains the partial derivatives of the output calculated by the Mamdani inference method for the given  $\mathbf{x}^{(p)}$  input pattern according to the  $\mathbf{b}_k$  bacterium encoded parameters. The detailed calculations of derivatives of the  $\mathbf{J}(\mathbf{b}_k)$ , are given in [54].

The calculation of the Levenberg-Marquardt  $\mathbf{s}_k$  update vector is carried out as

$$\mathbf{s}_k = -(\mathbf{J}^T(\mathbf{b}_k)\mathbf{J}(\mathbf{b}_k) + \gamma_k \mathbf{I})^{-1} \mathbf{J}^T(\mathbf{b}_k) \mathbf{e}_k, \quad (12)$$

where  $\gamma_k$  is the bravery factor initially set to positive, and  $\mathbf{I}$  is the identity matrix. The value of  $\gamma_k$  controls both the search direction and the magnitude of the update. If the value of  $\gamma_k$  converges towards zero, then the algorithm applies the Gauss-Newton method, if towards infinite, then the algorithm gives the steepest descent approach.

---

#### Algorithm 4 Levenberg-Marquardt Algorithm

---

**Require:** *Patterns* :  $N_{pattern}$  patterns  
**Require:** *Parameters* :  $LM_{prob}, LM_{iter}, \gamma_{init}, \tau$   
**if** *Random(probability)* <  $LM_{prob}$  **then**  $k = 1$   
   **while**  $k < LM_{iter}$  **and**  $\tau_k > \tau$  **do**  
     Calculate  $\mathbf{s}_k$  update vector as Eq. (12)  
     Calculate  $r_k$  trust region as Eq. (14)  
     Calculate  $\gamma_{k+1}$  bravery factor as Eq. (13)  
     Evaluate the update vector's effect as Eq. (15)  
     Calculate  $\tau_k$  stopping criteria  
      $k = k + 1$   
   **end while**  
**end if**

---

$$y(\mathbf{x}) = \frac{1}{3} \frac{\sum_{i=1}^{N_{rule}} 3w_i(d_i^2 - a_i^2)(1 - w_i) + 3w_i^2(c_i d_i - a_i b_i) + w_i^3(c_i - d_i + a_i - b_i)(c_i - d_i - a_i + b_i)}{\sum_{i=1}^{N_{rule}} 2w_i(d_i - a_i) + w_i^2(c_i + a_i - d_i - b_i)}. \quad (7)$$



**Algorithm 5** Gene Transfer

**Require:** Population :  $N_{ind}$  Bacterium  
**Require:** Parameters :  $N_{ind}, N_{inf}, l_{gt}, I_{unit}$   
**for**  $i = 1$  to  $N_{inf}$  **do**  
    Ascending order of the population according to Eq. (8)  
     $SrcBact = Random(0, \dots, N_{ind}/2 - 1)$   
     $DestBact = Random(N_{ind}/2, \dots, N_{ind})$   
    Select random  $l_{gt}$  genes  
    Transfer the selected genes from  $SrcBact$  to  $DestBact$   
**end for**

The value of parameter  $\gamma_k$  is adjusted dynamically depending on the value of  $r_k$  trust region:

$$\gamma_{k+1} = \begin{cases} 4\gamma_k, & \text{if } r_k < 0.25 \\ \gamma_k/2, & \text{if } r_k > 0.75 \\ \gamma_k, & \text{otherwise} \end{cases} \quad (13)$$

The value of  $r_k$  for the supervised training is calculated as:

$$r_k = \frac{E(\mathbf{b}_k) - E(\mathbf{b}_k + \mathbf{s}_k)}{E(\mathbf{b}_k) - \|\mathbf{J}(\mathbf{b}_k)\mathbf{s}_k + \mathbf{e}_k\|_2^2} \quad (14)$$

The evaluation of the local search success is verified as

$$\mathbf{b}_{k+1} = \begin{cases} \mathbf{b}_k + \mathbf{s}_k, & \text{if } Eval(\mathbf{b}_k + \mathbf{s}_k) < Eval(\mathbf{b}_k) \\ \mathbf{b}_k, & \text{otherwise} \end{cases} \quad (15)$$

$Eval$  function means  $E$  as defined in Eq. (8) during the supervised training. Equation (15) is interpreted as if the update vector modifies the bacterium towards the local minimum then we carry on with the new value, otherwise it is left unchanged. The iterations stops if maximum  $LM_{iter}$  number is reached or the  $\tau_k < \tau$  complex stopping criteria [54] is fulfilled.

The average  $O_{LM}$  computational cost describing the number of fitness calls of the Levenberg-Marquardt algorithm in one generation is:

$$O_{LM} = N_{ind} \cdot LM_{prob} \cdot LM_{iter}, \quad (16)$$

where an additional  $(4 \cdot N_{rule} \cdot (N_{input} + 1))^3$  computational cost is required for each fitness function call, due to the pseudo-inverse calculation of the  $\mathbf{J}(\mathbf{b}_k)$  Jacobian matrix.

3) GENE TRANSFER

The last operation in a generation is the horizontal gene transfer (Algorithm 5), allowing the recombination of genetic information between two bacteria. This operation is performed  $N_{inf}$  (number of infections) times in one generation. First, the population is split into two halves according to the fitness values. Then, a randomly chosen, better bacteria overwrites a randomly chosen, worse one's gene with its own. Here a gene (infection unit ( $I_{unit}$ )) means either a breakpoint of the trapezoid, or an entire trapezoid, or an entire rule. The  $l_{gt}$  parameter means the infection segment length, i.e., the number of overwritten genes.

The  $O_{GT}$  computational cost of the gene transfer operator in one generation is:

$$O_{GT} = N_{ind} \cdot \log(N_{ind}) + N_{inf} \quad (17)$$

The total  $O_{BMA}$  computational cost of the BMA can be estimated as:

$$O_{BMA} = (O_{BM} + O_{LM} + O_{GT}) \cdot N_{gen}, \quad (18)$$

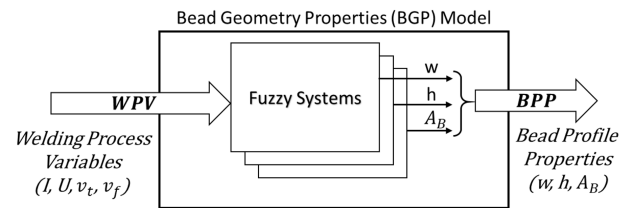
which is the number of fitness function calls during run time.

The number of executions of the Mamdani inference process (Eq. (7)) is:

$$O_{FS} = O_{BMA} \cdot N_{pattern}. \quad (19)$$

**D. BEAD GEOMETRY PROPERTIES MODEL**

The BGP model is following a classical approach for reconstructing the weld bead shape. The model provides a direct relation between the WPVs and BPPs, where a dedicated fuzzy system is defined for each property. The  $w, h,$  and  $A_B$  parameters are depending on the fitting model applied during the data preprocessing. The scheme of the model is shown in Fig. 4. The bead profiles, as parabola curves are reconstructed from the  $w, h$  parameters using the Equation (1).



**FIGURE 4.** Schematic structure of the bead geometry properties model.

The computational cost of building the BGP model is given according to the cost of the utilized fuzzy systems:

$$O_{BGP} = 3 \cdot O_{FS} \quad (20)$$

Since the BPP parameters are detached from each other, the BGP model's fuzzy systems can be separately used to estimate only the required BPPs in an application where the full shape is not needed.

**E. DIRECT PROFILE MEASUREMENTS MODEL**

The DPM model (Fig. 5) is an extended version of the BGP model and provides the  $y = f(x, WPVs)$  non-linear function. The central part of the model is the single fuzzy system, where the WPVs are extended with the bead profile's abscissa vector to estimate the corresponding ordinate directly for each profile point. The training patterns are generated from the measured  $N_{point}$  profile points of the cross-sections after preprocessing, following the process described in Sec. III-B.

The corresponding BPPs are extracted during the profile postprocessing, using the same parabola fitting principles as discussed in the BGP model development. Since the DPM model enables us to maintain the curve fitting model outside

the training, we can apply further fitting models and other considered limitations. This could allow examining different curve fitting functions such as cosine, arc, or ellipsoid, and also extending the model beyond the flat plate experiments as the subject of further research.

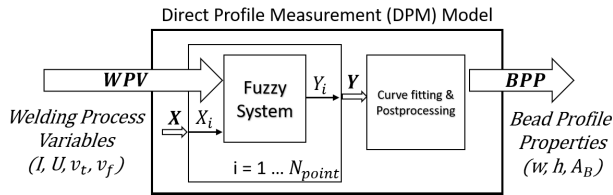


FIGURE 5. Schematic structure of the direct profile measurement model.

The computational cost of building the DPM model is  $O_{FS}$ , but the  $N_{pattern}$  number of patterns is much higher than in the case of BGP model (see Table 3). However, the cost of the utilization of the DPM model is:

$$O_{DPM} = N_{point} \cdot O_{FS}, \quad (21)$$

where  $N_{point}$  is defined by the required resolution of the estimated bead profile, with an additional cost of curve fitting to retract the BPPs.

#### F. OPTIMIZING THE WELDING PROCESS VARIABLES

It was previously shown how the BMA was used to tune the fuzzy systems during the model development. In this section, the BMA is utilized as an optimizer [64] (Step 4. Fig. 1) to identify different sets of WPVs to achieve the desired BPP by minimizing the value of the  $f_{obj}$  objective function on the predefined interpretation intervals. The evaluation of the  $f_{obj}$  includes the calculation of the fuzzy system response for each BPP. The target BPPs are defined by manual calculations or numerical calculations to satisfy the criteria of the mechanical loads and stresses of the workpiece. The bacteria encode the WPVs in their chromosomes. During the evaluation of the bacteria – which is used in all BMA steps – the bacteria's response is calculated by the BGP model for each target value, and the result is given as the value of the  $f_{obj}$ .

In our approach, the computational task involved the following three steps:

- 1) **Selection of target weld geometry** from the available sets of values of  $w$ ,  $h$ ,  $A_B$
- 2) **Running the optimization process** to obtain multiple combinations of WPVs (Algorithm 6)
- 3) **Verification of the produced results** for each target value

It was shown in the literature [32] that multiple combinations of welding process variables could be estimated to achieve a target weld bead geometry. On the full range of the search window, the BMA optimizer provided almost identical solutions at the global minimum. Therefore, each input variable's search window was segmented into  $N_{subint} = 5$

intervals, and the full range was left for the other inputs. For each segmented search window, multiple  $N_{run} = 3$  evaluation runs were performed, then the averaged values of WPVs contained in the best bacteria were given as the result of that segment. Due to the overlapping search windows, multiple identical WPV sets occurred and were unified to provide the final list of solutions.

At the start of the BMA operation, an initial population of 50 bacteria was defined. The meta parameters of the algorithm are presented in Table 3. Each bacterium in the population contains a set of randomly chosen WPVs. Values of the welding process variables  $I$ ,  $U$ ,  $v_i$  and  $v_f$  were chosen in the range of 160.0 – 280.0 A, 10.50 – 13.50 V, 1.80 – 3.40 mm/s, and 600 – 2100 mm/min, respectively. Since in the application of BMA as an optimizer, no training sets were applied, the course of the algorithm is different, and the evaluations carried out on vectors instead of matrices. The evaluation is replaced by calculating the value of the single-objective  $f_{obj}$  function for three weld bead profile properties:

$$f_{obj} = \lambda_1 \left( \frac{w^p}{w^t} - 1 \right)^2 + \lambda_2 \left( \frac{h^p}{h^t} - 1 \right)^2 + \lambda_3 \left( \frac{A_B^p}{A_B^t} - 1 \right)^2, \quad (22)$$

where  $\lambda_i$  are the weights of the objectives, set to constant 0.6, 0.3, 0.1, respectively, upper index  $p$  represents the BPP value estimated by the BGP model, and  $t$  represents the desired target BPP. The target BPP of the desired bead geometry is obtained experimentally; thus, the target values would maintain a realistic ratio to each other. The verification of the computed solutions requires that the set of WPVs used to produce the weld should appear with a small deviation in the provided solutions.

Further modifications are applied in the Levenberg-Marquardt algorithm, where the steps remain unchanged, but the definitions of  $r_k$ , and  $s_k$  are updated. The update vector is defined for a given bacterium at the  $k$ -th iteration as:

$$s_k = - \left( \mathbf{g}(\mathbf{b}_k) \otimes \mathbf{g}(\mathbf{b}_k)^T + \gamma_k \mathbf{I} \right)^{-1} \mathbf{g}(\mathbf{b}_k), \quad (23)$$

---

#### Algorithm 6 Welding Process Parameter Optimizer

---

**Require:** *Optimizer Parameters* : *Intervals*,  $N_{run}$ , *RuleBases*,  $BPP_{target}$ ,  $N_{subint}$ ,  $N_{input}$

**Require:** *BMA Parameters* :  $N_{ind}$ ,  $N_{gen}$ ,  $N_{clone}$ ,  $I_{bm}$ ,  $M_{unit}$ ,  $LM_{prob}$ ,  $LM_{iter}$ ,  $\gamma_{init}$ ,  $\tau$ ,  $N_{inf}$ ,  $I_{gt}$   
Define intervals and number of runs

```

for  $i = 1$  to  $N_{input}$  do
  for  $sub = 1$  to  $N_{subint}$  do
    Set recent sub-interval
    for  $run = 1$  to  $N_{run}$  do
      Bacterial memetic algorithm
    end for
  end for
end for
Remove duplicates and miss-matches

```

---

where  $\mathbf{g} = \nabla f_{obj}$  is the gradient vector, approximated by second-order accurate, central finite differences;  $\otimes$  stands for the dyadic product of two vectors, and the other parameters are defined as in Eq. (12). The value of  $r_k$  for the optimization is calculated as:

$$r_k = \frac{f_{obj}(\mathbf{b}_k + \mathbf{s}_k) - f_{obj}(\mathbf{b}_k)}{\mathbf{g}(\mathbf{b}_k)^T \mathbf{s}_k} \quad (24)$$

Value of  $\gamma_k$  still depends on  $r_k$  as given in Eq. (13). The bacterium  $\mathbf{b}_{k+1}$  is given according to Eq. (15), where the *Eval* function is calculated as the  $f_{obj}$ . The iteration stops if the stopping criteria  $\tau_k = \|\mathbf{g}(\mathbf{b}_k)\|_2 < \tau$  is fulfilled or maximum  $LM_{iter}$  number is reached.

In the optimizer version of the BMA the additional computational cost of  $O_{LM}$  is decreased to  $(N_{input})^3$  due to the different amount of calculations needed for the  $\mathbf{s}_k$  in the two roles. The computation cost, i.e. the number of objective function calls (Eq. (22)) in the BMA optimizer is:

$$O_{opt} = N_{input} \cdot N_{subint} \cdot N_{run} \cdot O_{BMA}, \quad (25)$$

due to the segmentation of the intervals of the input parameters. The WPV sets encoded in the bacterium's chromosome were improved through the generations using the BMA (Sec. III-C, meta parameters in Table 3), monitored by calculating the  $f_{obj}$  objective function values for each set of bacteria. A bacterium with a low  $f_{obj}$  value indicates that the WPVs in it produces a target geometry with a small deviation from the estimated geometry.

#### IV. RESULTS AND DISCUSSION

In this section, the results of the development work will be discussed, including the overview of the main findings of the weld bead profile modeling, the comparison with other published models, and the evaluation of the WPVs optimization. The computations were carried out on a PC using an Intel® Core™ i7-5820K Processor at 3.30 GHz and an NVIDIA GeForce® GTX 970 graphics card. The trained models' performance was evaluated by comparing the estimated and the measured values to define the goodness of the fitting using the root mean square error (RMSE); the normalized RMSE value to the output range (NRMSE); and the *R* Pearson product-moment correlation coefficient. The normalization was carried out on the range of the measured properties  $w : 8.82 - 13.75 \text{ mm}$ ,  $h : 0.93 - 2.19 \text{ mm}$ , and  $A_B : 6.18 - 16.88 \text{ mm}^2$ .

The evaluation of fitting was performed on an independent validation data set to avoid the models' overfitting, and the data was preprocessed in the same way as the training patterns. The comparison to other models includes models from the literature and a multiple regression analysis (MRA) model on the available data set, see in Sec. IV-B.

##### A. EVALUATION OF THE TRAINED MODELS

The training of the fuzzy systems of the BGP model was carried out on multiple numbers of rules between two and seven. The required calculation time depended on the number

**TABLE 4. Comparison of the RMSE and NRMSE values between the training and validation data for the BGP model.**

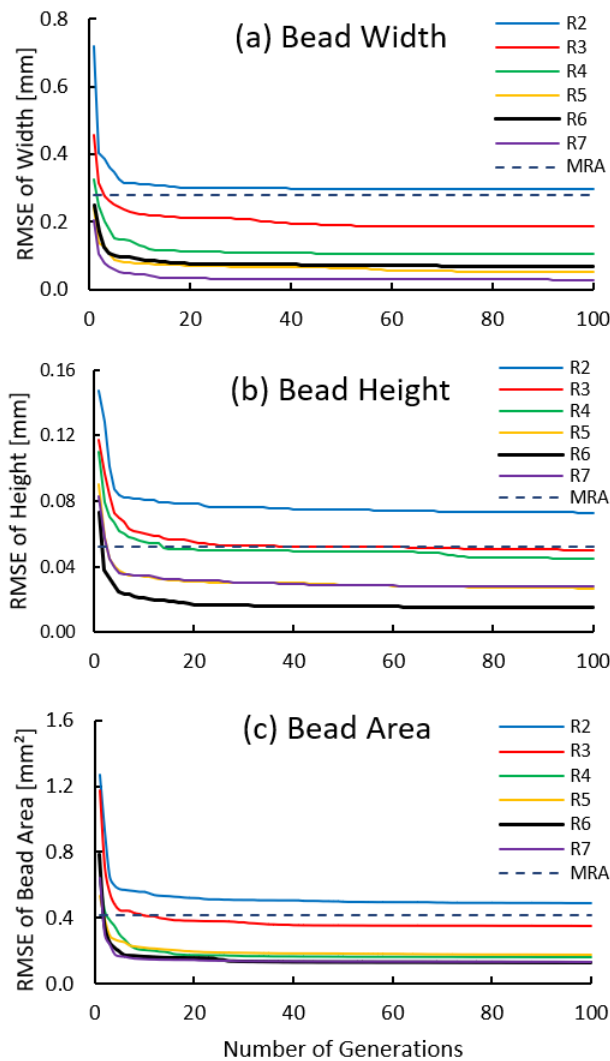
Rules	Width		Height		Area	
	Train.	Valid.	Train.	Valid.	Train.	Valid.
MRA	0.2781	0.3293	0.0525	0.0597	0.4143	0.2610
R2	0.2970	0.3025	0.0729	0.0431	0.4867	0.2754
R3	0.1859	0.1291	0.0503	0.0440	0.3489	0.2082
R4	0.1043	0.1305	0.0450	0.0301	0.1626	0.2197
R5	0.0514	0.0808	0.0268	0.0194	0.1721	0.1595
R6	0.0683	0.0769	0.0153	0.0054	0.1245	0.1082
R7	0.0287	0.0765	0.0279	0.0140	0.1303	0.1154
NRMSE values:						
MRA	0.0565	0.0669	0.0385	0.0439	0.0366	0.0230
R2	0.0603	0.0614	0.0536	0.0317	0.0429	0.0243
R3	0.0378	0.0262	0.0369	0.0323	0.0308	0.0184
R4	0.0212	0.0265	0.0331	0.0221	0.0144	0.0194
R5	0.0104	0.0164	0.0197	0.0142	0.0152	0.0141
R6	0.0139	0.0156	0.0113	0.0040	0.0110	0.0095
R7	0.0058	0.0155	0.0205	0.0103	0.0115	0.0102

**TABLE 5. Comparison of the RMSE and NRMSE values between the training and validation data for the DPM model.**

Rules	Width		Height		Area	
	Train.	Valid.	Train.	Valid.	Train.	Valid.
MRA	0.2781	0.3293	0.0525	0.0597	0.4143	0.2610
R6	0.2617	0.2971	0.0745	0.1240	0.5969	0.8496
R7	0.1960	0.3344	0.0561	0.1295	0.3840	0.9632
R8	0.2096	0.4268	0.0431	0.1117	0.3131	0.8319
R9	0.1855	0.3739	0.0679	0.1260	0.4560	0.6949
R10	0.1957	0.4150	0.0501	0.1662	0.3030	1.1079
R11	0.2760	0.4320	0.0615	0.1023	0.5345	0.5092
NRMSE values:						
MRA	0.0565	0.0669	0.0385	0.0439	0.0366	0.0230
R6	0.0532	0.0603	0.0548	0.0911	0.0527	0.0750
R7	0.0398	0.0679	0.0412	0.0952	0.0339	0.0850
R8	0.0426	0.0867	0.0317	0.0821	0.0276	0.0734
R9	0.0377	0.0760	0.0499	0.0926	0.0402	0.0613
R10	0.0398	0.0843	0.0368	0.1221	0.0267	0.0978
R11	0.0561	0.0877	0.0452	0.0752	0.0472	0.0449

of rules. As long as for two rules (R2), one generation for each parameter was calculated just under 2 s requiring 191 s for the whole calculation. For six rules (R6), one generation is calculated 5 s, and the whole calculation took 501 s; for seven rules (R7), the run times were 6 s and about 10 min. The response time of the model for the input parameter change is approximately 5 ms.

Table 4 shows the comparison of the RMSE values obtained on the training and the validation data for the property estimation of weld bead profiles. It can be seen that the roughest estimation is given by the FS with two rules. Its estimation accuracy is similar to the MRA model. However, when the number of rules increases, the estimation accuracy is also increasing, and the best result for the validation data was obtained using six rules (R6) for all three geometry properties. In the case of estimating the bead width, R6 and R7 provide a similar result, but the difference is in the last digit; thus, the lower number of rules is preferred. Figure 6 illustrates the evolutionary process for each BPP to compare



**FIGURE 6.** Evolutionary process with different number of rules for each Bead Profile Property (BPP) and comparison to the MRA model output (RMSE values).

the RMSE values over the generations. In each case, the R6 is among the best performers, and the estimation accuracy increases with the number of applied rules. Furthermore, the steepest increase in the fitting is observed in the first tens of generation, with a slow but steady increase during the later generations. These results support the quick convergence of the BMA towards the global optimum and increase accuracy as long as the algorithm is running.

The rules of the BGP model for each BPP are shown in Fig. 7 and tabulated in Table 6. Each row represents one FS estimating a BPP, and the membership functions of the same rule are marked by the same color.

The interpretation of the rule bases shows that the weld beads' width is not affected by the amount of the deposited metal, but the applied arc current and torch travel speed. The weld bead height has a base value, which is independent of the WPVs changes, but the additional gain affected mostly the

applied material (increase) and the applied current (decrease). The rule base shows that the weld bead area calculation depends on the variation in the torch travel speed and the wire feed rate.

These findings strengthen the applicability of the proposed method since it provided a reasoning base from experimental data, providing similar assumptions that are part of an experienced welder's knowledge base. Therefore, these rules can be used in training or supporting a robot cell's commissioning when monitoring the degree of changes in the weld geometry while tuning the WPVs. Another possible application is in the WPV optimization to achieve a predefined bead geometry.

In Fig. 8, the correlation between the estimated and the measured values is illustrated for the models with the best performing rule sets. The results are compared with the expected values and the estimated values of the reference MRA model. In the case of the BGP model, correlation coefficients for the training are 0.9970, 0.9970, and 0.9975, respectively, with an RMSE value of 0.0683, 0.0153 and, 0.1245. Consequently, the estimated and the measured values correlate with a high degree, and the approximation accuracy in the training is above 98.5 percent. Similarly, the validation set's correlation coefficients are 0.9903, 0.9997, and 0.9984, showing a similar good correlation. The RMSE values are 0.0769, 0.0054, and 0.1082, meaning that the model estimates unknown values within its application range under one percent of the average error for each parameter.

The training of the DPM model required a higher number of fuzzy rules but one system instead of three since it created an estimation model for the curve fitting itself. The training was also carried out with a various number of rules as their RMSE values are presented in Table 5. The best-fitting rule number was found as six (R6) for estimating  $w$  with a 5.3 and 6.0 percent of average error for the training and the validation set. Eleven rules (R11) provided the best estimations for  $h$  and  $A_B$ , with an average approximation accuracy of 95 percent. The calculation required 215 min for up to eight rules, 392 min for nine rules (R9), and 646 min for eleven rules (R11) in total for the 50 generations on the same hardware used for the BGP model. The response time of the model for the input parameter change is approximately 8 ms. A similar execution time was experienced up to eight rules (but as the number of rules increased, a raise in required time was observed) since the model calculation utilized the advantages of the GPU provided parallel programming.

As Fig. 8 shows, a good agreement exists between the actual values and the estimated parameters of the weld bead profiles. The BPPs of the training data set were estimated with the correlation coefficients of 0.9734, 0.9649, and 0.9528. Similarly, 0.9001, 0.9069, and 0.9604 values were achieved for the validation set. As observed, the width estimation showed the best result at six rules, and the estimation was overfitted on the training data as the rule number increased. The height and bead area approximation accuracy increased with the number of applied rules.

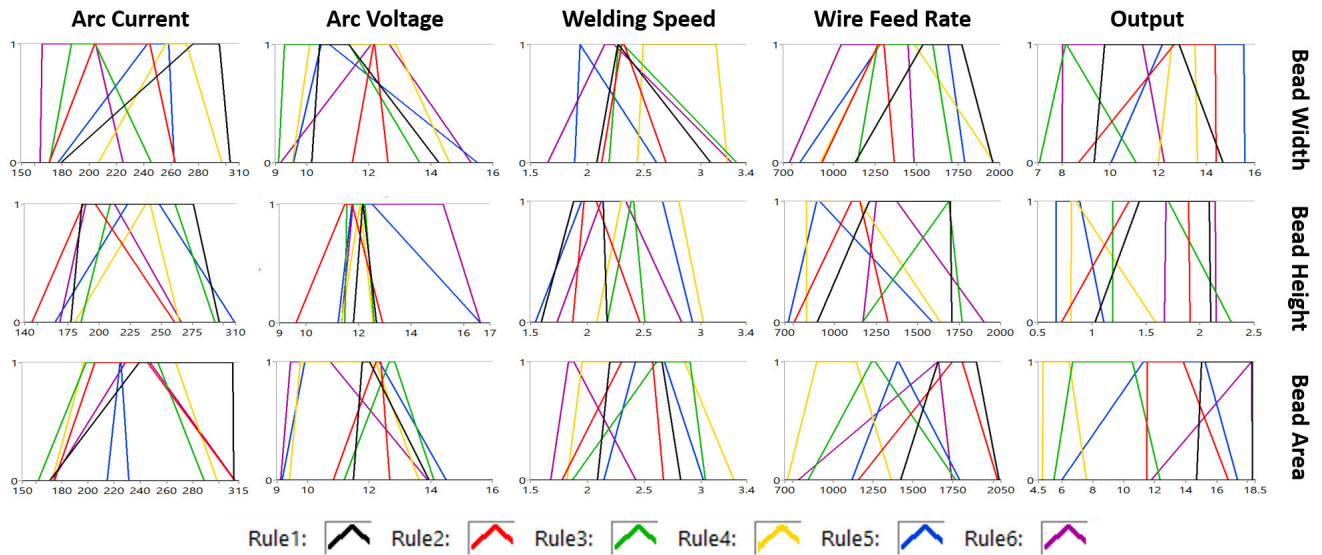


FIGURE 7. Rule bases of the BGP model to estimate BPPs from WPVs.

TABLE 6. Mamdani Fuzzy system definitions for the bead profile parameters by the BGP model.

ID	Current [A]	Voltage [V]	Welding speed [mm/s]	Feed rate [mm/min]	Output
Fuzzy System of Bead Width (Output in [mm]):					
R1	[179.1 276.1 295.5 303.4]	[10.17 10.42 11.37 14.27]	[2.08 2.27 2.27 3.09]	[1129 1538 1773 1957]	[ 9.33 9.77 12.86 14.68]
R2	[170.3 203.9 244.3 263.0]	[11.47 12.16 12.18 12.62]	[2.13 2.31 2.33 2.69]	[ 930 1269 1304 1367]	[ 8.68 12.66 14.36 14.40]
R3	[170.9 186.9 204.4 245.5]	[ 9.10 9.28 11.38 13.64]	[2.19 2.27 2.32 3.32]	[1143 1265 1599 1713]	[ 7.07 8.15 8.19 11.09]
R4	[206.5 255.9 271.5 297.4]	[ 9.54 10.10 12.87 14.59]	[2.45 2.49 3.14 3.24]	[ 918 1274 1487 1966]	[12.00 12.57 13.52 13.62]
R5	[176.9 242.6 258.2 262.3]	[ 9.58 10.46 10.74 15.51]	[1.89 1.94 1.94 2.61]	[ 799 1280 1689 1791]	[10.02 12.16 15.56 15.58]
R6	[164.0 165.4 204.8 224.8]	[ 9.16 12.08 12.67 15.29]	[1.65 2.15 2.24 3.27]	[ 736 1048 1449 1484]	[ 8.01 8.03 11.35 12.24]
Fuzzy System of Bead Height (Output in [mm]):					
R1	[177.3 186.0 276.0 296.8]	[11.81 12.16 12.17 12.68]	[1.60 1.88 2.14 2.18]	[ 895 1209 1698 1707]	[ 1.03 1.44 2.08 2.10]
R2	[145.9 187.2 196.8 260.7]	[ 9.62 11.47 11.76 12.91]	[1.87 1.97 2.08 2.47]	[ 756 1103 1153 1321]	[ 0.72 1.34 1.89 1.91]
R3	[185.2 208.8 261.3 293.5]	[11.48 11.55 12.25 12.60]	[2.18 2.39 2.41 2.51]	[1167 1688 1689 1768]	[ 1.19 1.19 1.71 2.29]
R4	[180.4 237.5 241.2 264.6]	[11.33 12.05 12.07 12.56]	[2.08 2.30 2.81 3.03]	[ 829 829 1151 1637]	[ 0.81 0.81 0.85 1.59]
R5	[164.6 223.0 248.6 309.4]	[11.23 11.75 12.53 16.64]	[1.54 1.95 2.67 2.93]	[ 721 893 910 1590]	[ 0.67 0.67 0.89 1.11]
R6	[168.1 189.3 212.4 266.1]	[11.35 11.79 15.22 16.63]	[1.74 2.12 2.34 2.83]	[1172 1252 1376 1903]	[ 1.67 1.68 2.14 2.15]
Fuzzy System of Bead Area (Output in [mm <sup>2</sup> ):					
R1	[170.6 239.1 310.5 311.2]	[11.48 11.78 12.01 13.94]	[2.09 2.20 2.66 2.82]	[1423 1658 1897 2038]	[14.70 15.01 18.28 18.32]
R2	[174.5 205.3 245.4 312.2]	[10.84 12.22 12.38 12.66]	[1.78 2.30 2.57 2.67]	[1159 1747 1806 2027]	[11.48 11.50 13.85 16.75]
R3	[161.9 198.5 253.4 288.7]	[11.19 12.65 12.82 14.10]	[1.87 2.61 2.90 3.03]	[ 846 1241 1266 1767]	[ 5.48 6.70 10.56 12.33]
R4	[172.7 197.1 266.8 298.6]	[ 9.42 9.77 12.20 13.61]	[1.82 1.96 2.86 3.29]	[ 716 900 1145 1365]	[ 4.76 4.78 6.48 7.58]
R5	[214.6 224.5 225.0 231.1]	[ 9.17 9.91 12.33 14.50]	[2.15 2.42 2.68 3.02]	[1119 1402 1412 1793]	[ 6.01 11.26 15.24 17.36]
R6	[172.5 228.4 247.3 312.5]	[ 9.13 9.45 10.73 13.89]	[1.68 1.83 1.88 2.42]	[ 784 1654 1659 1743]	[11.77 18.26 18.31 18.31]

The comparison of the two models in both cases showed good fitness for the training and the verification data sets. However, the DPM model provided a bit of noisier output. The correlation coefficients for the verification sets were slightly worse than that of the training sets, which can be explained by the sensitivity of curve-fitting on the width and height parameters during the data preprocessing. The computational cost of the training of the DPM model is 30-times more than that of the BGP model, which corresponds to the increased number of samples. Despite the additional complexity of the DPM during the BPPs feature extraction, both models responded within 10 ms. Due to the quicker

response, the BGP model was utilized during the evaluation phase of the optimization process.

### B. COMPARISON TO OTHER MODELS

To provide further evaluation of the models' performances, a comparison was made with the MRA model - using the same datasets - and with similar models from the literature. The regression analysis was carried out with the Data Analysis tools of Microsoft Excel™. The coefficients of the analysis are tabulated in Table 7. The Pearson correlation coefficients were found to be  $w$ : 0.9493,  $h$ : 0.9715, and  $A_B$ : 0.9745 for

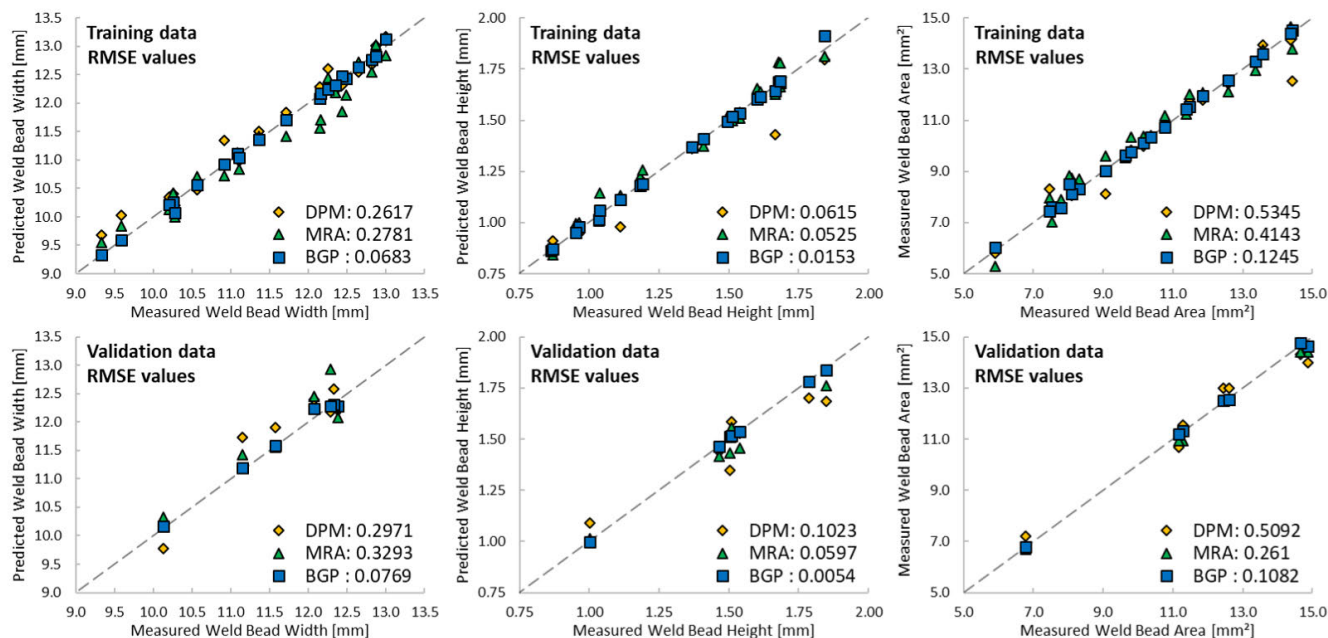


FIGURE 8. Comparison between the expected and estimated values of the fuzzy systems for each output (BGP with the estimated outputs, DPM model with the fitted and extracted values).

the training, and  $w$ : 0.8698,  $h$ : 0.9728, and  $A_B$ : 0.9964 for the validation.

The RMSE values of the MRA model are listed in the first rows of Table 4 and Table 5. Their values fit into the list of the proposed models between R2 and R3, meaning that using two rules in the fuzzy systems of the BGP model can have a similar accuracy than using the MRA. However, the increased rule number provides more accurate estimations of the proposed models. In the case of the DPM model, the trend is not that visible, but it has a similar performance as the MRA, as shown in Table 5.

The performance of the developed models was also compared with already published models. The comparison is given in Table 8 for the bead width estimation, and in Table 9 for the bead height. Models estimating the weld beads' cross-sectional area were not presented in the literature since the area is usually calculated from the bead width and height or even neglected due to prioritizing other parameters. As discussed in the introduction, a limited number of bead property estimation models exist for TIG welding in the literature. Therefore a few models developed for Metal Inert Gas (MIG) welding were also included. The presented model's estimation accuracy was normalized to the corresponding range of the output parameter, discussed in the reference. Unfortunately, in the study by Xiong *et al.* [12], the error was given as the mean absolute percentage error; thus, it was included instead of the NRMSE because of the model's comparably good performance.

The  $w$  bead width estimation models are presented in Table 8. The models presented by Subashini and Vasudevan [25] and Bestard *et al.* [22] (Table 8 1., 4., 5.) are modeling methods based on analyzing the image of an IR camera to

TABLE 7. Coefficients for the multi-variable regression analysis.

Profile Property	Intercept [—]	$I$ [A]	$U$ [V]	$v_t$ [mm/s]	$v_f$ [mm/min]
Width [mm]	6.6237	0.0330	0.0769	-1.5470	0.0002
Height [mm]	2.3753	-0.0040	-0.0516	-0.3142	0.0010
Area [mm <sup>2</sup> ]	12.7651	0.0011	-0.2482	-3.9040	0.0076

estimate the width of weld beads during welding. The other models are based on the classical CI approaches to estimate the bead geometry by alternating the selected process variables. As the results show, only the Infra Red (IR) imaging-based ANN model by Subashini and Vasudevan [25] showed slightly better estimation accuracy (1.32% error) than our proposed BGP model (1.56%). However, in that case, only the value of  $I$  arc current is altered from the WPVs.

Furthermore, this high accuracy was observed only in this setup, and different network architecture was more in the range of the best performing classical CI models. Furthermore, the proposed DPM model provides a 6.03% estimation error, which is still better than some other models. Comparing the results to the performance of other BMA developed FS with different rules from Table 4 shows, that even with three rules (R3), the estimation accuracy is among the best models.

Estimating the  $h$  bead height, our BGP model was found to be better than the models of the literature, with a highly accurate estimation (0.4% error). The approximation accuracy of the DPM model was not outstanding, but comparable good to the existing methods. It provided estimations with 6.03%, 7.52%, and 4.49% NRMS error for the width, height, and area of the weld bead, respectively. Therefore,

we can state that our proposed method, including the bacterial memetic algorithm, can be utilized to develop welding applications, which can outperform many of the existing models, as demonstrated with the BGP model.

**TABLE 8. Performance of reference models for predicting width of the weld bead (RMSE values are normalized to the parameter range).**

No.	Model type	Method	NRMSE	Variables	Ref.
	BMA FS(tranf)	TIG	0.0156	$I, U, v_t, v_f$	BGPR6
	BMA FS(tranf)	TIG	0.0603	$X, I, U, v_t, v_f$	DPMR6
	Regression	TIG	0.0669	$I, U, v_t, v_f$	MRA
1.	ANN(4-10-1)	A-TIG	0.0132	IR image, $I$	[25]
2.	ANN(4-12-2) <sup>1</sup>	MIG	0.0192	$U, v_t, v_f, D_w$	[12]
3.	Regression <sup>1</sup>	MIG	0.0263	$U, v_t, v_f, D_w$	[12]
4.	ANFIS(gaussmf)	A-TIG	0.0346	IR image, $I$	[25]
5.	MLP ANN	MIG	0.0355	IR image, $U, v_t, v_f$	[22]
6.	ANFIS(trimf)	TIG	0.0564	$I, U, v_t$	[26]
7.	FLR(trimf)	MIG	0.1029	$U, d_w, v_t, v_f$	[20]
8.	FS(trimf)	TIG	0.4531	$I, v_t, l_a$	[24]

<sup>1</sup> Mean Absolute Percentage Error

**TABLE 9. Performance of reference models for predicting height of the weld bead (NRMSE values are normalized to the parameter range).**

No.	Model type	Method	NRMSE	Variables	Ref.
	BMA FS(tranf)	TIG	0.0040	$I, U, v_t, v_f$	BGPR6
	BMA FS(tranf)	TIG	0.0752	$X, I, U, v_t, v_f$	DPMR11
	Regression	TIG	0.0439	$I, U, v_t, v_f$	MRA
1.	ANN(4-12-2) <sup>1</sup>	MIG	0.0210	$U, v_t, v_f, D_w$	[12]
2.	Regression <sup>1</sup>	MIG	0.0231	$U, v_t, v_f, D_w$	[12]
3.	BP-ANN	MIG	0.0725	$I, U, v_t$	[23]
4.	Regression	TIG/MIG	0.0863	Gas, $I, U, v_t, v_f$	[21]
5.	GA-NN	TIG	0.1430	$I, U, v_t, v_f$	[29]

<sup>1</sup> Mean Absolute Percentage Error

**TABLE 10. Various Set of WPV obtained by the BMA optimized BGP model to achieve the target geometry.**

ID	$I$	$U$	$v_t$	$v_f$	$w$	$h$	$A_B$
(i)	182.9	11.27	2.01	864	10.40	1.18	8.17
(ii)	193.3	11.27	2.34	1028	10.40	1.18	8.17
(iii)	197.2	11.40	2.76	1143	10.40	1.18	8.17
(iv)	206.0	11.27	2.82	1143	10.40	1.18	8.17
(v)	222.2	12.20	3.05	1157	10.40	1.18	8.17

Bead geometry was obtained experimentally using the following WPVs:  $I = 220.0A, U = 12.10V, v_t = 3.00mm/s, v_f = 1200mm/min$   
 Target geometry:  $w = 10.40mm, h = 1.18mm, A_B = 8.17mm^2$

**C. EVALUATION OF THE OPTIMIZATION**

The WPV optimization by the BMA provides an application of the developed fuzzy rule bases since the BGP model is used during the evaluation of the objective function.

The initialization of the BMA optimization process requires the specification of the target BPPs. The set of  $\{w = 10.40 mm, h = 1.18 mm, A_B = 8.17 mm^2\}$  was chosen from the validation sets of the experiment, produced by using the following WPVs:  $\{I = 220.0 A, U = 12.10 V, v_t = 3.00 mm/s, v_f = 1200 mm/min\}$ . The obtained WPVs combinations are expected to contain a similar set to the

one used in the experiment. During the optimization process, the progressive reduction of the  $f_{obj}$  objective function values presented by Eq. (22) of the best bacteria indicated that the solutions were improving by generations and converging towards an optimum. By the end of the iterations, the calculated values of the  $f_{obj}$  were under  $10^{-5}$ , indicating that the provided geometric parameters of the solutions agreed well with the corresponding desired experimental values.

The result of the optimization process is shown in Table 10. Significantly different values of the WPVs are provided as solutions, indicating the multiple paths to obtain the specified bead geometry. As Table 10 shows, the arc current values ranged from 183.0 to 222.0 A, the arc voltage varied between 11.30 and 12.20 V, the welding speed changed from 2.00 to 3.05 mm/s, and the wire feed rate took the interval of 864 to 1157 mm/min in various sets of optimized WPVs. Note that the values of WPVs in solution (v) of Table 10 is recognizable as the corresponding experimental values. The parameters were verified by the DPM model and found a sub-digit deviation from the expected values.

Therefore, the application of our model could support the decision to select WPV combinations to produce a certain target bead geometry. This is important when the robotic operation needs to be initialized or when alternating those parameters required to initialize the numerical analysis of the produced weld.

**V. CONCLUSION**

The paper proposed a fuzzy system-based method, providing high accuracy models to describe the weld bead geometry (WBG) from the welding process variables (WPVs). The bacterial memetic algorithm (BMA) provided a suitable tool to tune the membership functions of the fuzzy systems (FS) in order to model the weld bead geometry (WBG). Furthermore, it optimized the WPVs to produce a specified WBG by listing multiple solutions.

Based on experimental data, two models were developed to estimate the weld bead geometry. The Bead Geometry Property (BGP) model followed the more traditional approach by defining the relationship between the WPVs and one WBG property at a time. The Direct Profile Measurement (DPM) model utilized the measured profile points and described the bead profiles from points by a non-linear function realized in the form of fuzzy rules.

The proposed BGP model with six fuzzy rules outperformed the models from the overviewed literature and estimated the width, height, and cross-sectional area of the bead geometry with 1.56%, 0.40%, and 0.95% normalized root means square error (NRMSE), respectively. Furthermore, the BGP with at least three rules performed among the best models of the literature. The DPM model provided above 92% of estimation accuracy, with an NRMSE of 6.03%, 7.52%, and 4.49% for the width, height, and cross-sectional area, placing the DPM model into the average region. In both cases, the BMA tuned rule bases provided a proper tool to interpret the models' behavior on the inputs' changes.

The developed FSs were applied to optimize the WPVs to produce a specified weld bead geometry. A single objective function defined the optimization problem of the WPVs as a combined least square error function of the three WBG properties. The estimated WBG properties were calculated by the evaluation of the previously developed FS. The performance of the optimizer was tested by experimental data to define a realistic bead geometry. The proposed method provided multiple, distinctly different solutions of WPVs, and contained a recognizable match of the variable set, which produced the initial target geometry. This proved the method's capability to support the decision of WPV selection by listing candidates to be evaluated in order to fulfill additional criteria of the welding process. The outcome of this research is being implemented in a robotic welding application in the industry.

## ACKNOWLEDGMENT

The authors would like to thank the UiT The Arctic University of Norway and PPM Robotics AS to provide laboratory infrastructure of this work through the CoRoWeld Project. They would also like to thank Dr. Gábor Sziebig (now with SINTEF Manufacturing, Trondheim, Norway), and Tanja Kerezović Malešević (now with Trelleborg Group, Trelleborg, Sweden), the former laboratory members of the UiT The Arctic University of Norway, for the important contribution to the execution and consultancy of the welding experiments.

## REFERENCES

- [1] J. N. Pires, A. Loureiro, and G. Bolmsjö, *Welding Robots: Technology, System Issues and Application*. Springer, May 2006, doi: 10.1007/1-84628-191-1.
- [2] P. J. Modenesi, "The chemistry of TIG weld bead formation," *Weld. Int.*, vol. 29, no. 10, pp. 771–782, Oct. 2015. [Online]. Available: <http://www.tandfonline.com/doi/full/10.1080/09507116.2014.932990>
- [3] T. Sándor, C. Mekler, J. Dobránszky, and G. Kaptay, "An improved theoretical model for A-TIG welding based on surface phase transition and reversed Marangoni flow," *Metall. Mater. Trans. A*, vol. 44, no. 1, pp. 351–361, Jan. 2013. [Online]. Available: <http://link.springer.com/10.1007/s11661-012-1367-2>
- [4] A. Matsunawa and T. Ohji, "Role of surface tension in fusion welding (part 1): Hydrostatic effect," *Trans. JWRI*, vol. 11, no. 2, pp. 145–154, 1982. [Online]. Available: [https://ir.library.osaka-u.ac.jp/repo/ouka/all/5800/jwri11\\_02\\_145.pdf](https://ir.library.osaka-u.ac.jp/repo/ouka/all/5800/jwri11_02_145.pdf)
- [5] Y. Li, Y. Sun, Q. Han, G. Zhang, and I. Horváth, "Enhanced beads overlapping model for wire and arc additive manufacturing of multi-layer multi-bead metallic parts," *J. Mater. Process. Technol.*, vol. 252, pp. 838–848, Feb. 2018. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0924013617304661>
- [6] A. American Welding Society, *Welding Handbook: Welding Science and Technology*, vol. 1, 9th ed. Miami, FL, USA: American Welding Society, Jan. 2001.
- [7] S. Mishra and T. DebRoy, "A heat-transfer and fluid-flow-based model to obtain a specific weld geometry using various combinations of welding variables," *J. Appl. Phys.*, vol. 98, no. 4, Aug. 2005, Art. no. 044902. [Online]. Available: <http://scitation.aip.org/content/aip/journal/jap/98/4/10.1063/1.2001153>
- [8] K. Y. Benyounis and A. G. Olabi, "Optimization of different welding processes using statistical and numerical approaches—a reference guide," *Adv. Eng. Softw.*, vol. 39, no. 6, pp. 483–496, Jun. 2008. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0965997807001020>
- [9] D. K. Pratihari, "Expert systems in manufacturing processes using soft computing," *Int. J. Adv. Manuf. Technol.*, vol. 81, pp. 887–896, May 2015.
- [10] Y. Feng, Z. Chen, D. Wang, J. Chen, and Z. Feng, "DeepWelding: A deep learning enhanced approach to GTAW using multisource sensing images," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 465–474, Jan. 2020.
- [11] D. Fan, G. Zhang, Y. Shi, and M. Zhu, "Progress and trend in intelligent sensing and control of weld pool in arc welding process," in *Transactions on Intelligent Welding Manufacturing* (Transactions on Intelligent Welding Manufacturing), S. Chen, Y. Zhang, and Z. Feng, Eds. Singapore: Springer, 2019, pp. 27–43.
- [12] J. Xiong, G. Zhang, J. Hu, and L. Wu, "Bead geometry prediction for robotic GMAW-based rapid manufacturing through a neural network and a second-order regression analysis," *J. Intell. Manuf.*, vol. 25, no. 1, pp. 157–163, Feb. 2014. [Online]. Available: <https://link.springer.com/article/10.1007/s10845-012-0682-1>
- [13] T. DebRoy, H. L. Wei, J. S. Zuback, T. Mukherjee, J. W. Elmer, J. O. Milewski, A. M. Beese, A. Wilson-Heid, A. De, and W. Zhang, "Additive manufacturing of metallic components—process, structure and properties," *Prog. Mater. Sci.*, vol. 92, pp. 112–224, Mar. 2018. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0079642517301172>
- [14] L. Yuan, D. Ding, Z. Pan, Z. Yu, B. Wu, S. van Duin, H. Li, and W. Li, "Application of multidirectional robotic wire arc additive manufacturing process for the fabrication of complex metallic parts," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 454–464, Jan. 2020.
- [15] Y. Wang, C. Zhang, J. Lu, L. Bai, Z. Zhao, and J. Han, "Weld reinforcement analysis based on long-term prediction of molten pool image in additive manufacturing," *IEEE Access*, vol. 8, pp. 69908–69918, 2020.
- [16] H. C. Fang, S. K. Ong, and A. Y. C. Nee, "Adaptive pass planning and optimization for robotic welding of complex joints," *Adv. Manuf.*, vol. 5, no. 2, pp. 93–104, Jun. 2017. [Online]. Available: <http://link.springer.com/10.1007/s40436-017-0181-x>
- [17] C. M. Horváth and P. Korondi, "Supportive robotic welding system for heavy, small series production with non-uniform welding grooves," *Acta Polytechnica Hungarica*, vol. 15, no. 8, p. 25, 2018.
- [18] K. Somlo and G. Sziebig, "Aspects of multi-pass GTAW of low alloyed steels," *IFAC-PapersOnLine*, vol. 52, no. 22, pp. 101–107, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2405896319309917>
- [19] N. Murugan and V. Gunaraj, "Prediction and control of weld bead geometry and shape relationships in submerged arc welding of pipes," *J. Mater. Process. Technol.*, vol. 168, no. 3, pp. 478–487, Oct. 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0924013605003237>
- [20] Y. Xue, I. S. Kim, J. S. Son, C. E. Park, H. H. Kim, B. S. Sung, I. J. Kim, H. J. Kim, and B. Y. Kang, "Fuzzy regression method for prediction and control the bead width in the robotic arc-welding process," *J. Mater. Process. Technol.*, vols. 164–165, pp. 1134–1139, May 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0924013605002220>
- [21] C. Schneider, C. Lisboa, R. Silva, and R. Lermen, "Optimizing the parameters of TIG-MIG/MAG hybrid welding on the geometry of bead welding using the Taguchi method," *J. Manuf. Mater. Process.*, vol. 1, no. 2, p. 14, Oct. 2017. [Online]. Available: <http://www.mdpi.com/2504-4494/1/2/14>
- [22] G. Bestard, R. Sampaio, J. Vargas, and S. Alfaro, "Sensor fusion to estimate the depth and width of the weld bead in real time in GMAW processes," *Sensors*, vol. 18, no. 4, p. 962, Mar. 2018. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5948544/>
- [23] I. S. Kim, J. S. Son, C. E. Park, C. W. Lee, and Y. K. D. V. Prasad, "A study on prediction of bead height in robotic arc welding using a neural network," *J. Mater. Process. Technol.*, vols. 130–131, pp. 229–234, Dec. 2002. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0924013602008038>
- [24] H. Narang, U. Singh, M. Mahapatra, and P. Jha, "Prediction of the weld pool geometry of TIG arc welding by using fuzzy logic controller," *Int. J. Eng., Sci. Technol.*, vol. 3, no. 9, pp. 77–85, Oct. 2012. [Online]. Available: <https://www.ajol.info/index.php/ijest/article/view/82198>
- [25] L. Subashini and M. Vasudevan, "Adaptive neuro-fuzzy inference system (ANFIS)-based models for predicting the weld bead width and depth of penetration from the infrared thermal image of the weld pool," *Metall. Mater. Trans. B*, vol. 43, no. 1, pp. 145–154, Feb. 2012, doi: 10.1007/s11663-011-9570-x.



- [26] K. N. Gowtham, M. Vasudevan, V. Maduraimuthu, and T. Jayakumar, "Intelligent modeling combining adaptive neuro fuzzy inference system and genetic algorithm for optimizing welding process parameters," *Metall. Mater. Trans. B*, vol. 42, no. 2, pp. 385–392, Apr. 2011, doi: 10.1007/s11663-010-9471-4.
- [27] M. Vasudevan, "Soft computing techniques in stainless steel welding," *Mater. Manuf. Processes*, vol. 24, no. 2, pp. 209–218, Jan. 2009. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/10426910802612338>
- [28] A. Karpagaraj, K. Parthiban, and S. Ponmani, "Optimization techniques used in gas tungsten arc welding process—a review," *Mater. Today, Proc.*, vol. 27, no. 3, pp. 2187–2190, 2020. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S221478531933336X>
- [29] P. Dutta and D. K. Pratihari, "Modeling of TIG welding process using conventional regression analysis and neural network-based approaches," *J. Mater. Process. Technol.*, vol. 184, nos. 1–3, pp. 56–68, Apr. 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0924013606009137>
- [30] M. A. Kesse, E. Buah, H. Handroos, and G. K. Ayetor, "Development of an artificial intelligence powered TIG welding algorithm for the prediction of bead geometry for TIG welding processes using hybrid deep learning," *Metals*, vol. 10, no. 4, p. 451, Mar. 2020. [Online]. Available: <https://www.mdpi.com/2075-4701/10/4/451>
- [31] H. Geng, J. Li, J. Xiong, X. Lin, and F. Zhang, "Optimization of wire feed for GTAW based additive manufacturing," *J. Mater. Process. Technol.*, vol. 243, pp. 40–47, May 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0924013616304137>
- [32] S. Mishra and T. DebRoy, "Tailoring gas tungsten arc weld geometry using a genetic algorithm and a neural network trained with convective heat flow calculations," *Mater. Sci. Eng., A*, vols. 454–455, pp. 477–486, Apr. 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0921509306025111>
- [33] J. Pan, S. Hu, L. Yang, and H. Li, "Simulation and analysis of heat transfer and fluid flow characteristics of variable polarity GTAW process based on a tungsten-arc-specimen coupled model," *Int. J. Heat Mass Transf.*, vol. 96, pp. 346–352, May 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0017931016000028>
- [34] H. M. E. Ramos, S. M. O. Tavares, and P. M. S. T. de Castro, "Numerical modelling of welded T-joint configurations using SYSWELD," *Sci. Technol. Mater.*, vol. 30, pp. 6–15, Dec. 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2603636318300538>
- [35] H. Zhang, H. Lu, C. Cai, and S. Chen, "Robot path planning in multi-pass weaving welding for thick plates," in *Robotic Welding, Intelligence and Automation* (Lecture Notes in Electrical Engineering), no. 88, T.-J. Tarn, S.-B. Chen, and G. Fang, Eds. Berlin, Germany: Springer, 2011, pp. 351–359. [Online]. Available: <http://link.springer.com/chapter/10.1007/978-3-642-19959-2-43>
- [36] D. Ding, Z. Pan, D. Cuiuri, and H. Li, "A multi-bead overlapping model for robotic wire and arc additive manufacturing (WAAM)," *Robot. Comput.-Integr. Manuf.*, vol. 31, pp. 101–110, Feb. 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0736584514000696>
- [37] S. Yan, H. Fang, S. Ong, and A. Nee, "Optimal pass planning for robotic welding of large-dimension joints with nonuniform grooves," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 232, no. 13, pp. 2386–2397, Jul. 2017, doi: 10.1177/0954405417718877.
- [38] S. Suryakumar, K. P. Karunakaran, A. Bernard, U. Chandrasekhar, N. Raghavender, and D. Sharma, "Weld bead modeling and process optimization in hybrid layered manufacturing," *Comput.-Aided Des.*, vol. 43, no. 4, pp. 331–344, Apr. 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0010448511000078>
- [39] Y. Cao, S. Zhu, X. Liang, and W. Wang, "Overlapping model of beads and curve fitting of bead section for rapid manufacturing by robotic MAG welding process," *Robot. Comput.-Integr. Manuf.*, vol. 27, no. 3, pp. 641–645, Jun. 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0736584510001729>
- [40] M. Vasudevan, A. K. Bhaduri, B. Raj, and K. P. Rao, "Genetic algorithm-based computational models for optimizing the process parameters of A-TIG welding to achieve target bead geometry in type 304 L(N) and 316 L(N) stainless steels," *Mater. Manuf. Processes*, vol. 22, no. 5, pp. 641–649, Jun. 2007. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/10426910701323342>
- [41] D. S. Nagesh and G. L. Datta, "Genetic algorithm for optimization of welding variables for height to width ratio and application of ANN for prediction of bead geometry for TIG welding process," *Appl. Soft Comput.*, vol. 10, no. 3, pp. 897–907, Jun. 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1568494609001975>
- [42] B. Panda, K. Shankhwar, A. Garg, and M. M. Savalani, "Evaluation of genetic programming-based models for simulating bead dimensions in wire and arc additive manufacturing," *J. Intell. Manuf.*, vol. 30, no. 2, pp. 809–820, Feb. 2019, doi: 10.1007/s10845-016-1282-2.
- [43] Y. Liu and Y. Zhang, "Iterative local ANFIS-based human welder intelligence modeling and control in pipe GTAW process: A data-driven approach," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1079–1088, Jun. 2015. [Online]. Available: <https://ieeexplore.ieee.org/document/6949113>
- [44] O. Cordón, Ed., *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (Advances in Fuzzy Systems). Singapore: World Scientific, 2001.
- [45] A. Fernandez, F. Herrera, O. Cordón, M. Jose Del Jesus, and F. Marcelloni, "Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?" *IEEE Comput. Intell. Mag.*, vol. 14, no. 1, pp. 69–81, Feb. 2019.
- [46] L.-X. Wang, "Fuzzy systems are universal approximators," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Mar. 1992, pp. 1163–1170. [Online]. Available: <https://ieeexplore.ieee.org/document/258721>
- [47] V. Kurková, "Kolmogorov's theorem and multilayer neural networks," *Neural Netw.*, vol. 5, no. 3, pp. 501–506, Jan. 1992. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0893608092900128>
- [48] J.-S.-R. Jang and C.-T. Sun, "Functional equivalence between radial basis function networks and fuzzy inference systems," *IEEE Trans. Neural Netw.*, vol. 4, no. 1, pp. 156–159, Jan. 1993. [Online]. Available: <http://ieeexplore.ieee.org/document/182710/>
- [49] L. T. Koczy, "Fuzzy if... then rule models and their transformation into one another," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 26, no. 5, pp. 621–637, Sep. 1996.
- [50] L. T. Koczy, D. Tikk, and T. D. Gedeon, "On functional equivalence of certain fuzzy controllers and RBF type approximation schemes," *Int. J. Fuzzy Syst.*, vol. 2, no. 3, pp. 164–175, 2000.
- [51] Q. Hancheng, X. Bocai, L. Shangzheng, and W. Fagen, "Fuzzy neural network modeling of material properties," *J. Mater. Process. Technol.*, vol. 122, nos. 2–3, pp. 196–200, Mar. 2002. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0924013602000195>
- [52] K. R. Naik and A. K. Khandelwal, "Effects of the bead geometry of MIG arc welding analysis by fuzzy logic method," *Int. J. Sci., Eng. Technol.*, vol. 5, no. 6, p. 6, 2017.
- [53] J. Botzheim, C. Cabrita, L. T. Koczy, and A. E. Ruano, "Estimating fuzzy membership functions parameters by the Levenberg-Marquardt algorithm," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, vol. 3, Jul. 2004, pp. 1667–1672.
- [54] J. Botzheim, C. Cabrita, L. T. Koczy, and A. E. Ruano, "Fuzzy rule extraction by bacterial memetic algorithms," *Int. J. Intell. Syst.*, vol. 24, no. 3, pp. 312–339, Mar. 2009. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/int.20338>
- [55] T. Bartz-Beielstein, J. Branke, J. Mehnen, and O. Mersmann, "Evolutionary algorithms," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 4, no. 3, pp. 178–195, May 2014, doi: 10.1002/swidm.1124.
- [56] B. Doerr and F. Neumann, Eds., *Theory of Evolutionary Computation: Recent Developments in Discrete Optimization* (Natural Computing Series). Cham, Switzerland: Springer, 2020. [Online]. Available: <http://link.springer.com/10.1007/978-3-030-29414-4>
- [57] N. Nawa and T. Furuhashi, "Fuzzy system parameters discovery by bacterial evolutionary algorithm," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 5, pp. 608–616, Oct. 1999.
- [58] P. Moscato, "On evolution, search, optimization, genetic algorithms and martial arts-towards memetic algorithms," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. C3P 826, 1989. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.27.9474>
- [59] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quart. J. Appl. Math.*, vol. 2, no. 2, pp. 164–168, Jul. 1944. [Online]. Available: <https://www.jstor.org/stable/43633451>
- [60] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Ind. Appl. Math.*, vol. 11, no. 2, pp. 431–441, Jun. 1963.

- [61] J. Botzheim, Y. Toda, and N. Kubota, "Bacterial memetic algorithm for offline path planning of mobile robots," *Memetic Comput.*, vol. 4, no. 1, pp. 73–86, Mar. 2012. [Online]. Available: <http://link.springer.com/10.1007/s12293-012-0076-0>
- [62] K. Balázs, J. Botzheim, and L. T. Kóczy, "Comparative investigation of various evolutionary and memetic algorithms," in *Computational Intelligence in Engineering* (Studies in Computational Intelligence), vol. 313. Berlin, Germany: Springer, 2010, pp. 129–140. [Online]. Available: [http://link.springer.com/10.1007/978-3-642-15220-7\\_11](http://link.springer.com/10.1007/978-3-642-15220-7_11)
- [63] D. Zhou, Y. Fang, J. Botzheim, N. Kubota, and H. Liu, "Bacterial memetic algorithm based feature selection for surface EMG based hand motion recognition in long-term use," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2016, pp. 1–7.
- [64] A. Csik, J. Botzheim, J. Balazs, T. Csoknyai, and J. L. Hontvari, "Energy and cost optimal design for the reconstruction of residential building envelopes by bacterial memetic algorithms," in *Proc. 6th Int. Conf. Soft Comput. Intell. Syst., 13th Int. Symp. Adv. Intell. Syst.* Kobe, Japan: IEEE, Nov. 2012, pp. 1226–1231. [Online]. Available: <http://ieeexplore.ieee.org/document/6505181/>
- [65] J. Botzheim and P. Földesi, "Novel calculation of fuzzy exponent in the sigmoid functions for fuzzy neural networks," *Neurocomputing*, vol. 129, pp. 458–466, Apr. 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0925231213009211>
- [66] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *Int. J. Man-Mach. Stud.*, vol. 7, no. 1, pp. 1–13, Jan. 1975. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0020737375800022>
- [67] R. N. Kacker, E. S. Lagergren, and J. J. Filliben, "Taguchi's orthogonal arrays are classical designs of experiments," *J. Res. Nat. Inst. Standards Technol.*, vol. 96, no. 5, pp. 577–591, 1991. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4927234/>



**JÁNOS BOTZHEIM** (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the Budapest University of Technology and Economics, Hungary, in 2001 and 2008, respectively. He is currently an Associate Professor with the Department of Mechatronics, Optics and Mechanical Engineering Informatics, Budapest University of Technology and Economics. His research interests include computational intelligence and cognitive robotics. He is also a member of several scientific societies, such as the John von Neumann Computer Science Society and the Hungarian Fuzzy Association.



**TRYGVE THOMESSEN** received the M.Sc. and Ph.D. degrees in mechanical engineering from the Norwegian Institute of Technology (NTH; currently, NTNU), in 1987 and 1992, respectively, with a specialization in industrial robotics. His professional career includes SINTEF (Research and Development Institute), a part time Professor at NTNU and the Budapest University of Technology and Economics, and a Visiting Professor at Chuo University, Hachioji. Since 2000, he has been the Managing Director of high-tech robot system integrator at PPM Robotics AS, Trondheim, Norway. He has successfully managed national and international research and development projects and industrial implementations, developing and implementing cutting-edge technology within industrial robotics, especially within sensor-based robot control of heavy industrial processes.



**PÉTER KORONDI** (Senior Member, IEEE) received the Dipl.-Eng. and Ph.D. degrees in electrical engineering from the Budapest University of Technology and Economics (formerly the Technical University of Budapest), Budapest, Hungary, in 1984 and 1995, respectively. Since 1986, he has been with the Budapest University of Technology and Economics. From April 1993 to April 1995, he worked in the laboratory of Prof. Harashima and Prof. Hashimoto with the Institute of Industrial Science, The University of Tokyo, Japan, where he continues to spend a month each year, working on a joint research. As a result of this cooperation, the Intelligent Integrated System Japanese–Hungarian Joint Laboratory was founded, in 2001. His research interests include telemanipulation and motion control. He is a Founding Member of the International PEMC Council.

...



**CSONGOR MÁRK HORVÁTH** received the B.Sc. and M.Sc. degrees in mechatronics engineering from the Budapest University of Technology and Economics, Budapest, in 2012 and 2014, respectively, where he is currently pursuing the Ph.D. degree in robotized, autonomous welding with the Department of Mechatronics, Optics and Mechanical Engineering Informatics. His research interests include robotics, computational intelligence, and human–machine interaction.