ilmedia

TECHNISCHE UNIVERSITÄT
ILMENAU

Prinz, Sebastian; Thomann, Jana; Eichfelder, Gabriele; Boeck, Thomas; Schumacher, Jörg

Expensive multi-objective optimization of electromagnetic mixing in a liquid metal

Original published in:	Optimization and engineering. - Dordrecht [u.a.] : Springer Science + Business Media B.V. - 22 (2021), 2, p. 1065-1089.
Original published:	2020-10-07
ISSN:	1573-2924
DOI:	10.1007/s11081-020-09561-4
[*Visited:*	2021-02-22]

This work is licensed under a Creative Commons Attribution 4.0 International license. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

TU Ilmenau | Universitätsbibliothek | ilmedia, 2022
http://www.tu-ilmenau.de/ilmedia

Optimization and Engineering
https://doi.org/10.1007/s11081-020-09561-4

RESEARCH ARTICLE

Expensive multi-objective optimization of electromagnetic mixing in a liquid metal

Sebastian Prinz[1] · Jana Thomann[2] · Gabriele Eichfelder[2] · Thomas Boeck[1] · Jörg Schumacher[1]

Received: 12 December 2019 / Revised: 29 May 2020 / Accepted: 7 September 2020
© The Author(s) 2020

Abstract

This paper presents a novel trust-region method for the optimization of multiple expensive functions. We apply this method to a biobjective optimization problem in fluid mechanics, the optimal mixing of particles in a flow in a closed container. The three-dimensional time-dependent flows are driven by Lorentz forces that are generated by an oscillating permanent magnet located underneath the rectangular vessel. The rectangular magnet provides a spatially non-uniform magnetic field that is known analytically. The magnet oscillation creates a steady mean flow (steady streaming) similar to those observed from oscillating rigid bodies. In the optimization problem, randomly distributed mass-less particles are advected by the flow to achieve a homogeneous distribution (objective function 1) while keeping the work done to move the permanent magnet minimal (objective function 2). A single evaluation of these two objective functions may take more than two hours. For that reason, to save computational time, the proposed method uses interpolation models on trust-regions for finding descent directions. We show that, even for our significantly simplified model problem, the mixing patterns vary significantly with the control parameters, which justifies the use of improved optimization techniques and their further development.

Keywords Multi-objective optimization · Expensive optimization · Trust-region method

✉ Gabriele Eichfelder
gabriele.eichfelder@tu-ilmenau.de

[1] Institut für Thermo- und Fluiddynamik, Technische Universität Ilmenau, Postfach 100565, 98684 Ilmenau, Germany

[2] Institut für Mathematik, Technische Universität Ilmenau, Postfach 100565, 98684 Ilmenau, Germany

Published online: 07 October 2020

1 Introduction

The use of electromagnetic induction to manipulate electrically conducting fluids is common in industrial applications, most notably in metallurgy, where time-dependent magnetic fields are used to generate a stirring motion inside a molten metal that is supposed to mix additives. A homogeneous distribution of these additives is desired since it usually has a strong influence on the quality of the final ingot. The electromagnetic forcing is then achieved by rotating magnetic fields that are, for example, generated by electromagnets (Eckert et al. 2007; Davidson 2001; Ben-David et al. 2014). Permanent magnets offer an interesting alternative to resistive electromagnets since they do not require a continuous supply of electrical currents to generate the electromagnet's magnetic field.

A possible scenario of generating a stirring motion inside a liquid metal is either by moving one or more permanent magnets with respect to the liquid metal (Prinz et al. 2016; Rivero et al. 2016; Beltrán et al. 2010) or by injecting an electric current that interacts with the magnetic field of a permanent magnet (Lara et al. 2017). Flows considered in the present manuscript are of the first type. In most cases, the length scale of the permanent magnet is small compared to the flow domain. Hence, induced Lorentz forces only affect a small fraction of the liquid. When the motion of the conducting fluid is driven by external forces, such as an applied pressure gradient, and passes a region of a non-uniform magnetic field, vorticity is generated, and the flow behaves similarly to hydrodynamic flow past a solid obstacle (Cuevas et al. 2006).

In general, the investigation of mixing processes in liquid metal flows is challenging. For experiments, one challenge lies in the opaqueness of the liquid metal, which precludes optical measurement techniques. There are additional dangers due to the reactivity and the elevated temperatures of many liquid metals. For numerical simulations that are used in the present work, the main difficulty is the computational demand, since such flows are usually three-dimensional and time-dependent. Furthermore, the generated flow and, therefore, its stirring properties, depend on the particular configuration characterized, e.g., by the movement of the magnet, the geometry of the flow domain as well as the distribution and the strength of the magnetic field.

New contribution from engineering perspective In the present paper we study a relatively simple numerical model for mixing in a liquid metal layer that is stirred by a harmonically oscillating permanent magnet. The generated flows are laminar and time-dependent. Mixing in such flows can be described and analyzed by methods that have been developed in the research field of chaotic advection (Aref 2017). Although it would be interesting in its own right, we do not attempt to achieve a detailed understanding of the physical mechanisms and mathematical properties of the particular flows from the viewpoint of chaotic advection.

Instead, we are interested in an optimization of the stirring process. The goals are to obtain a relatively homogeneous distribution of an initial local cloud of Lagrangian particles across the layer with a minimal amount of work that needs

to be done for moving the permanent magnet. The control parameters are the oscillation amplitude and frequency as well as the magnet field strength.

New contribution from optimization perspective A function evaluation is only possible by a time consuming simulation which can be considered as a black-box. Owing to that, the application of classical optimization methods such as gradient based descent methods is not possible. To reduce the computational work we propose a new algorithm based on the idea of trust regions (Conn et al. 2000). The algorithm is based on Thomann and Eichfelder (2019b). There, a trust-region based solver for multi-objective optimization problems was proposed which was developed for so called heterogeneous problems. This means that one of the objective functions is expensive in terms of computational time, while the others are inexpensive, i.e. given analytically.

The algorithm used in this work differs in two aspects. First, we apply it for two expensive functions rather than to a cheap and an expensive objective. This has only minor impact, as it only requires that the Taylor model used in Thomann and Eichfelder (2019b) for the cheap function has to be replaced by an interpolation model.

Second, we apply an acceptance test for the candidates for the next iterate. This change in the formulation of the algorithm seems to be small at first glance. However, it has a strong impact. From the practical point of view it guarantees a strict descent in each objective function from some starting configuration, which is of interest especially for our application. From the theoretical point of view, the new test is stronger, as will be shown. Nevertheless, the theoretical convergence proof from Thomann and Eichfelder (2019b) does not apply directly. Some modifications are necessary. The changes compared to the proof of the original version as presented in Thomann and Eichfelder (2019b) are detailed in about five pages in Chapter 4.6.3 of the dissertation (Thomann 2019) of the second author of the present work. These proofs are technical and require several additional results and assumptions. For that reason, we present the algorithm and the main ingredients in this paper only and refer to Thomann (2019) for the details of the proof of the new algorithm as presented here.

Related literature In the literature there are a lot of solution methods for multi-objective optimization problems. One common approach is scalarization (Miettinen 1999), i.e. to formulate a parameter dependent single-objective replacement problem. By using a weighted sum approach for such a scalarization we might loose the property of a strict descent in each objective function. With the ε-constraint method we loose the structure of our original problem which has box constraints only. For these reasons we will not scalarize our optimization problem.

Other methods for multi-objective optimization problems, like the generalized steepest descent method (Drummond and Svaiter 2005 ; Fliege and Svaiter 2000) or the generalized Newton method (Fliege et al. 2009) require derivative information. However, in our setting the derivatives are not available with reasonable efforts. There are also derivative-free methods such as direct search (Audet and Dennis 2006; Custodio et al. 2011; Audet et al. 2008). This approach only needs function values. As our objectives are supposed to be smooth, we propose here to use model functions on trust regions to reduce the numerical effort. For heterogeneous

problems, this was advantageous in the considered test instances, see Thomann and Eichfelder (2019a).

There are also other trust-region based multi-objective optimization solvers for expensive functions, see Ryu and Kim (2014). Trust region methods can easily be adapted for expensive functions as the original Taylor models can be replaced by interpolation models of the functions. The algorithm in Ryu and Kim (2014) is for bi-objective problems. It uses a scalarization technique and approximates the Pareto front, which is the set of optimal solutions in the image space. As the evaluation of the objective functions is very expensive in our setting, we abstain from finding an approximation of the Pareto front. Our approach is limited to improving a starting value toward one optimal solution.

For an approximation of the whole Pareto front, next to direct search approaches (see above), there exist also algorithms based on the idea of Bayesian global optimization and meta-model-assisted evolutionary computation. The paper Ponweiser et al. (2008) provides a good review and introduction to meta-model-assisted multiobjective algorithms which are based on the idea of so-called efficient global optimization, as for instance ParEGO (Knowles 2006). The significant difference to these approaches is that we will determine models locally, change the local area on which these models are calculated, and improve the models by choosing smaller local regions. The cited approaches determine the model over the full parameter space and improve the model towards promising or less explored regions of the parameter space, where the models are built using Gaussian processes or Kriging, cf. Emmerich et al. (2001). For a comparison of trust region methods with such surrogate methods we refer to Brockhoff et al. (2020).

Our setting contains a deterministic simulation, i.e. for each parameter choice and for various simulation runs at the same parameters, we will always obtain the same function value. In case of a Monte Carlo simulation this would be different and the function values will differ for different simulation runs. Then different methods would be necessary, see for instance Hunter et al. (2019).

As already mentioned above, we do not aim at approximating the Pareto front. Instead, we present an algorithm with a mathematical convergence guarantee, but also with a guarantee of an iterative improvement from a good starting guess. One reason are the time consuming simulation runs which are required for one function evaluation. As for example discussed in Emmerich et al. (2001), for a simple test instance with just two parameters and two objective functions with a very similar structure already 25 function evaluations are necessary. A good approximation of the Pareto front was generated, but no proof of the quality of the found solutions is possible. It is well known that the number of function evaluations scales up for an increase of the number of parameters. We required instead 62 function evaluations for the real-world problem with three parameters, but the algorithm stopped which is a guarantee of having found a Pareto critical point and thus a candidate for being efficient locally.

Structure of the manuscript The manuscript is structured as follows. First, we introduce the multi-objective optimization problem in Sect. 2. We give the basic definitions and present the new optimization procedure in Sect. 3. In particular, we discuss the used acceptance test. In Sect. 4 we describe the application

problem in detail, together with the physical model including the governing equations and the used numerical methods to solve them. Moreover, the general structure of the flow is briefly described. In Sect. 5 we present the results of the optimization. Concluding remarks are given in Sect. 6.

2 The multi-objective optimization problem and basic definitions

In this paper we study an application problem which can be modeled as an optimization problem with three variables, also called parameters. With $\mathbf{p} \in \mathbb{R}^3$ we will denote the parameter vector, i.e.

$$\mathbf{p} = \left(p_1, p_2, p_3\right)^\top = (\beta, \mathrm{KC}, \mathrm{Ha})^\top, \tag{1}$$

where the meaning of β, KC and Ha will be explained in Sect. 4.

The two objective functions $f_1 : \mathbb{R}^3 \to \mathbb{R}$ and $f_2 : \mathbb{R}^3 \to \mathbb{R}$ will have to be minimized w.r.t. box constraints. The feasible set for the main calculations is

$$\Omega = \{\mathbf{p} \in R^3 \mid 100 \le p_1 \le 1000, 1 \le p_2 \le 5, 10 \le p_3 \le 40\}. \tag{2}$$

The intervals for the parameters are chosen in a way that the numerical simulations can be reproduced by experiments under conventional laboratory conditions.

The considered multi-objective optimization problem (MOP) is then as follows:

$$\min_{\mathbf{p} \in \Omega} \left(f_1(\mathbf{p}), f_2(\mathbf{p})\right)^{\mathrm{T}} \tag{MOP}$$

where we will choose as first objective function the quality of the mixing, i.e. $f_1(\mathbf{p}) = \xi(\mathbf{p})$, and for the second objective function the work to be done by the magnet, i.e. $f_2(\mathbf{p}) = W(\mathbf{p})$. For both functions smaller values will mean better values, i.e. we are minimizing both functions. For the quality of the mixing this will be realized by a so-called mixing norm which should be as small as possible. The interpretation and calculation of these two objective functions will be given in detail in Sect. 4.

In the following, we give all definitions and results for the specific formulation (MOP) as we need them only for this setting. Naturally, the definitions extend to more than three parameters and to more than two objective functions. The algorithm which we are going to discuss attempts to find efficient solutions of (MOP). Recall that a feasible point $\mathbf{p} \in \Omega$ is efficient for (MOP) if there is no other feasible point $\mathbf{p}' \in \Omega$ with $f_i(\mathbf{p}') \le f_i(\mathbf{p})$, for $i = 1, 2$ and with $f_j(\mathbf{p}') < f_j(\mathbf{p})$ for at least one $j \in \{1, 2\}$.

In fact, the algorithm cannot guarantee to find an efficient solution of (MOP). The algorithm generates a sequence of points where the accumulation points satisfy some necessary optimality condition for a point to be efficient. But note that points which satisfy necessary optimality conditions might also not be efficient

or only locally efficient. For more details we refer to Thomann and Eichfelder (2019b). The necessary optimality condition is formulated in the next definition:

Definition 1 Let $f_i : \mathbb{R}^3 \to \mathbb{R}$, $i = 1, 2$, be continuously differentiable. A point $\mathbf{p} \in \mathbb{R}^3$ is called Pareto critical for (MOP) if for every vector $\mathbf{d} \in \mathbb{R}^3$ there exists an index $j \in \{1, 2\}$ such that $\nabla f_j(\mathbf{p})^\top \mathbf{d} \geq 0$ holds.

This concept is a generalization of the stationarity notion for single-objective optimization problems. Numerical methods for single-objective optimization typically also do not guarantee to find a globally optimal solution but only a point which satisfies some optimality conditions which are necessary for local optimality. Pareto criticality is a necessary condition for local weak efficiency, see for example Fliege and Svaiter (2000), and thus for efficiency as defined above.

The following lemma gives a characterization of Pareto critical points. It stems from multi-objective descent methods (Drummond and Svaiter 2005; Fliege et al. 2009; Fliege and Svaiter 2000). It is important for the description of the convergence of the proposed algorithm.

Lemma 1 *Let $f_i : \mathbb{R}^3 \to \mathbb{R}$ be continuously differentiable functions for all $i = 1, 2$. For the function*

$$\omega(\mathbf{p}) := -\min_{\|\mathbf{d}\| \leq 1} \max_{i=1,2} \nabla f_i(\mathbf{p})^\top \mathbf{d} \tag{3}$$

the following statements hold.

(i) *The mapping $\mathbf{p} \mapsto \omega(\mathbf{p})$ is continuous.*
(ii) *It holds $\omega(\mathbf{p}) \geq 0$ for all $\mathbf{p} \in \mathbb{R}^n$.*
(iii) *A point $\mathbf{p} \in \mathbb{R}^3$ is Pareto critical for (MOP) if and only if it holds $\omega(\mathbf{p}) = 0$.*

3 The trust-region based solver for expensive multi-objective optimization

In this section we describe the numerical approach for solving problems of the type (MOP). Recall that the difference to the algorithm presented in Thomann and Eichfelder (2019b) is that it works for the setting that all objective functions are expensive (and not that at least one is analytically given as assumed there). In addition, we make use of another trial point acceptance test. The proposed method can also be easily applied to more than two objective functions. As written, it assumes that all objective functions are expensive, i.e. not analytically given, and that function values are only accessible by time-consuming black-box simulations. Nevertheless, the approach assumes that the objective functions are smooth, i.e. differentiable, even

though derivatives will neither be calculated nor approximated. The latter would be too time consuming as many additional function evaluations are required for numerical differentiation.

The algorithm can also be applied to problems with a larger dimension of the parameter space. This would increase the number of function evaluations which will be required for building models of the objectives, see the next subsection for details. For the feasible set we have much stricter assumptions. The algorithm is developed for unconstrained problems and can be applied to problems with lower and upper bounds of the variables. However, more complex constraints cannot be handled.

To solve the optimization problem (MOP), we use an iterative algorithm which is based, as already mentioned, on the algorithm in Thomann and Eichfelder (2019b), see also Thomann (2019). The first modification compared to the algorithm there is that we use interpolation models for both objective functions, and we built the models by using a joint base of interpolation points. This also allows us to handle problems with two objective functions where function values are obtained by one simulation run. Another aspect is the trial point acceptance test for a new iterate, which guarantees a strict descent in each objective function. We discuss it in Sect. 3.3.

3.1 Description of the method

Let $k \in \mathbb{N}$ be an iteration index and $\mathbf{p}^k \in \mathbb{R}^3$ the current iteration point, i.e., we also have done a simulation of the fluid flow problem with parameter vector $\mathbf{p}^k$. In every iteration the computations are restricted to the local sphere

$$B_k := \left\{ \mathbf{p} \in \mathbb{R}^3 \mid \left\| \mathbf{p} - \mathbf{p}^k \right\| \leq \delta_k \right\}$$

called trust region defined by the current iteration point $\mathbf{p}^k$, a radius $\delta_k > 0$, and the Euclidean norm $\|\cdot\|$. The objective functions f_1 and f_2 are replaced by model functions $m_1^k, m_2^k : \mathbb{R}^3 \to \mathbb{R}$, in every iteration k. For both functions quadratic interpolation models based on Lagrange polynomials are used that satisfy the interpolation conditions

$$f_1(\mathbf{p}^k) = m_1^k(\mathbf{p}^k) \text{ and } f_2(\mathbf{p}^k) = m_2^k(\mathbf{p}^k) .$$

We used quadratic models in our implementation as recommended for small numbers of parameters. For more than 10 parameters, linear models need much less function evaluations to be built, see Table 1, which is then advantageous even while

Table 1 Function evaluations for computing one model function depending on the number of variables

Number of parameters	2	3	4	5	10	20	30	40	50
Quadratic model	6	10	15	21	66	231	496	861	1326
Linear model	3	4	5	6	11	21	31	41	51

these models are less accurate. Thus we need 10 function evaluations to built one quadratic model. But in future iterations, interpolation points can often be reused. This reduces the amount of function evaluations per iteration significantly.

Then, a search direction is computed by so-called local ideal points $\mathbf{q}^k = (q_1^k, q_2^k)^\top \in \mathbb{R}^2$ which use the individual minima of the model functions, i.e.

$$q_1^k = \min_{\mathbf{p} \in B_k} m_1^k(\mathbf{p}) \text{ and } q_2^k = \min_{\mathbf{p} \in B_k} m_2^k(\mathbf{p}).$$

This guarantees a descent for the model functions and, depending on the quality of the approximations, also for the original functions. Thus, the aim is to move as far as possible –as far as the trust region B_k allows– in this search direction $\mathbf{q}^k$. This is done by solving the auxiliary optimization problem

$$\begin{aligned} &\min\ t \\ \text{s.t.}\ & f_1(\mathbf{p}^k) + t(f_1(\mathbf{p}^k) - q_1^k) - m_1^k(\mathbf{p}) \geq 0 \\ & f_2(\mathbf{p}^k) + t(f_2(\mathbf{p}^k) - q_2^k) - m_2^k(\mathbf{p}) \geq 0 \\ & t \in \mathbb{R},\ \mathbf{p} \in B_k. \end{aligned} \tag{4}$$

The optimal solution is denoted by $(t^{k+}, \mathbf{p}^{k+})^\top$. The newly generated point $\mathbf{p}^{k+}$ is accepted or discarded based on a comparison of the model behavior with the original functions. The algorithm produces a sequence of iterates that converges to a Pareto critical point.

The criterion for deciding whether a newly generated point is discarded or not, i.e. the trial point acceptance test, differs from the criterion in Thomann and Eichfelder (2019b). As can be seen from the numerical experiments with this algorithm provided in Thomann and Eichfelder (2019a, Fig. 11), the original acceptance test does not necessarily guarantee a descent for both objective functions in each iteration. However, from a practical point of view it is better to improve both objective functions from some good starting guess. Therefore, we use another acceptance test in Sect. 3.3.

The algorithm uses a combination of three stopping criteria. Since the objective functions are expensive, the number of function evaluations is limited and the algorithm stops if the maximum allowed number is reached. Besides, it is proved in Thomann and Eichfelder (2019b) that the trust region radius δ_k converges to zero if the iteration points $\mathbf{p}^k$ converge to a Pareto critical point. Thus, the algorithm stops if the radius δ_k is smaller than a pre-defined constant $\varepsilon > 0$. The third stopping criterion also characterizes the behavior of approaching a Pareto critical point and is based on the accuracy of the model functions m_1^k, m_2^k and the step size that is obtained by solving the auxiliary optimization problem in Eq. (4). We did not insert these stopping criteria in the representation of the algorithm in Algorithm 1 for a better readability. The output of the algorithm is then the last iteration point $\mathbf{p}^k$, with additional information as for instance the size of the last trust region.

Running the algorithm produces one Pareto critical point in case all assumptions as detailed in Thomann (2019) (and adapted for two expensive functions, which is straightforward) are satisfied. Due to the design of the algorithm different starting

points result in general in different Pareto critical points. For the mixing problem, it cannot be expected to approximate the whole set of efficient points within a reasonable amount of time. This is due to the expensive simulation-given functions.

3.2 Algorithm

For the full algorithm see Algorithm 1.

Algorithm 1 Expensive multi-objective trust region algorithm

Input: Functions f_1, f_2, initial point $\mathbf{p}^0$, initial trust region radius $\delta_0 > 0$, parameters $0 < \eta_1 \leq \eta_2 < 1$, $0 < \gamma_1 \leq \gamma_2 < 1$

Step 0: Initialization
Set $k = 0$ and compute initial model functions m_i^k for $i = 1, 2$.

Step 1: Ideal point
Compute $\mathbf{q}^k = (q_1^k, q_2^k)$ by $q_i^k = \min_{\mathbf{p} \in B_k} m_i^k(\mathbf{p})$ for $i = 1, 2$.

Step 2: Trial point
If $r_i^k = f_i(\mathbf{p}^k) - q_i^k > 0$ holds for all $i \in \{1, 2\}$, compute $(t^{k+}, \mathbf{p}^{k+})$ by solving (4).
Otherwise, set $(t^{k+}, \mathbf{p}^{k+}) = (0, \mathbf{p}^k)$.

Step 3: Trial point acceptance test
If $t^{k+} = 0$ or $\phi_m^k(\mathbf{p}^k) - \phi_m^k(\mathbf{p}^{k+}) = 0$, set $\tilde{\rho}^k = 0$.
Otherwise, compute $f_i(\mathbf{p}^{k+})$, $i = 1, 2$, and
$\rho_i^k = \frac{f_i(\mathbf{p}^k) - f_i(\mathbf{p}^{k+})}{m_i^k(\mathbf{p}^k) - m_i^k(\mathbf{p}^{k+})}$ for $i = 1, 2$.
Set $\tilde{\rho}^k = \min_{i=1,2} \rho_i^k$.
If $\tilde{\rho}^k \geq \eta_1$, set $\mathbf{p}^{k+1} = \mathbf{p}^{k+}$, otherwise set $\mathbf{p}^{k+1} = \mathbf{p}^k$.

Step 4: Trust region update

$$\text{Set } \delta_{k+1} \in \begin{cases} [\gamma_1 \delta_k, \gamma_2 \delta_k] & , \text{ if } \tilde{\rho}^k < \eta_1 \\ [\gamma_2 \delta_k, \delta_k] & , \text{ if } \eta_1 \leq \tilde{\rho}^k < \eta_2 \\ [\delta_k, \infty) & , \text{ if } \tilde{\rho}^k \geq \eta_2 \end{cases}.$$

Step 5: Model update
Compute new model m_i^{k+1} for $i = 1, 2$, set $k = k + 1$ and go to **Step 1**.

The function $\phi_m^k : \mathbb{R}^3 \to \mathbb{R}$ is defined by $\phi_m^k(\mathbf{p}) = \max_{i=1,2} m_i^k(\mathbf{p})$ and will be explained in the next subsection.

3.3 Trial point acceptance test

Step 3 of Algorithm 1 is the trial point acceptance test in which it is decided if $\mathbf{p}^{k+}$ is accepted as next iteration point. In case it is not accepted, the trust region radius is reduced for the next iteration and the model functions are updated to improve their accuracy.

For the trial point acceptance test, the function values $m_i^k(\mathbf{p}^{k+})$, $i = 1, 2$, of the model functions are compared to the function values $f_i(\mathbf{p}^{k+})$, $i = 1, 2$, of the original functions, i.e. the prediction of the model functions is compared to the actual behavior of the original functions.

In the single-objective trust region approach with a scalar valued objective function $g : \mathbb{R}^3 \to \mathbb{R}$ this is realized by considering the quotient

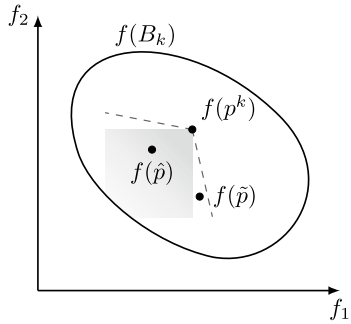

Fig. 1 Trial point acceptance test with ρ_ϕ^k and ρ_i^k, $i = 1, 2$

$$\frac{g(\mathbf{p}^k) - g(\mathbf{p}^{k+})}{m_g(\mathbf{p}^k) - m_g(\mathbf{p}^{k+})}.$$

The model function of g is denoted by $m_g : \mathbb{R}^3 \to \mathbb{R}$. If this quotient is larger than a given nonnegative constant, the trial point is accepted. This criterion can be transferred to multi-objective trust region approaches by applying it to the maximum over all functions. This was done in Thomann and Eichfelder (2019b) and is based on Villacorta et al. (2014):

$$\rho_\phi^k = \frac{\phi(\mathbf{p}^k) - \phi(\mathbf{p}^{k+})}{\phi_m^k(\mathbf{p}^k) - \phi_m^k(\mathbf{p}^{k+})}$$

with the functions

$$\phi(\mathbf{p}) = \max_{i=1,2} f_i(\mathbf{p}) \;\text{ and }\; \phi_m^k(\mathbf{p}) = \max_{i=1,2} m_i^k(\mathbf{p}) \,.$$

The trial point $\mathbf{p}^k$ is then accepted if $\rho_\phi^k \geq \eta_1$ holds with $\eta_1 > 0$.

Due to the determination of $\mathbf{p}^{k+}$ by (4) it holds $m^k(\mathbf{p}^{k+}) \leq m^k(\mathbf{p}^k)$ in all iterations $k \in \mathbb{N}$. Therefore, we conclude

$$\phi_m^k(\mathbf{p}^k) - \phi_m^k(\mathbf{p}^{k+}) \geq 0$$

for all $k \in \mathbb{N}$. Thus, if $\rho_\phi^k < 0$ holds, it follows $\phi(\mathbf{p}^k) - \phi(\mathbf{p}^{k+}) < 0$. As described in Thomann and Eichfelder (2019b), this guarantees only a descent for at least one objective function.

Another possibility to extend the trial point acceptance test from single-objective optimization to multi-objective optimization is to formulate it for every function individually, that is by considering the quotients

$$\rho_i^k = \frac{f_i(\mathbf{p}^k) - f_i(\mathbf{p}^{k+})}{m_i^k(\mathbf{p}^k) - m_i^k(\mathbf{p}^{k+})} \text{ for } i = 1, 2. \tag{5}$$

If both quotients are larger than a given nonnegative constant, the trial point is accepted. This trial point acceptance test is used for example in the trust region

approach from Ryu and Kim (2014). Using the acceptance test with ρ_i^k, $i = 1, 2$, guarantees a descent for every objective function. Thus, the latter acceptance criterion is stricter. This is proved in Lemma 2 below.

The difference is schematically illustrated in Fig. 1. Two areas are depicted: the gray shaded area includes the images of all points that would be accepted by the strict version of the acceptance test, as for example $\hat{p}$. This area is a subset of the area contoured by dashed lines. This larger region contains the images of those points that would be accepted by the trial point acceptance test defined by ρ_ϕ^k.

Lemma 2 *Let $k \in \mathbb{N}$ be an arbitrary index, $\eta_1 \in (0, 1)$ a constant and let the interpolation condition hold. If $\rho_i^k \geq \eta_1$ for all $i \in \{1, 2\}$, then it holds $\rho_\phi^k \geq \eta_1$.*

Proof Let $\rho_i^k \geq \eta_1$ hold for $i = 1, 2$. According to the interpolation condition, it holds $f(\mathbf{p}^k) = m^k(\mathbf{p}^k)$ for all $k \in \mathbb{N}$. This implies together with the definition of ρ_i^k

$$f_i(\mathbf{p}^k) - f_i(\mathbf{p}^{k+}) \geq \eta_1 \left(m_i^k(\mathbf{p}^k) - m_i^k(\mathbf{p}^{k+})\right) = \eta_1 \left(f_i(\mathbf{p}^k) - m_i^k(\mathbf{p}^{k+})\right)$$

for $i = 1, 2$. This is equivalent to

$$\left(1 - \eta_1\right) f_i(\mathbf{p}^k) \geq f_i(\mathbf{p}^{k+}) - \eta_1 m_i^k(\mathbf{p}^{k+})$$

for $i \in 1, 2$. Since it holds $\eta_1 \in (0, 1)$, $f_i(\mathbf{p}^k) \leq \phi(\mathbf{p}^k)$ and $m_i^k(\mathbf{p}^{k+}) \leq \phi_m^k(\mathbf{p}^{k+})$ for $i = 1, 2$, it follows that

$$\left(1 - \eta_1\right) \phi(\mathbf{p}^k) \geq f_i(\mathbf{p}^{k+}) - \eta_1 m_i^k(\mathbf{p}^{k+}) \geq f_i(\mathbf{p}^{k+}) - \eta_1 \phi_m^k(\mathbf{p}^{k+})$$

for $i \in 1, 2$. Let $j \in \{1, 2\}$ be the index with $f_j(\mathbf{p}^{k+}) = \phi(\mathbf{p}^{k+})$. Then it holds

$$\left(1 - \eta_1\right) \phi(\mathbf{p}^k) \geq f_j(\mathbf{p}^{k+}) - \eta_1 \phi_m^k(\mathbf{p}^{k+}) = \phi(\mathbf{p}^{k+}) - \eta_1 \phi_m^k(\mathbf{p}^{k+}).$$

It follows from the interpolation condition that

$$\phi(\mathbf{p}^k) - \phi(\mathbf{p}^{k+}) \geq \eta_1 \left(\phi(\mathbf{p}^k) - \phi_m^k(\mathbf{p}^{k+})\right) = \eta_1 \left(\phi_m^k(\mathbf{p}^k) - \phi_m^k(\mathbf{p}^{k+})\right).$$

From the definition of ρ_ϕ^k it then follows that $\rho_\phi^k \geq \eta_1$. □

Since the version of the trial point acceptance test using ρ_i^k, $i = 1, 2$, is stricter than the version using ρ_ϕ^k, it is possible that not as many iterations are successful when using ρ_i^k, i.e. the trial point is not accepted as often as with ρ_ϕ^k. Thus, the softer acceptance test can save function evaluations. However, we lose the property of improving the starting situation for each objective. Thus we use here the strict test.

With this strict version of the trial point acceptance test the convergence results from Thomann and Eichfelder (2019b) can be transferred with slight modifications. The sufficient decrease condition for the function ϕ_m^k has to be replaced by an analogous assumption for the model functions m_i^k for all $i \in \{1, 2\}$.

The remaining assumptions are quite technical but typical for convergence results for trust-region based methods with expensive functions. The required adaptions and modifications are discussed in detail in section 4.6.3 of the dissertation Thomann (2019). As describing the changes in the proof due to the new acceptance test is not possible without giving all assumptions and details of the full proof, we omit it here and refer instead to Thomann (2019, Subsection 4.6.3). The result is that the algorithm generates a sequence of iterates $\left\{\mathbf{p}^k\right\}_k$ with

$$\lim_{k\to\infty} \omega(\mathbf{p}^k) = 0.$$

If the sequence $\left\{\mathbf{p}^k\right\}_k$ has accumulation points, then all these points are Pareto critical for (MOP).

4 Description of the application

In this section, we describe the application problem and the numerical calculation of the objective functions in detail.

4.1 Physical model

Figure 2 shows an illustration of the present problem. The origin of the coordinate system is the geometric center of the liquid metal layer. The length scale L is the thickness of the liquid metal layer. In the following, we specify dimensions based on this scale, i.e. the thickness $L_z = 1$. The quadratic footprint of the box is 3×3 for $L_x \times L_y$. The rectangular permanent magnet has the dimensions of $1 \times 1 \times 0.5$ for $L_{m,x} \times L_{m,y} \times L_{m,z}$. It is uniformly magnetized along the z-direction. The center of the permanent magnet is

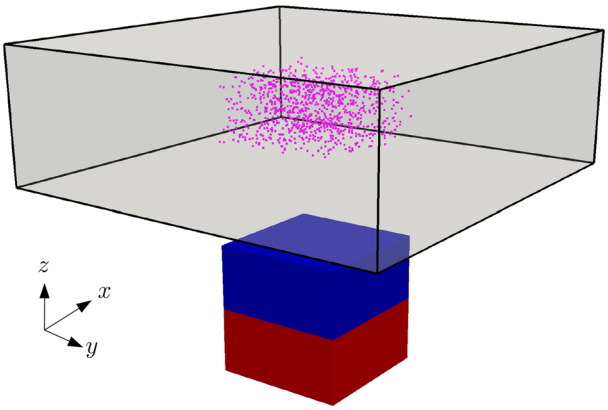

Fig. 2 Sketch of the present problem—the domain of the liquid metal cell is $L_x \times L_y \times L_z = 3 \times 3 \times 1$; a permanent magnet of size $L_{m,x} \times L_{m,y} \times L_{m,z} = 1 \times 1 \times 0.5$ oscillates in the x-direction with neutral position $x = 0$. The gap between the permanent magnet and the bottom of the liquid metal layer is 0.25; at $t = 0$, 2500 Lagrangian particles are seeded in subsection with the dimension $1 \times 1 \times 0.5$ in the center of the liquid metal layer

at $z_m = -1$, i.e. the gap between the bottom of the liquid metal layer and the surface of the magnet is 0.25. At $t = 0$, the position of the magnet is $x_m(t = 0) = y_m = 0$ and N_p mass-less particles are randomly seeded in a rectangular subsection of the domain with the dimensions 1×1 in the horizontal, and 0.5 in the vertical direction. The subsection is centered in the liquid metal layer. The magnetic field is normalized by B, which is the maximal value in the middle plane of the computational domain. All boundaries Γ are electrically insulating, i.e., the electrical current density vector field $\mathbf{j}$ has a vanishing normal component $\mathbf{j} \cdot \mathbf{n} = 0$ on Γ, where $\mathbf{n}$ is the surface-normal vector. The velocity vector field $\mathbf{u}$ satisfies the no-slip condition $\mathbf{u} = 0$ on Γ.

For low magnetic Reynolds numbers, the quasi-static approximation of the full magnetohydrodynamic equations can be applied (Davidson 2001). We define the oscillation period

$$T = \frac{1}{f} = \frac{2\pi}{\omega} \tag{6}$$

as time scale and the maximal velocity

$$U = \omega A \tag{7}$$

as velocity scale based on the amplitude A of the oscillation. Introducing further B, ρU^2, and LUB as scales for the magnetic field, pressure (ρ is the mass density of the fluid), and electric potential, respectively, the full set of non-dimensional equations reads

$$\nabla \cdot \mathbf{u} = \nabla \cdot \mathbf{j} = 0, \tag{8}$$

$$\frac{1}{\mathrm{KC}} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \frac{1}{\mathrm{Re}} \nabla^2 \mathbf{u} + \frac{\mathrm{Ha}^2}{\mathrm{Re}} (\mathbf{j} \times \mathbf{B}), \tag{9}$$

$$\mathbf{j} = -\nabla \phi + \big((\mathbf{u} - \mathbf{u}_m) \times \mathbf{B} \big), \tag{10}$$

$$\nabla^2 \phi = \nabla \cdot \big((\mathbf{u} - \mathbf{u}_m) \times \mathbf{B} \big), \tag{11}$$

where

$$\mathbf{x}_m(t) = x_m(t)\mathbf{e}_x, \quad x_m(t) = x_{m,0} + \frac{\mathrm{KC}}{2\pi} \sin(2\pi t), \tag{12}$$

$$\mathbf{u}_m(t) = u_m(t)\mathbf{e}_x, \quad u_m(t) = \cos(2\pi t). \tag{13}$$

Equations (9) are the incompressible Navier–Stokes equations for the three-dimensional velocity vector field, which is generated by the Lorentz force $\mathbf{j} \times \mathbf{B}$. Equation (10) is Ohm's law for a moving conductor with the induced electric field represented by the gradient of the scalar electric potential in accordance with the quasistatic approximation. The condition (11) ensures that the current density field is solenoidal. The quantity p denotes the pressure field and ϕ the electric potential.

The quantities $\mathbf{x}_m(t)$ and $\mathbf{u}_m(t)$ denote the position and the velocity of the permanent magnet at time t, respectively. The non-uniform magnetic field $\mathbf{B}$ is computed using an analytical expression presented in Furlani (2001). Note that the induction [i.e., Eqs. (10) and (11)] results from the relative motion between the fluid and the permanent magnet. More details on the derivation can be found in Prinz et al. (2016). In Eq. (9), the following three non-dimensional parameters

$$\mathrm{Re} = \frac{UL}{\nu}, \quad \mathrm{KC} = \frac{2\pi A}{L}, \quad \mathrm{Ha} = BL\sqrt{\frac{\sigma}{\rho\nu}} \tag{14}$$

occur. These are the Reynolds number Re, the Keulegan–Carpenter number KC, and the Hartmann number Ha. The quantities ν and σ in Eq. (14) are the kinematic viscosity and the electrical conductivity. A further useful parameter is the interaction parameter N (also called Stuart number). It characterizes the strength of the Lorentz force relative to inertial forces and is defined by

$$N = \frac{\mathrm{Ha}^2}{\mathrm{Re}} = \frac{B^2 L\sigma}{\rho U}. \tag{15}$$

In addition to the magnet motion one has to specify its duration. One could take a fixed number of cycles but this would imply a change of the duration with the frequency for a given liquid and vessel geometry. We have therefore decided to limit the duration of the stirring to 10% of the viscous diffusion time, i.e., $0.1\ L^2/\nu$, which does not depend on the frequency. Based on the period T, the non-dimensional duration is

$$T_{\max} = \frac{0.1L^2}{T\nu} = 0.1\beta, \tag{16}$$

where

$$\beta = \frac{\mathrm{Re}}{\mathrm{KC}} = \frac{fL^2}{\nu} \tag{17}$$

denotes the frequency parameter (Troesch and Kim 1991). It is also useful later on to define time in units of the duration, i.e.

$$\tilde{t} = \frac{t}{T_{\max}}. \tag{18}$$

Here, the tilde indicates proportionality to the viscous scale L^2/ν. The conversion of the physical model into a dimensionless form is a typical procedure in continuum mechanics and dynamical systems theory. In this way one reduces the number of parameters such as viscosity, conductivity, oscillation frequency, length, magnetic field strength to the essential minimum, i.e. one retains those whose effect cannot be represented by a scaling factor. Note that the non-dimensional parameters typically represent the ratio of competing physical effects, e.g. inertia forces to friction forces for the Reynolds number Re.

The first objective of the optimization is to generate a well-mixed particle distribution. To quantify the mixing process, we introduce a mixing norm ξ according to

$$\xi = \frac{1}{M}\sqrt{\sum_{i=0}^{M-1}\left(\frac{N_p}{M} - N_i\right)^2}. \tag{19}$$

This mixing norm should be as small as possible for a good mixing quality, hence we minimize it. For other norms in a Brownian particle cloud, we refer to Okabe et al. (2008). To maintain a large particle to cell ratio, we compute ξ on an equidistant mixing grid consisting of $6 \times 6 \times 3$ rectangular cells in the x, y, and z-direction, respectively. In Eq. (19), M denotes the total number of cells (here $M = 108$), N_i denotes the number of particles in cell i of the mixing grid, and N_p is the total number of Lagrangian particles (here $N_p = 2500$).

The second objective function is the work $W = \int \mathbf{F} \cdot \mathrm{d}\mathbf{r}$ done by the permanent magnet on the flow over the duration of the magnet motion. The force is the total Lorentz force on the flow. Non-dimensionalization of W needs to be done with care since time and velocity scales depend on the magnet motion. For a comparison between different frequencies and amplitudes the reference unit for W should be independent of A and f. Since the magnet displacement is only in x, $\mathrm{d}\mathbf{r} = \mathbf{e}_x u_m \mathrm{KC}\,\mathrm{d}t$. We therefore obtain the non-dimensional expression

$$W = \mathrm{Ha}^2 \mathrm{Re}\,\mathrm{KC} \int_0^{T_{max}} u_m(t) F_x \,\mathrm{d}t, \tag{20}$$

for the work in units $\rho L v^2$, where

$$F_x = \int_{-1/2}^{1/2} \int_{-L_y/2}^{L_y/2} \int_{-L_x/2}^{L_x/2} (\mathbf{j} \times \mathbf{B})_x \,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z \tag{21}$$

is the integral of the x-component of the non-dimensional Lorentz force density.

4.2 Numerical method for objective function evaluation

The governing equations are solved by a code which is adapted from Krasnov et al. (2011). The code was extensively used to perform DNS and LES of turbulent magnetohydrodynamic shear flows in various setups (Zikanov et al. 2014; Krasnov et al. 2012; Prinz et al. 2018; Prinz et al. 2019). The equations are discretized on a structured mesh by a second-order finite-volume scheme within a collocated variable arrangement following the definitions of Morinishi et al. (1998). The incompressibility condition is incorporated by a standard projection method. The elliptic problems for pressure p and electric potential ϕ are solved by adapting the FishPack libraries (Adams et al. 1999). To adequately resolve the thin magnetohydrodynamic boundary layers, the computational mesh can be refined in vertical (z) direction by using a coordinate transformation based on the hyperbolic

tangent, which transforms the uniform coordinate η to the non-uniform coordinate z, i.e.

$$z(\eta) = \frac{\tanh(\alpha\eta)}{\tanh(\alpha)}, \tag{22}$$

where α is a constant that determines the grid-clustering. To avoid particle communication, the parallelization is based on shared-memory parallelization (OpenMP) solely. The particles are advected with a second-order Euler scheme; trilinear interpolation is used to obtain the velocity field at the Lagrangian coordinates.

All simulations are conducted on a mesh with 64^2 equidistantly spaced grid points in both horizontal directions (i.e., in the x and y-direction), and 32 grid points resolve the vertical direction while α from Eq. (22) is set to 1.0 to better resolve the Hartmann boundary layer. Furthermore, all simulations are initiated from a quiescent state.

4.3 General structure of the flow

For a better understanding of the mixing process, it is useful to consider some characteristics of the obtained laminar flows. We conducted three series of simulations where only one parameter is varied. The remaining parameters are kept constant at values that fall into the range of the optimization problem, as described below. The parameters are

- $\beta = 100 \ldots 1000$, KC $= \pi$, and Ha $= 35$,
- $\beta = 500$, KC $= 1 \ldots 5$, and Ha $= 35$, as well as
- $\beta = 200$, KC $= 3$, and Ha $= 10 \ldots 40$.

The flow is evaluated after 50 periods.

Figure 3 shows the dependence of the integral kinetic energy (one half of $\langle E_k \rangle_T$, where $E_k = \frac{1}{V}\int \left(u_x^2 + u_y^2 + u_z^2\right) \mathrm{d}V$ and $\langle \cdot \rangle_T$ denotes averaging over one period) on the varied quantities. Figure 3a shows a decrease of the kinetic energy with

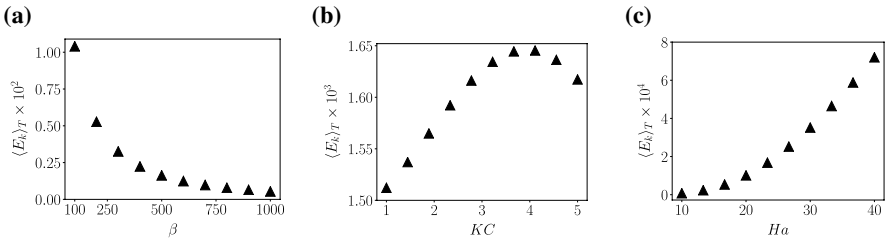

Fig. 3 Dependence of the kinetic energy on the variables of the optimization problem—**a** dependence on β for KC $= \pi$ and Ha $= 35$, **b** dependence on KC for $\beta = 500$ and Ha $= 35$, and **c** dependence on Ha for KC $= 3$ and $\beta = 200$; $\langle \cdot \rangle_T$ denotes averaging over one period.

increasing β. This behavior is, in general, similar to an oscillating boundary layer in the hydrodynamic case and resembles the reduced penetration depth for increasing frequencies. Furthermore, the Stuart number N, i.e., the ratio between Lorentz to inertial forces, decreases as β increases when KC and Ha remain fixed.

Figure 3b shows that, for the studied parameters, the kinetic energy first increases almost linearly with increasing KC, reaches a maximum for KC ≈ 4, and decreases for KC $\geq\approx 4$. The non-monotonic on KC is a geometrical effect, i.e., the side-walls of the computational domain affect the evolution of the flow. To ensure that this effect is not due to a change in the flow structure, we computed the same parameters on a larger domain (not shown) resulting in a purely monotonic increase of $\langle E_k \rangle_T$ for KC $= 1 \ldots 5$. Figure 3c shows a monotonic increase of $\langle E_k \rangle_T$ for increasing Ha.

For the chosen parameters, the instantaneous flow structure behaves similarly to that described by the two-dimensional numerical simulations of Beltrán et al. (2010), i.e., the flow is characterized by a symmetric pair of vortices changing sign in each half cycle. A more detailed description of the flow and an analysis of its mixing properties by concepts of chaotic advection (Aref 2017) is beyond the scope of the present manuscript.

However, an important mechanism, not shown in Beltrán et al. (2010), is the presence of the steady-streaming motion revealed by time-averaging the three-dimensional flow field. From classical hydrodynamics, it is known that the steady-streaming motion provides an efficient mechanism for mixing processes (Riley 2001; Kumar et al. 2011). For the present flows, a steady streaming motion is detected that shows substantial similarity to the streaming motion reported for an oscillating sphere (Otto et al. 2008).

Figure 4 shows the steady-streaming motion visualized by three-dimensional streamlines for $\beta = 100$, KC $= \pi$, and Ha $= 40$, viewed from the y-direction (Fig. 4a), and viewed from the x-direction (Fig. 4b). For this, the velocity field is time-averaged over one period after transitional effects vanished. The three-dimensional flow motion consists of a pair of vortices, mainly aligned in the

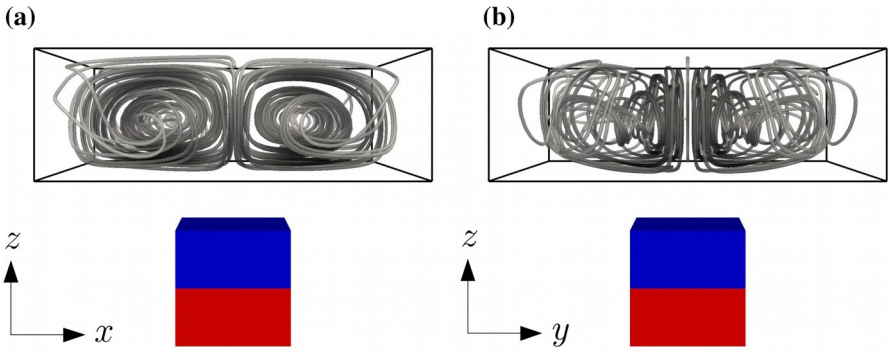

Fig. 4 Steady streaming motion visualized by three-dimensional streamlines for $\beta = 100$, KC $= \pi$, and Ha $= 40$—**a** viewed from the y-direction; **b** viewed from the x-direction

y-direction. The streaming motion affects the complete flow domain and, as shown later, appears to play a significant role in the spreading of the Lagrangian particles.

5 Results from the optimization

In the following we collect the results of the numerical experiments on the optimization problem as well as the results of the new numerical optimization approach described in Sect. 3.

5.1 Preliminary study with two control parameters

To show that the optimization problem is well-defined, we performed a test where two variables, β and KC, are varied while the Hartmann number is kept constant at Ha = 30. The simulations are performed using the same number of grid points as in Sect. 4.3. Since only two variables are considered, the problem can be addressed and visualized by a systematic analysis of the parameter space. Both variables, β and KC, are chosen to match the boundaries that are also used in the optimization problem in the next subsection, i.e., $100 \leq \beta \leq 1000$ and $1 \leq \mathrm{KC} \leq 5$. This parameter space is discretized by a 10×10 grid. For simplicity, the results of the objective functions are normalized by their maximal values over the grid, i.e., $\xi^* = \xi/\xi_{max}$ and $W^* = W/W_{max}$, where $\xi_{max} = 74.95$ and $W_{max} = 766.64$.

Figure 5 shows contours of the normalized objective functions. Both objective functions are competing, i.e., the mixing norm decreases for parameters where the work done by the permanent magnet increases.

Figure 6a shows the approximated efficient solutions. They were found by plotting the values ξ^* and W^* for all grid points and retaining a set that visually corresponds to a smooth lower boundary.

Figure 6b shows the values of both objective functions plotted against the Stuart number N. The intersection between both functions happens for a Stuart number of

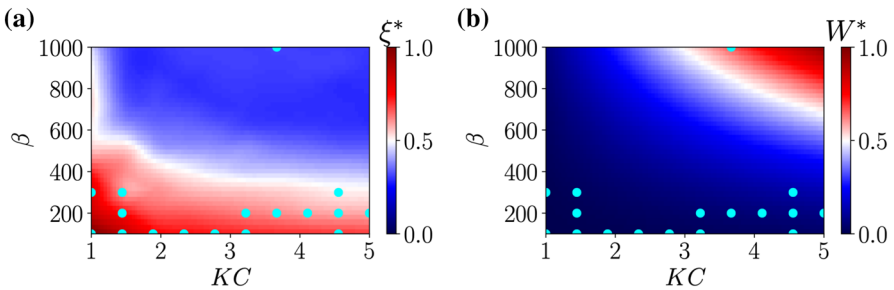

Fig. 5 Normalized objective functions in a two-dimensional discretized parameter plane—**a** the mixing norm (Eq. 19), and **b** the work done by the permanent magnet (Eq. 20). Note that the visualized contours are interpolated. Pareto-optimal points are added as filled circles to both panels

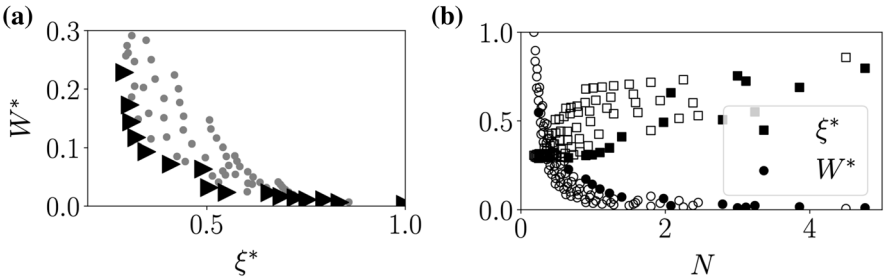

Fig. 6 Normalized objective functions in a two-dimensional discretized parameter plane. **a** Extracted efficient solutions in the image space are plotted as filled triangles. All other simulations as smaller circles. **b** The values of both objective functions plotted against the Stuart number N. Again we add all other simulations as open circles and squares in correspondance with the legend

Table 2 Values for selected feasible points of the optimization problem (MOP); rows 2–4 show the variables of the optimization **p**; row 5 states the Reynolds number Re and row 6 the interaction parameter N. Values of the objective functions (normalized by the maximal value that was obtained in the optimization run) are given in rows 7 and 8

	Starting pt. $\mathbf{p}^{(0)}$	Solution $\mathbf{p}^{S}$	Example $\mathbf{p}^{E}$	Lowest ξ^*	Lowest W^*
β	710.00	375.78	1000.00	392.14	146.08
KC	3.00	3.54	5.00	4.42	1.00
Ha	40.00	37.33	15.88	39.98	22.07
Re	2130.00	1330.52	5000.00	1731.57	146.08
N	0.75	1.05	0.05	0.92	3.33
W^*	0.2505	0.1505	0.1648	0.2758	0.0017
ξ^*	0.2504	0.2471	0.5571	0.1937	0.9928

about unity. For $N \geq 1$, the work required to move the permanent magnet rapidly decreases, while the mixing norm increases.

Overall, the results from the two-dimensional discretized parameter plane show that the present problem is well suited for a multivariate optimization study, as conducted for three variables in the next section.

5.2 Results for three parameters

In addition to the frequency parameter β and the amplitude parameter KC, we now consider the strength of the magnetic field represented by the Hartmann number Ha as a control parameter.

For the optimization, we study an experiment where the geometry and the fluid properties are constant and frequency, amplitude and magnet strength can be varied as stated in Sect. 2. Hence, the corresponding non-dimensional quantities β, KC and Ha are the variables for the optimization problem. The two objective functions are ξ and W.

We performed optimization runs with different starting points. We did this for experimental reasons to learn about different outputs of our algorithm. The algorithm results each time in a single approximated efficient solution. Thereby, several runs resulted in –from a physical perspective– unsatisfactory solutions, such as very large ξ and small W (or vice versa). Albeit mathematically correct, these solutions do not allow meaningful physical interpretations. Hence, we only analyze the result of one optimization run for the starting point $\mathbf{p}^{(0)} = (710, 3, 40)$, see Table 2.

The choice of the starting point is motivated from the practical engineering perspective where a certain range for each of the three components of $\mathbf{p}^{(0)}$ is possible in a concrete setup. $p_1^{(0)} = \beta$, the frequency parameter, and $p_2^{(0)} = \mathrm{KC}$, the oscillation amplitude parameter, were chosen in the middle of the possible range sufficiently far away from the interval bounds. As the mixing is driven by an external magnetic field, the start with a stronger field is also reasonable. Note that there is no rigorous mathematical criterion that guided us here, but a reasonable procedure in the real-life application. For $\mathbf{p}^{(0)}$, the optimization algorithm called in a total of 62 simulations lasting in total 3477.6 hours of single processor CPU time. Similar or even longer computing times were required for the other runs that resulted in unsatisfactory solutions. All simulations were conducted on the computing cluster facility of TU Ilmenau.

Figure 7 shows the result of applying the proposed algorithm to the multi-objective optimization problem. Again, the values of the objective functions are normalized by their maximal values that were obtained within the optimization run, which are 71.99 for ξ_{max} and 1.32×10^3 for W_{max}. Each black dot represents an individual simulation, i.e., an evaluation of the objective functions, for instance, to build a model of the objectives. Hence it can be seen that already for just three optimization

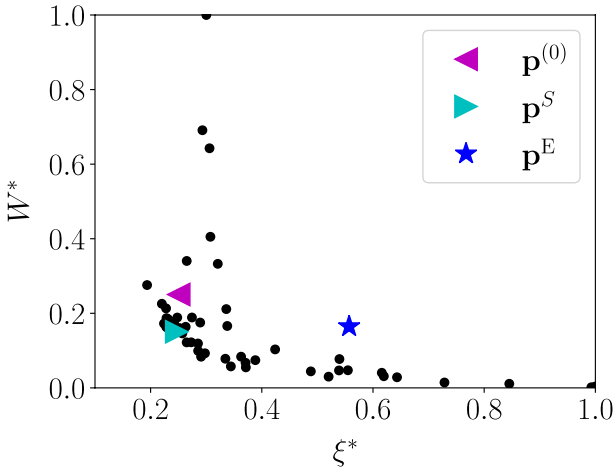

Fig. 7 Results of the optimization run—image space; $\mathbf{p}^{(0)}$ denotes the starting point, $\mathbf{p}^{\mathrm{S}}$ denotes the solution of the algorithm, and $\mathbf{p}^{\mathrm{E}}$ denotes a example point. The black dots are images of points to which the functions have been evaluated to built the models. (Color figure online)

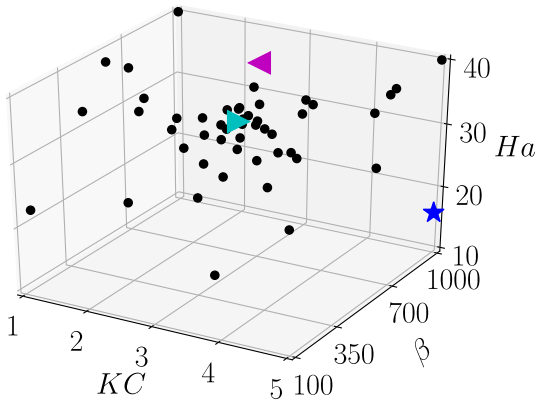

Fig. 8 Results of the optimization run—domain space. $\mathbf{p}^{(0)}$ denotes the starting point, $\mathbf{p}^{\mathrm{S}}$ denotes the solution of the algorithm, and $\mathbf{p}^{\mathrm{E}}$ denotes an example point. The black dots correspond to those in Fig. 7. (Color figure online)

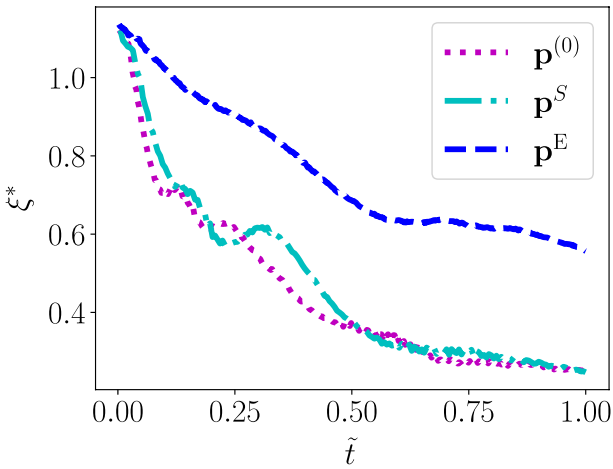

Fig. 9 Temporal development of the mixing norm ξ^* during the flow simulation for three different parameter vectors $\mathbf{p}$, namely the starting point of the optimization $\mathbf{p}^{(0)}$, the solution $\mathbf{p}^{\mathrm{S}}$ of the optimization algorithm, and the example point $\mathbf{p}^{\mathrm{E}}$

variables many function evaluations are necessary, even while we are using models of the functions. The starting point $\mathbf{p}^{(0)}$, the solution of the algorithm $\mathbf{p}^{\mathrm{S}}$, as well as an example $\mathbf{p}^{\mathrm{E}}$ that will be used for the further discussion are highlighted.

The optimization run finds a set of variables that reduces the work done by the magnet by about 40% and improves the mixing efficiency slightly by about 1% compared to the values for the starting point $\mathbf{p}^{(0)}$. Thus both objective function values have been improved by the algorithm, which is a direct implication of the new acceptance test. All variables are significantly altered from $\mathbf{p}^{(0)} = (710, 3, 40)$ to $\mathbf{p}^{\mathrm{S}} = (375.78, 3.54, 37.33)$, see Table 2. The domain space is shown in Fig. 8.

The example point $\mathbf{p}^{\mathrm{E}}$ demonstrates the complexity since roughly the same amount of work required to move the permanent magnet results in very different mixing efficiencies. Consistently with the previous subsection, the optimization

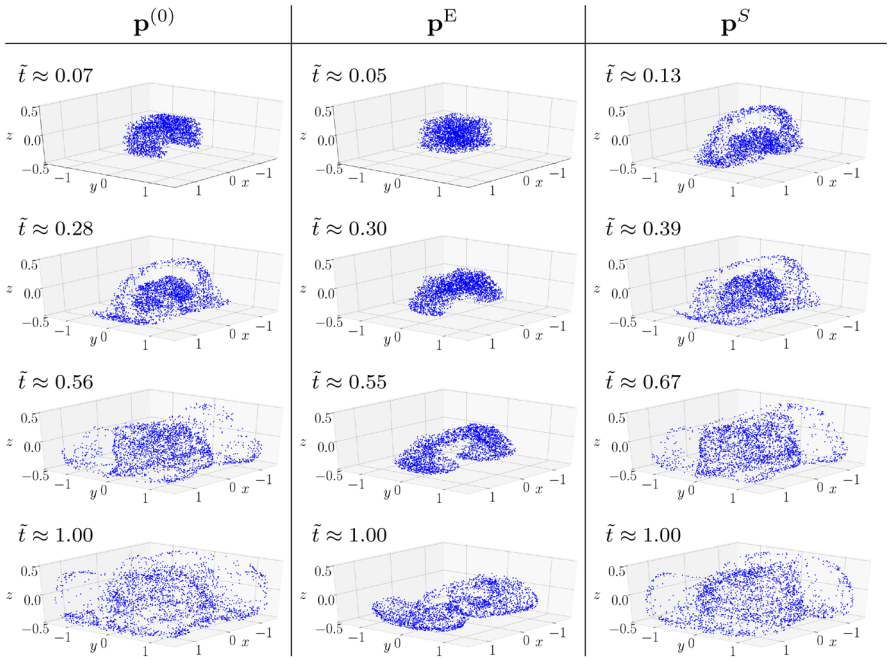

Fig. 10 Results of the optimization run—temporal development of the particle distribution. $\mathbf{p}^{(0)}$ denotes the starting point, $\mathbf{p}^{\mathrm{S}}$ denotes the solution of the algorithm, and $\mathbf{p}^{\mathrm{E}}$ denotes the example point

run tends towards values of N about unity to provide a good mixing efficiency and reasonable work required to move the permanent magnet.

Further insight into the mixing processes is provided in Figs. 9 and 10, where time-dependent quantities are plotted. Figure 9 shows the temporal development of ξ^* for three selected cases, i.e., the starting point $\mathbf{p}^{(0)}$, the solution obtained by the algorithm $\mathbf{p}^{\mathrm{S}}$, and the example point $\mathbf{p}^{\mathrm{E}}$. Starting from $\tilde{t} = 0$, ξ^* decreases in all cases. For the mixing efficiency, the evolution between the starting point $\mathbf{p}^{(0)}$ and the solution $\mathbf{p}^{S}$ is similar. However, it should be kept in mind that this concerns only one of the two objective functions.

The three-dimensional particle distributions in Fig. 10 show that the particles mainly follow the streaming motion. When the mixing process is initiated, the particles follow the vortex pair aligned within the xz-plane and get transported towards the side-walls in x-direction. This is readily true for $\mathbf{p}^{(0)}$ and $\mathbf{p}^{\mathrm{S}}$. For $\mathbf{p}^{\mathrm{E}}$ the high value of β results in a reduced penetration depth of the oscillating fluid motion and, hence, the mixing process only takes places within the lower half space of the domain. At the end of the mixing process, i.e., at $\tilde{t} = 1.0$, the particles are well mixed in the solution $\mathbf{p}^{\mathrm{S}}$ of the algorithm.

6 Conclusion and outlook

We presented results from an optimization study of the mixing process in electrically conducting fluids, where Lorentz forces generate the flow due to an oscillating permanent magnet. We modified a trust-region based mathematical algorithm for multi-objective optimization such that we are able to handle two expensive functions. This makes the algorithm suitable for the presented three-dimensional optimization problem with time-dependent simulation based objective functions. We also adapted the original algorithm in such a way that it guarantees an improvement of the objective function values from a starting guess.

The present series of numerical simulations offer quantitative insights into the mixing processes. Our study serves as a proof-of-concept and may form the basis for further investigations on this problem. It also reveals the complex dependencies on the individual variables. Even for the present problem, which is still very simple, the objective functions vary over a wide range within narrow intervals of the variables. The solution of the optimization algorithm suggests a set of variables that improve the mixing efficiency by 1% and reduce the work done by the permanent magnet by 40%. Most individual simulations that are of interest from an engineering point of view are characterized by interaction parameters N between 0.5 and 1.5.

The present study demonstrates the need for and the viability of advanced optimization methods for studying the mixing process in industrially relevant flows, like the stirring of additives within liquid metal melts during the solidification process. For this, the numerical model requires several refinements, i.e., finer grids, more particles, and larger flow domains, which should be improved in future works. Additional physical effects such as particle weight and drag, free surfaces and even solidification may also have to be added.

From the algorithmic point of view, the examination of further possible acceptance tests is of interest. For instance, the hyper volume, see Zitzler and Thiele (1998), with the starting point as reference point could be used. A hypervolume-based expected improvement was already applied, e.g., in Emmerich et al. (2001) for multiobjective Bayesian global optimization for determining an approximation of the full image set of efficient points in case of costly functions. Next to numerical experiments this new acceptance test would require a careful reevaluation of the convergence proof. In addition, one could examine combinations of methods based on surrogate models for global optimization with a local refinement based on a trust region approach to guarantee Pareto criticality.

Acknowledgements Open Access funding enabled and organized by Projekt DEAL. We acknowledge Dmitry Krasnov for help with the numerical code for the MHD flow. SP and JT were financially supported by the Research Training Group Lorentz Force Velocimetry and Lorentz Force Eddy Current Testing which was funded by the Deutsche Forschungsgemeinschaft under Grant No. GRK 1567.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your

intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Adams JC, Swarztrauber P, Sweet R (1999) Efficient FORTRAN subprograms for the solution of separable elliptic partial differential equations. http://www.cisl.ucar.edu/css/software/fishpack/

Aref H et al (2017) Frontiers of chaotic advection. Rev Mod Phys 89:025007

Audet C, Dennis J (2006) Mesh adaptive direct search algorithms for constrained optimization. SIAM J Optim 17:188–217

Audet C, Savard G, Zgal W (2008) Multiobjective optimization through a series of single-objective formulations. SIAM J Optim 19:188–210

Beltrán A, Ramos E, Cuevas S, Brøns M (2010) Bifurcation analysis in a vortex flow generated by an oscillatory magnetic obstacle. Phys Rev E 81:036309

Ben-David O, Levy A, Mikhailovich B, Azulay A (2014) Magnetohydrodynamic flow excited by rotating permanent magnets in an orthogonal container. Phys Fluids 26(9):097104

Brockhoff D, Eichfelder G, Fonseca C, Hunter S, Rigonie E, Stiglmayr M (2020) Computationally expensive functions and large scale test instances. Dagstuhl report Seminar 20031—Scalability in Multi-objective Optimization

Conn A, Gould N, Toint P (2000) Trust-region methods. SIAM, Philadelphia

Cuevas S, Smolentsev S, Abdou MA (2006) On the flow past a magnetic obstacle. J Fluid Mech 553:227–252

Custodio A, Madeira J, Vaz A, Vicente L (2011) Direct multisearch for multiobjective optimization. SIAM J Optim 21:1109–1140

Davidson PA (2001) Introduction to magnetohydrodynamics. Cambridge University Press, Cambridge

Drummond L, Svaiter BF (2005) A steepest descent method for vector optimization. J Comput Appl Math 175:395–414

Eckert S, Nikrityuk PA, Räbiger D, Eckert K, Gerbeth G (2007) Efficient melt stirring using pulse sequences of a rotating magnetic field: part I. Flow field in a liquid metal column. Metall Mater Trans B 38(6):977–988

Emmerich M, Deutz A, Klinkenberg J (2001) Hypervolume-based expected improvement: monotonicity properties and exact computation. In: 2011 IEEE Congress on Evolutionary Computation (CEC). New Orleans, 5–8 June 2011, pp 2147–2154

Fliege J, Drummond LG, Svaiter B (2009) Newton‘s method for multiobjective optimization. SIAM J Optim 20:602–626

Fliege J, Svaiter BF (2000) Steepest descent methods for multicriteria optimization. Math Methods Oper Res 51:479–494

Furlani EP (2001) Permanent magnet and electromechanical devices: Materials, analysis, and applications. Academic Press, Cambridge

Hunter S, Applegate E, Arora V, Chong B, Cooper K, Rincón-Guevara O, Vivas-Valencia C (2019) An introduction to multiobjective simulation optimization. ACM Trans Model Comput Simul (TOMACS) 29(1):1–36

Knowles J (2006) Parego: a hybrid algorithm with on-line landscape approximation for expensive multi-objective optimization problems. IEEE Trans Evol Comput 10(1):50–66

Krasnov D, Zikanov O, Boeck T (2011) Comparative study of finite difference approaches in simulation of magnetohydrodynamic turbulence at low magnetic Reynolds number. Comput Fluids 50(1):46–59

Krasnov D, Zikanov O, Boeck T (2012) Numerical study of magnetohydrodynamic duct flow at high Reynolds and Hartmann numbers. J Fluid Mech 704:421–446

Kumar H, Tawhai MH, Hoffman EA, Lin CL (2011) Steady streaming: a key mixing mechanism in low-Reynolds-number acinar flows. Phys Fluids 23(4):041902

Lara C, Figueroa A, Cuevas S (2017) Nested dipolar vortices driven by electromagnetic forces in a thin liquid metal layer. Magnetohydrodynamics 53(1):97–106

Miettinen K (1999) Nonlinear multiobjective optimization. Kluwer Academic Publishers, Boston

Morinishi Y, Lund T, Vasilyev O, Moin P (1998) Fully conservative higher order finite difference schemes for incompressible flow. J Comput Phys 143(1):90–124

Okabe T, Eckhardt B, Thiffeault JL, Doering CR (2008) Mixing effectiveness depends on the source-sink structure: simulation results. J Stat Mech Theory Exp 2008(07):P07018

Otto F, Riegler EK, Voth GA (2008) Measurements of the steady streaming flow around oscillating spheres using three dimensional particle tracking velocimetry. Phys Fluids 20(9):093304

Ponweiser W, Wagner T, Biermann D, Vincze M (2008) Multiobjective optimization on a limited budget of evaluations using model-assisted *S*-metric selection. In: PPSN: 10th International Conference on Parallel problem solving from nature, 13–17 September 2008, Dortmund, Springer, pp 784–794

Prinz S, Bandaru V, Kolesnikov Y, Krasnov D, Boeck T (2016) Numerical simulations of magnetohydrodynamic flows driven by a moving permanent magnet. Phys Rev Fluids 1(4):043601

Prinz S, Boeck T, Schumacher J (2018) Large eddy simulation of hydrodynamic and magnetohydrodynamic channel flows with a collocated finite-volume scheme and improved subgrid-scale modeling. Eur J Mech-B/Fluids 72:189–198

Prinz S, Boeck T, Schumacher J (2019) Comparison of subgrid-scale models for large eddy simulation of hydrodynamic and magnetohydrodynamic channel flows. Int J Numer Methods Heat Fluid Flow 29:2224–2236

Riley N (2001) Steady streaming. Ann Rev Fluid Mech 33(1):43–65

Rivero M, Cuevas S, Ramos E (2016) Magnetohydrodynamic flow induced by arrays of rotating permanent magnets. Exp Therm Fluid Sci 78:30–40

Ryu J, Kim S (2014) A derivative-free trust-region method for biobjective optimization. SIAM J Optim 24:334–362

Thomann J (2019) A trust region approach for multi-objective heterogeneous optimization. Ph.D. thesis, TU Ilmenau. https://nbn-resolving.org/urn:nbn:de:gbv:ilm1-2019000059

Thomann J, Eichfelder G (2019) Numerical results for the multiobjective trust region algorithm mht. Data Brief 25:104103

Thomann J, Eichfelder G (2019) A trust region algorithm for heterogeneous multi-objective optimization. SIAM J Optim 29:1017–1047

Troesch AW, Kim S (1991) Hydrodynamic forces acting on cylinders oscillating at small amplitudes. J Fluids Struct 5(1):113–126

Villacorta K, Oliveira P, Soubeyran A (2014) A trust-region method for unconstrained multiobjective problems with applications in satisficing processes. J Optim Theory Appl 160:865–889

Zikanov O, Krasnov D, Li Y, Boeck T, Thess A (2014) Patterned turbulence in spatially evolving magnetohydrodynamic duct and pipe flows. Theor Comput Fluid Dyn 28(3):319–334

Zitzler E, Thiele L (1998) Multiobjective optimization using evolutionary algorithms—a comparative case study. In: Eiben AE, Bäck T, Schoenauer M, Schwefel HP (eds) Parallel problem solving from nature—PPSNV. Springer, Berlin, pp 292–301

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.