

# Particle swarm optimisation strategies for IOL formula constant optimisation

Achim Langenbacher<sup>1</sup> | Nóra Szentmáry<sup>2,3</sup> | Alan Cayless<sup>4</sup> | Jascha Wendelstein<sup>1,5</sup> | Peter Hoffmann<sup>6</sup>

<sup>1</sup>Department of Experimental Ophthalmology, Saarland University, Homburg/Saar, Germany

<sup>2</sup>Dr. Rolf M. Schwiete Center for Limbal Stem Cell and Aniridia Research, Saarland University, Homburg/Saar, Germany

<sup>3</sup>Department of Ophthalmology, Semmelweis-University, Budapest, Hungary

<sup>4</sup>School of Physical Sciences, The Open University, Milton Keynes, UK

<sup>5</sup>Department of Ophthalmology, Johannes Kepler University Linz, Linz, Austria

<sup>6</sup>Augen- und Laserklinik Castrop-Rauxel, Castrop-Rauxel, Germany

## Correspondence

Achim Langenbacher, Department of Experimental Ophthalmology, Saarland University, Kirrberger Str 100 Bldg. 22, 66424 Homburg, Germany.  
Email: [achim.langenbacher@uks.eu](mailto:achim.langenbacher@uks.eu)

## Abstract

**Purpose:** To investigate particle swarm optimisation (PSO) as a modern purely data driven non-linear iterative strategy for lens formula constant optimisation in intraocular lens power calculation.

**Methods:** A PSO algorithm was implemented for optimising the root mean squared formula prediction error (rmsPE, defined as achieved refraction minus predicted refraction) for the Castrop formula in a dataset of  $N = 888$  cataractous eyes with implantation of the Hoya Vivinex hydrophobic acrylic aspheric lens. The hyperparameters were set to inertia: 0.8, accelerations  $c1 = c2 = 0.1$ . The algorithm was initialised with  $N_p = 100$  particles having random positions and velocities within the box constraints of the constant triplet parameter space  $C = 0.25$  to  $0.45$ ,  $H = -0.25$  to  $0.25$  and  $R = -0.25$  to  $0.25$ . The performance of the algorithm was compared to classical gradient-based Trust-Region-Reflective and Interior-Point algorithms.

**Results:** The PSO algorithm showed fast and stable convergence after 37 iterations. The rmsPE reduced systematically to 0.3440 diopters (D). With further iterations the scatter of the particle positions in the swarm decreased but without further reduction of rmsPE. The final constant triplet was  $C/H/R = 0.2982/0.2497/0.1435$ . The Trust-Region-Reflective/Interior-Point algorithms showed convergence after 27/17 iterations, respectively, resulting in formula constant triplets  $C/H/R = 0.2982/0.2496/0.1436$  and  $0.2982/0.2495/0.1436$ , both with the same rmsPE as the PSO algorithm (rmsPE = 0.3440 D).

**Conclusion:** The PSO appears to be a powerful adaptive nonlinear iteration algorithm for formula constant optimisation even in formulae with more than 1 constant. It acts independently of an analytical or numerical gradient and is in general able to search for the best solution even with multiple local minima of the target function.

## KEYWORDS

formula constant optimisation, formula prediction error, lens power calculation, nonlinear iterative algorithm, particle swarm optimisation, performance metrics

## 1 | INTRODUCTION

The power of intraocular lenses (IOL) is currently calculated either using empirical formulae, with ‘theoretical-optical formulae’ based on linear Gaussian optics including empirical elements or deep learning structures, or using full aperture raytracing (Hoffer & Savini, 2020; Wendelstein et al., 2021). By inversion of these calculation concepts, the postoperative refraction at the spectacle plane can be predicted if the IOL power (PIOL) is

known. With the first empirical formulae published in the early eighties, the PIOL for emmetropisation is derived from the corneal power (calculated from the corneal radius using a keratometer index of 1.3375), axial length (AL) and a formula constant (A), which adapts the formula to a specific lens design, characteristics of the study population, or to a particular surgical technique (Hoffer & Savini, 2020). In contrast to this simplistic regression formula, several theoretical-optical formulae have been published based on a pseudophakic eye model, in which

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2023 The Authors. *Acta Ophthalmologica* published by John Wiley & Sons Ltd on behalf of Acta Ophthalmologica Scandinavica Foundation.

the IOL power for emmetropisation is based on: the AL, the corneal power calculated from corneal curvature with a keratometer index, other optional biometric measures such as anterior chamber depth (ACD, measured from the corneal epithelium to the front apex of the lens), central thickness of the cornea (CCT) or the crystalline lens (LT), the horizontal corneal diameter (CD), the age or gender of the patient and one or more formula constants which again adapt the generally defined formula to the specific lens design (Aristodemou et al., 2011; Hoffer & Savini, 2020; Schröder et al., 2016; Zhang et al., 2019). The classical calculation concepts use formula constants A (SRK/T formula, Retzlaff et al., 1990), personalised anterior chamber depth pACD (Hoffer Q formula, Hoffer, 1981; Hoffer, 1993; Hoffer, 2007; Hoffer QST formula, Taroni et al., 2023), surgeon factor SF (Holladay 1 formula, Holladay et al., 1988), constant triplets  $a_0/a_1/a_2$  (Haigis formula, Haigis et al., 2000) or constant triplets C/H/R (Castrop formula) (Langenbucher et al., 2022; Langenbucher, Szentmáry, Cayless, Müller, et al., 2021; Wendelstein et al., 2021). All of these theoretical optical formulae are restricted to linear Gaussian optics (paraxial optics) using a pseudophakic model eye with three refractive surfaces (spectacle correction, thin lens cornea and thin lens IOL) or with four refractive surfaces (spectacle correction, thick lens cornea with front and back surface and thin lens IOL). Where the full design data of the IOL and the material refractive index is known for each power, such formulae could also be upgraded to a 5 surface pseudophakic eye model with the IOL considered as thick lens (Langenbucher et al., 2022). The prediction of the axial position of the thin IOL implant (effective lens position, ELP) is always performed empirically using the formula constant (Olsen & Hoffmann, 2014). Over the last 2 decades, most of the newly developed IOL calculation concepts such as the Holladay 2, the Barrett, Kane, T2, K6 or DGS formula have remained undisclosed and unpublished by the formula authors, and therefore the benefits or drawbacks over the classical published formulae cannot be verified in a general case (Wendelstein et al., 2021).

When a new IOL model is launched to the market, formula constants are estimated before launch and then optimised in a validation cycle involving a sufficient number of treated eyes. For that purpose, the biometric measures before cataract surgery, the IOLP of the implanted lens, and the refractive outcome typically 2–6 months after cataract surgery are documented (Langenbucher et al., 2022; Langenbucher, Szentmáry, Cayless, Müller, et al., 2021; Schröder et al., 2016). In simple formulae with only 1 constant, reformulation and solving for the formula constant can generate an 'optimal' individual constant which maps the biometric data and the IOLP to the refractive outcome. If statistical metrics of all the individual constants (such as the mean or median) are used as the optimised constant this does not necessarily produce the best refractive outcome. In addition, this strategy cannot be used in formulae with more than 1 constant (Aristodemou et al., 2011; Langenbucher, Szentmáry, Cayless, Müller, et al., 2021). The classical engineering optimisation method in those situations involves using nonlinear iterative techniques such as Gauss-Newton, Gradient descent,

Levenberg–Marquardt, Trust-Region/Interior Point or direct line search algorithms to extract the set of formula constants (Boyd & Vandenberghe, 2004; Coleman & Li, 1994; Conn et al., 2000; Dikin, 1967; Kanzow et al., 2004; Karmarkar, 1984; Press et al., 2007). The major benefit of such nonlinear optimisation techniques is that – in contrast to the simple formula inversion – it is possible to minimise any target criterion such as the root-mean-squared refractive prediction error (rmsPE, defined as achieved refraction minus predicted refraction), which is the most relevant parameter for both surgeon and the patient. However, most of these nonlinear algorithms require the Jacobian or even the Hessian matrix (Boyd & Vandenberghe, 2004; Dikin, 1967; Kanzow et al., 2004) which might be difficult to derive. Furthermore, dealing with box constraints or linear equality or inequality constraints for the formula constant space might not be supported in those algorithms.

The **purpose of this paper** is to present the implementation and application of the particle swarm optimisation method (PSO, Kennedy & Eberhart, 1995) as a purely data driven nonlinear iterative optimisation strategy not requiring the definition of a Jacobian or Hessian matrix, and able to search for the minimum of the target function within boundaries of our formula constant parameter space. This PSO optimisation has been implemented using the Castrop formula (which has a constant triplet) as a test case. Using a large clinical data set with  $N = 888$  eyes treated with an IOL, the results were compared to two classical optimisation methods in order to investigate the performance in terms of final solution and convergence behaviour.

## 2 | MATERIALS AND METHODS

### 2.1 | Data set for formula constant optimisation

In this retrospective study, we analysed a data set containing measurements from 888 eyes from a cataract population from Augen-und Laserklinik Castrop-Rauxel, Castrop-Rauxel, Germany, which was transferred to us (490 right eyes and 398 left eyes; 495 female and 392 male). The mean age was  $71.2 \pm 9.1$  years (median: 71 years, range: 47–91 years). The local ethics committee (Ärztchamber des Saarlandes, registration number 157/21) provided a waiver for this study, meaning that informed patient consent was not required. The data were transferred to us in an anonymised fashion, precluding back tracing of the patient. The anonymised data contained preoperative biometric data derived with the IOLMaster 700 (Carl-Zeiss-Meditec, Jena, Germany) including AL, CCT, ACD measured from the corneal front apex to the anterior apex of the crystalline lens, LT, corneal front surface radius measured in the flat (R1) and in the steep meridian (R2) and corneal back surface radius in the flat (Rp1) and in the steep (Rp2) meridian. In all cases, a Vivinex 1 piece hydrophobic aspherical (aberration correcting) monofocal intraocular lens (Hoya Surgical Optics, Singapore) was inserted. In addition, the refractive power of the inserted lens PIOL and the

postoperative refraction (sphere and cylinder) 5–12 weeks after cataract surgery were measured by an experienced optometrist and recorded in the data set. The data set included only data with a postoperative Snellen decimal visual acuity of 0.8 (20/25 Snellen lines) or higher, to ensure reliable postoperative refraction. The descriptive data on pre-cataract biometry, PIOL and postoperative refraction are summarised in Table 1.

The anonymised Excel data (.xlsx-format) was imported into MATLAB to Matlab (Matlab 2019b, MathWorks, Natick, USA) for further processing.

## 2.2 | Preprocessing of the data

Custom software was written in Matlab. The patient age was derived from the date of cataract surgery and date of birth. The mean corneal front and back surface radius of curvature  $R_{\text{mean}}$  was calculated as  $R_{\text{m}} = 0.5 \cdot (R_1 + R_2)$  and  $R_{\text{pm}} = 0.5 \cdot (R_{p1} + R_{p2})$ . As an example, we decided to use the Castrop lens power calculation formula. This is a modern IOL calculation concept based on a pseudophakic model eye with four surfaces and three formula constants (a constant triplet with C, H, and R). The formula has been fully disclosed in 2021 by (Langenbucher, Szentmáry, Cayless, Weisensee, et al., 2021; Wendelstein et al., 2021) and uses a sum of segments correction for the AL according to Cooke (Cooke & Cooke, 2019a, 2019b).

## 2.3 | Formula constant optimisation

For the PSO optimisation, the rmsPE was used as the target criterion. An internal function code was programmed to calculate the rmsPE for the input parameters AL, CCT, ACD, LT, Rm and Rpm as function of the formula constants C, H and R. As box constraints we used 0.25/0.25/–0.25 as lower boundaries and 0.45/0.25/0.25 as upper boundaries for C/H/R, respectively. The PSO algorithm was initialised by setting the total number of particles to  $N_p = 100$ , and the positions of the particles in the C/H/R volume were selected to be uniformly distributed within the box constraints using random generators (Kennedy & Eberhart, 1995; Pedersen & Chipperfield, 2010; Shi & Eberhart, 1998). The starting velocities of the particles were chosen to be randomly distributed in the parameter space with the

maximum velocity in C, H and R direction being 5% of the entire parameter ranges in C, H and R (upper minus lower boundaries of the box constraints). The number of iterations was restricted to  $N_{\text{it}} = 100$ , and the acceleration coefficients or trust parameters  $c1$  (personal acceleration coefficient) and  $c2$  (social acceleration coefficient) were both set to 0.1 (Eberhart & Shi, 2000; Mezura-Montes & Coello Coello, 2011; Pedersen, 2010). The  $c1$  expresses how much confidence a particle has in itself, whereas  $c2$  expresses how much confidence a particle has in its neighbours. Both acceleration coefficients  $c1$  and  $c2$ , together with the random vectors  $r1$  and  $r2$ , control the stochastic influence of the cognitive and social components on the overall velocity of the particles in the swarm. The initial inertia was set to  $w = 0.8$  and the damping ratio for the inertia was set to 1.0 (Eberhart & Shi, 2000; Pedersen & Chipperfield, 2010; Wang et al., 2013).

The following section gives a brief description of the most relevant steps in the PSO algorithm:

1. Specify the hyperparameters  $N_{\text{it}}$ ,  $N_p$ ,  $c1$ ,  $c2$ ,  $w$  and boundaries of the box constraints,
2. Generate an initial population: For initialisation in the 0th iteration, we define  $i = 1:N_p$  particles with (uniformly distributed) random positions  $P^i$  within the box constraints volume and with initial particle velocities  $V^i$  at random (random direction and velocity within the velocity constraints),
3. Update position of the particles: For the  $k^{\text{th}}$  iteration, the positions of all  $N_p$  particles ( $k = 0-N_{\text{it}}$ ) are updated to  $P^i(k+1) = P^i(k) + V^i(k+1)$
4. Enforce the boundaries for the position. If  $P^i(k+1)$  is outside a boundaries with one or more vector components, we set this vector component back to this boundary.
5. Update velocity of the particles: At the same time, the velocities of all particles are updated to  $V^i(k+1) = w \cdot V^i(k) + c1 \cdot r1 \cdot (pbest^i - P^i(k)) + c2 \cdot r2 \cdot (gbest - P^i(k))$ , where  $pbest^i$  refers to the best position of particle  $i$  in terms of minimal target function for all iterations from 0 to  $k$  and  $gbest$  refers to the best position for all particles in the neighbourhood (e.g. the entire swarm) in terms of minimal target function.
6. Enforce the boundaries for the velocities: If one or more boundary conditions for the position  $P^i(k+1)$  have been enforced in step 4, the corresponding component or components of the velocity  $V^i(k+1)$  are set

**TABLE 1** Descriptive statistics of the dataset with mean, standard deviation (SD), median, and the lower (quantile 2.5%) and upper (quantile 97.5%) boundary of the 95% confidence interval.

$N = 888$	AL in mm	CCT in mm	ACD in mm	LT in mm	Rm in mm	Rpm in mm	PIOL in diopters	SEQ in diopters
Mean	24.0980	0.5589	3.1864	4.6176	7.7666	6.5316	20.6222	–0.5612
SD	1.40721	0.0361	0.4081	0.4568	0.2682	0.2256	3.7318	0.9236
Median	23.9026	0.5587	3.1848	4.5929	7.7654	6.5307	21.0000	–0.2500
Quantile 2.5%	21.6757	0.4891	2.3720	3.7333	7.2702	6.1143	12.0000	–2.2500
Quantile 97.5%	27.3514	0.6258	3.9435	5.5192	8.3029	6.9824	27.5000	0.5000

Note: AL refers to the axial length, CCT to the central corneal thickness, ACD to the external phakic anterior chamber depth measured from the corneal front apex to the front apex of the crystalline lens, LT to the central thickness of the crystalline lens, Rm and Rpm (mean of the flat and steep meridian each) to the mean radius of curvature for the corneal front and back surface, PIOL to the refractive power of the intraocular lens implant, and SEQ to the spherical equivalent power achieved 5–12 weeks after cataract surgery.

- to 0 to prevent movement of the particle outside the boundary in the next iteration,
7. Evaluate the fitness of the particles: The target function for all particle positions  $P^i(k+1)$  is derived,
  8.  $(pbest^i - P^i(k))$  refers to the vector of the actual position of the  $i$ th particle in the swarm to the best position of this particle explored in the history, and  $(gbest - P^i(k))$  to the vector of the actual position of the  $i$ th particle to the position of the particle which showed the best overall performance in the history,
  9. Update  $pbest^i$  and  $gbest$  after each iteration, update hyperparameters (inertia  $w$ ,  $c1$ ,  $c2$ )
  10. If iteration converged: exit, else: loop back to step 3.

The iteration is stopped when at least one of the following criteria is met: number of iterations reaches  $N_{it}$ , the relative change in the target function rmsPE over the last 5 iterations is less than  $1e-8$  or the standard deviation (SD) of particle swarm positions  $P^i$  is below a threshold of  $1e-4$  for the last 5 iterations for each vector component (Pedersen, 2010; Pedersen & Chipperfield, 2010; Wang et al., 2013).

As one reference, we used the Trust Region Reflective nonlinear iterative optimisation algorithm, which minimises the sum of squared prediction errors in a set of nonlinear equations (equivalent to minimising the rmsPE). This algorithm was implemented as a gradient-based algorithm (the target function included the calculation of the prediction error as well as the gradient/Jacobian of the prediction error with respect to the formula constants  $C$ ,  $H$  and  $R$ ). The maximum number of iterations was again set to  $N_{it} = 100$ , and the optimality tolerance was set to  $1e-8$  (Coleman & Li, 1994; Conn et al., 2000).

As another reference, we used the Interior Point nonlinear iterative optimisation algorithm, which searches for the minimum of a nonlinear target function. The rmsPE calculated over all  $N$  clinical cases was used as the target function. This algorithm was implemented using finite differences (with finite differences step size of  $1e-10$ ) instead of providing the analytical gradient/Jacobian, since the rms operation in the target function is extremely difficult to implement analytically. The maximum number of iterations was again set to  $N_{it} = 100$ , and the optimality tolerance was set to  $1e-8$  (Press et al., 2007).

For both reference algorithms, the same box constraints were used as in the PSO algorithm (Mezura-Montes & Coello Coello, 2011). For the PSO algorithm, the plot of the particle positions together with the particle velocities is shown as initialisation and after 10, 20 and 30 iterations. In addition, the convergence performance is shown using two graphs showing the rmsPE over iteration and the SD of the particle position scatter over iteration for each vector component  $C$ ,  $H$  and  $R$ . For comparison of the performance of the PSO algorithm to the Trust-Region-Reflective (Coleman & Li, 1994; Conn et al., 2000) and the Interior Point algorithm (Press et al., 2007), the final results for the constant triplets together with the rmsPE are provided. In addition, the number of iterations and function evaluations is listed.

### 3 | RESULTS

The effect of the hyperparameters for the PSO on the convergence behaviour was tested in a pilot experiment, where we identified accelerations  $c1 = 0.1$  and  $c2 = 0.1$  together with inertia of  $w = 0.8$  as a good compromise between fast and stable/robust convergence. Several sets of hyperparameters were tested with several positions and velocities for initialisation.

Table 2 shows the convergence performance for the PSO and for the 2 reference algorithms under test. All three algorithms showed fast and stable convergence after a couple of iterations. The PSO required 37 iterations for reaching the lowest value of the target function rmsPE = 0.3440 diopters (D). With the  $N_p = 100$  particles in the swarm, 100 evaluations of the target functions were required for each iteration. The PSO algorithm stopped on the first optimality criterion, which means that the rmsPE did not show a further decrease. The final formula constants were  $C = 0.2982$ ,  $H = 0.2497$  and  $R = 0.1435$ , respectively. Figure 1 shows the situation of the PSO algorithm for the first iteration (initialisation) as well as for the situation after 10, 20 and 30 iterations. In the 3D plot, the positions of the  $N_p = 100$  particles in the particle swarm (blue dot markers) and the actual velocities (red lines) are displayed. The initial position and the best position of each particle (history of all iterations from 1 to  $i$ ) are indicated by green and cyan dot markers, respectively. The respective value of the target function rmsPE as evaluated for the best particle in the swarm at iteration  $i$  is provided in the header of the graphs.

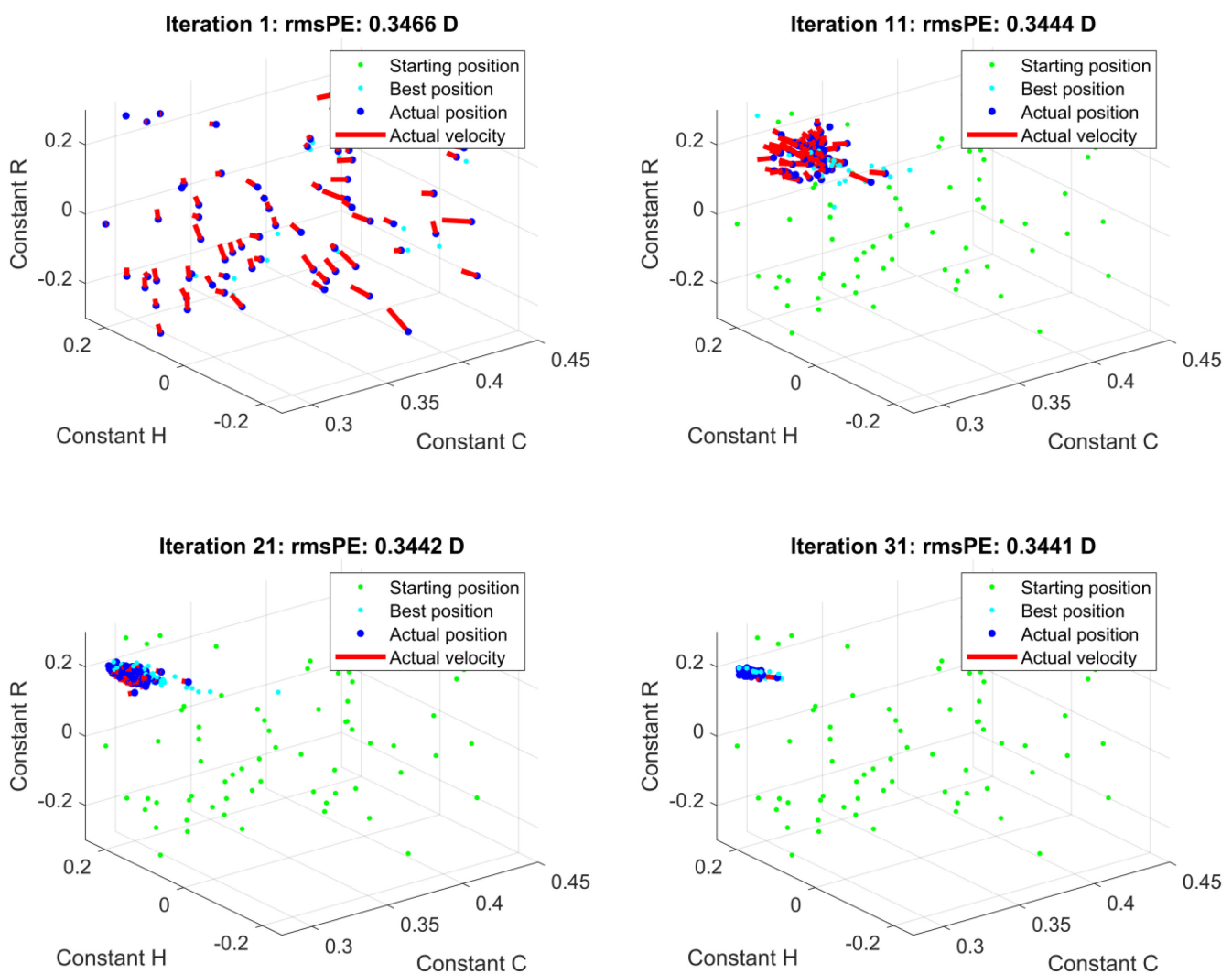
Figure 2 shows the convergence performance of the PSO algorithm. For that purpose, the iterations were continued to the maximum number ( $N_{it} = 100$ ) in spite of one of the stop criteria (first-order optimality) having already been reached after 37 iterations. On the left graph, the target function rmsPE is plotted against the number of iterations. We see from the graph that the target function systematically decreases from the initialisation to iteration 37 and subsequently does not show any further decrease in rmsPE. On the right graph, the standard deviation of the particle scatter is shown for all three components of the parameter space  $C$ ,  $H$  and  $R$ . As can be seen from the graph, the particle concentration increases (standard deviation decreases) for all three components even after reaching the stop criterion (iteration 37), but this further concentration of the particles does not further decrease the rmsPE, meaning that the position of the particle with the best performance in the swarm does not move further.

The Trust-Region-Reflective algorithm (designed to minimise the sum of squared prediction errors in the dataset with  $N = 888$  eyes) and the Interior-Point algorithm (designed to search for the lowest value of the function rmsPE =  $f(C, H, R)$ ) both showed a fast and stable convergence. The Trust-Region-Reflective algorithm stopped after 27 iterations and 28 function evaluations, whereas the Interior-Point algorithm required only 17 iterations. In the Trust-Region-Reflective algorithm for each function evaluation the gradient with respect to  $C$ ,  $H$  and  $R$  was derived algebraically, whereas in the Interior-Point algorithm the gradient was derived numerically by

**TABLE 2** Comparisons of the Particle Swarm Optimisation with the classical Trust-Region-Reflective algorithm (gradient supported) and the Interior-Point algorithm based on finite differences (numerical derivation of the gradient).

$N = 888$	Particle swarm optimisation	Trust-region-reflective	Interior-point
Number of iterations	37	27	17
Number of function evaluations	$37 \cdot NP = 3700$	28	$74 \cdot 7 = 518$
Number of gradient calculations	%	28	%
Iteration stopped for	First order optimality	Step size	First order optimality
Final rmsPE in diopters	0.3440	0.3440	0.3440
Castrop formula constants			
C	0.2982	0.2982	0.2982
H	0.2497	0.2496	0.2495
R	0.1435	0.1436	0.1436

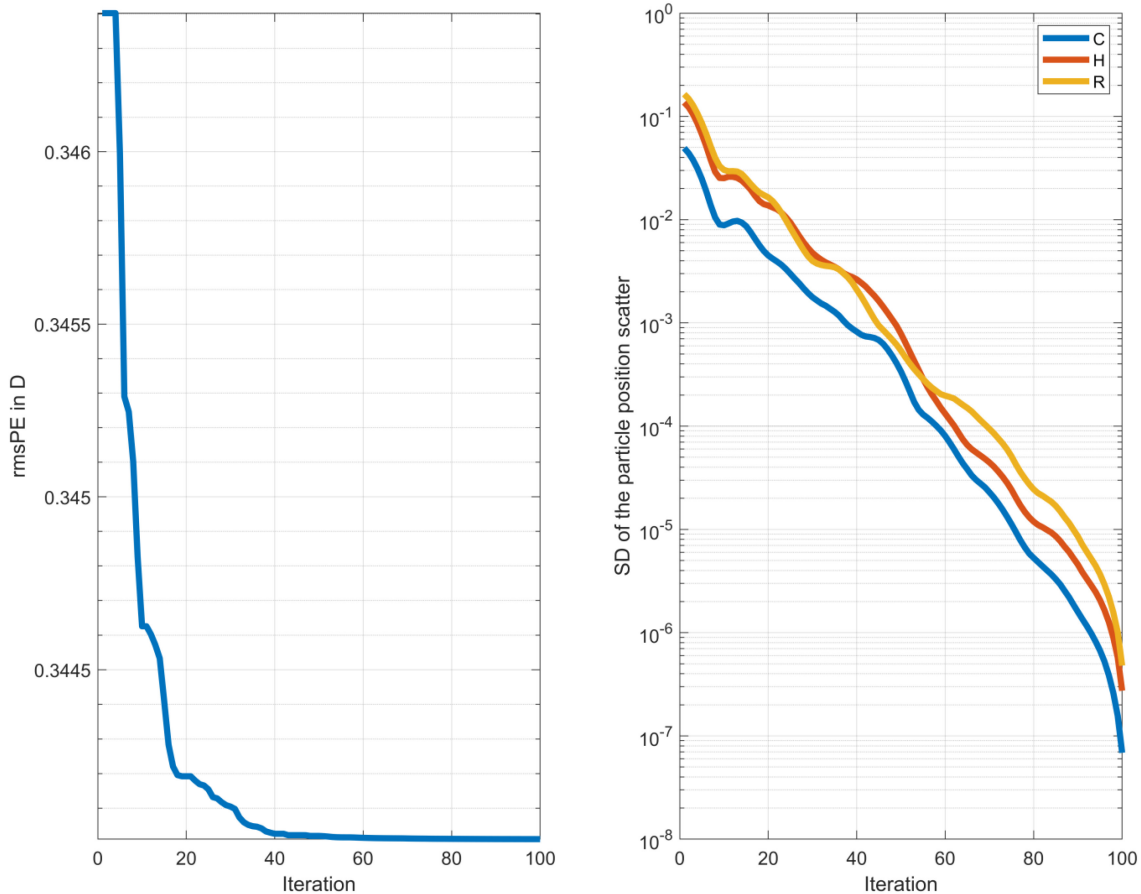
Note: The Particle Swarm optimisation requires the most evaluations of the target function, mostly due to the large number of particles  $N_p = 100$  in the swarm. In contrast, for the Trust-Region-Reflective algorithm for each function evaluation the respective gradient was evaluated with respect to C, H, and R. At the end, both the formula constants and the target function rmsPE showed no noticeable variations between the three algorithms.



**FIGURE 1** 3D plot with the positions of the  $N_p = 100$  particles in the particle swarm (blue dots) and the actual velocities (red lines) for the Particle Swarm Optimisation. The upper left graph refers to the initial state (first iteration), and the upper right, lower left, and lower right graphs to the situation after 10, 20, and 30 iterations respectively. The initial positions and the best position of each particle (history of all iterations from 1 to  $i$ ) are plotted with green dots and cyan dots respectively. The corresponding value of the target function rmsPE (in diopters (D)) evaluated for the best particle in the swarm is provided in the header of the graphs.

evaluating the target function at small variations of C, H, and R (finite differences). The final results of both algorithms used as reference were quite similar ( $C = 0.2982$ ,

$H = 0.2496$ ,  $R = 0.1436$  for the Trust-Region-Reflective versus  $C = 0.2982$ ,  $H = 0.2495$ ,  $R = 0.1436$  for the Interior-Point algorithm). The final iteration values for rmsPE



**FIGURE 2** On the left graph the target function rmsPE (in diopters, (D)) is plotted as a function of iterations. To show the convergence properties of the Particle Swarm Optimisation algorithm, the iteration was continued to the maximum number of iterations ( $N_{it} = 100$ ) even though the stopping criteria were fulfilled after 37 iterations. On the right graph the standard deviation of the particle scatter is shown for all three components of the parameter space C, H, and R. As can be seen from the graph, the particle concentration increases (standard deviation decreases) even after reaching the stopping criterion, but this further concentration of the particles does not change the target function, meaning that the position of the particle with the best performance in the swarm does not move further.

produced by both algorithms were identical to the result from the PSO algorithm (rmsPE = 0.3440 D).

## 4 | DISCUSSION

Classical iterative nonlinear algorithms such as the Gauss-Newton, gradient descent methods, direct line search or combinations are very popular, especially in engineering and modelling (Boyd & Vandenberghe, 2004; Dikin, 1967; Kanzow et al., 2004; Karmarkar, 1984). Where the sum of squared prediction errors is required to be minimised in a system of nonlinear equations, algorithms such as the Levenberg–Marquardt (Kanzow et al., 2004) or the Trust-Region-Reflective algorithms (Conn et al., 2000) are used, as the Gauss-Newton algorithm may show problems with its robustness (Coleman & Li, 1994). For such equation systems the analytical gradient or Jacobian matrix may be provided to support the algorithm in fast convergence. In contrast, if searching for a minimum in a single nonlinear function, algorithms such as the Interior-Point (Press et al., 2007), Active-Set or SQP algorithm may be a good choice. However, here all  $N$  nonlinear equations are condensed into a single equation (e.g.  $\text{rmsPE} = \text{rms}(\text{PE}_1 \dots \text{PE}_N)$ ) with a highly nonlinear operation, and consequently, the analytic gradient of this equation may be very complex. Therefore, the Interior-Point

algorithm (Press et al., 2007) was not implemented using the analytical gradient, but instead by calculating the numerical gradient using finite differences.

In this paper, we implemented a very modern iterative algorithm which was first established in 1995 in artificial intelligence and deep learning applications by (Kennedy & Eberhart, 1995). PSO is a simple and computational efficient technique which remodels the natural phenomenon of swarm behaviour of animals such as birds or fishes where parameters such as velocity and acceleration of the individual particle are affected by the corresponding parameters of the neighbours (Eberhart & Shi, 2000; Pedersen & Chipperfield, 2010; Shi & Eberhart, 1998). In contrast to algorithms such as the Levenberg–Marquardt which are basically unconstrained, the PSO is typically boundary constrained (Mezura-Montes & Coello Coello, 2011) and does not deal with a 1st or 2nd derivative (gradient/Jacobian or Hessian). This self-adaptive algorithm is perfectly suited to large scale optimisation problems and is able to handle noisy data, even in cases with more than one function minimum. The major advantage is that it can be used in situations where the lens power calculation formula is provided only as a ‘Black box’ function with biometric measures and the lens power as input and the predicted refraction as output. It is also memory-based which means that for each particle all the previous positions and target function evaluations are

stored (Pedersen & Chipperfield, 2010). In each iteration, the position of all particles with the best performance over all previous iterations in addition to the particle with the best performance ever in the swarm has to be identified (Eberhart & Shi, 2000; Shi & Eberhart 1998), and this affects the velocity of all the other particles. In this context, the inertia (Wang et al., 2013) refers to the memory of the previous velocity vector (flight direction), the acceleration parameter  $c1$  to the cognitive component which is affected by the position of this particle in previous iterations, and  $c2$  to the social component which is affected by the performance of the neighbour particles (Pedersen, 2010) with  $r1$  and  $r2$  being stochastic vectors. With  $c1 = c2 = 0$  particles keep flying at their current speed until they hit a boundary of the search space (assuming no inertia). If  $c1 = c2$ , particles are attracted towards the average. While most applications use  $c1 = c2$ , the ratio between these constants is problem-dependent. If  $c1 \gg c2$ , each particle is much more attracted to its own personal best position, resulting in excessive wandering. On the other hand, if  $c2 \gg c1$ , particles are more strongly attracted to the global best position, causing particles to rush prematurely towards optima. The inertia parameter  $w$  in the PSO controls the tendency of the particles to keep their previous direction (Wang et al., 2013). For  $w > 1$ , the velocity increases whereas for  $0 < w < 1$  the velocity decreases. Inertia is typically adjusted with each iteration according to a dumping factor (Liu et al., 2017) and the general condition of all particles in the swarm.

From our results, we can see that the PSO performs quite well, but that it requires more iterations compared to both gradient based algorithms used as reference. The benefit of the PSO in this context is that no gradient is required for this algorithm (neither an analytic nor a numerical gradient) and that according to the literature it deals perfectly with noisy data or a target function which has multiple (local) minima (Eberhart & Shi, 2000; Kennedy & Eberhart, 1995; Shi & Eberhart, 1998). We believe that further optimising the hyperparameters could even speed up the convergence behaviour, and that a reduction of the number of particles would significantly improve the calculation efficiency (Pedersen, 2010).

The character of the present paper is mostly a pilot study, which introduces a modern nonlinear iterative and fully data-driven optimisation technique adopted from natural phenomena to the problem of formula constant optimisation, which (to our knowledge) has not been described in the past. We used a large dataset containing all biometric data together with the power of the implanted intraocular lens and the postoperative refraction, to show the applicability of this technique for the example of a modern IOL calculation formula (the Castrop formula). We chose the Castrop formula (Langenbucher, Szentmáry, Cayless, Weisensee, et al., 2021; Wendelstein et al., 2021) as it is one of the more complex paraxial calculation strategies based on a 4 surface pseudophakic model eye and a triplet of formula constants. Whereas in disclosed formulae with one constant the constant optimisation could be performed in a very simplistic manner, optimisation of constant sets (formulae with more than 1 constant) could be a challenge. We have shown (both in this study and in several

previous experiments not reported in this paper) that the PSO algorithm always showed a stable convergence and generates the same formula constants as classical nonlinear iterative optimisation techniques. With the PSO algorithm we ultimately achieved the same performance as with the classical optimisation techniques.


However, our study has some limitations: first, we did not systematically optimise the hyperparameters  $N_{it}$ ,  $N_p$ ,  $c1$ ,  $c2$  and  $w$  for our implementation. Instead, we performed several hundreds of trials with different sets of the hyperparameters and found that the combination of parameters as recommended in some papers on PSO showed a good performance (Pedersen, 2010; Wang et al., 2013). In all of our tests, the algorithm converged much earlier than 100 iterations. We feel that the PSO even performs well with a lower number of particles in the swarm, but this reduction of the swarm size could increase the number of iterations necessary for convergence. And last but not least, we have not yet tested the PSO algorithm with other datasets or other IOL power calculation formulae. Such tests are, however, planned for the upcoming months.

In conclusion, we presented Particle Swarm Optimisation as a modern purely data-driven adaptive iterative nonlinear optimisation strategy which applies the concept of the natural phenomenon of a swarm behaviour (e.g. of birds or fishes) to the problem of formula constant optimisation in cataract surgery. This algorithm starts with multiple particles with positions and velocities at random, and evaluates the target function at all positions of the individual swarm particles for all iterations to adapt the velocities and accelerations of each particle in each iteration. The algorithm showed a proper performance in terms of fast and stable convergence, and the final result was identical to the result of two classical gradient based algorithms used as reference. Further experiments with other datasets and other formulae are required to underline the potential of this algorithm.

## ACKNOWLEDGEMENTS

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors. Open Access funding enabled and organized by Projekt DEAL.

## ORCID

Achim Langenbucher  <https://orcid.org/0000-0001-9175-6177>

Nóra Szentmáry  <https://orcid.org/0000-0001-8019-1481>

Jascha Wendelstein  <https://orcid.org/0000-0003-4145-2559>

## REFERENCES

- Aristodemou, P., Knox Cartwright, N.E., Sparrow, J.M. & Johnston, R.L. (2011) Intraocular lens formula constant optimization and partial coherence interferometry biometry: Refractive outcomes in 8108 eyes after cataract surgery. *Journal of Cataract and Refractive Surgery*, 37(1), 50–62. Available from: <https://doi.org/10.1016/j.jcrs.2010.07.037>
- Boyd, S. & Vandenberghe, L. (2004) *Convex optimization*. Cambridge: Cambridge University Press, p. 143. ISBN 978-0-521-83378-3. MR 2061575.

- Coleman, T.F. & Li, Y. (1994) On the convergence of interior-reflective Newton methods for nonlinear minimization subject to bounds. *Mathematical Programming*, 67, 189–224. Available from: <https://doi.org/10.1007/BF01582221>
- Conn, A.R., Gould, N.I.M. & Toint, P.L. (2000) *Trust-region-methods*. MOS-SIAM Series on Optimization. Philadelphia: Society for Industrial and Applied Mathematics. Available from: <https://doi.org/10.1137/1.978089871985>
- Cooke, D.L. & Cooke, T.L. (2019a) A comparison of two methods to calculate axial length. *Journal of Cataract and Refractive Surgery*, 45(3), 284–292. Available from: <https://doi.org/10.1016/j.jcrs.2018.10.039>
- Cooke, D.L. & Cooke, T.L. (2019b) Approximating sum-of-segments axial length from a traditional optical low-coherence reflectometry measurement. *Journal of Cataract and Refractive Surgery*, 45(3), 351–354. Available from: <https://doi.org/10.1016/j.jcrs.2018.12.026>
- Dikin, I.I. (1967) Iterative solution of problems of linear and quadratic programming. *Doklady Akademii Nauk SSSR*, 174(1), 747–748.
- Eberhart, R.C. & Shi, Y. (2000) Comparing inertia weights and constriction factors in particle swarm optimization. *Proceedings of the 2000 Congress on Evolutionary Computation (CEC '00)*, 1, 84–88. Available from: <https://doi.org/10.1109/CEC.2000.870279>
- Haigis, W., Lege, B., Miller, N. & Schneider, B. (2000) Comparison of immersion ultrasound biometry and partial coherence interferometry for intraocular lens calculation according to Haigis. *Graefes Archive for Clinical and Experimental Ophthalmology*, 238(9), 765–773. Available from: <https://doi.org/10.1007/s004170000188>
- Hoffer, K.J. (1981) Intraocular lens calculation: The problem of the short eye. *Ophthalmic Surgery*, 12(4), 269–272.
- Hoffer, K.J. (1993) The Hoffer Q formula: A comparison of theoretic and regression formulas. *Journal of Cataract and Refractive Surgery*, 19(6), 700–712. Available from: [https://doi.org/10.1016/s0886-3350\(13\)80338-0](https://doi.org/10.1016/s0886-3350(13)80338-0). Erratum. *J Cataract Refract Surg* 1994; 20: 677.
- Hoffer, K.J. (2007) Errors in self-programming the Hoffer Q formula. *Eye (London, England)*, 21(3), 429. Available from: <https://doi.org/10.1038/sj.eye.6702559>
- Hoffer, K.J. & Savini, G. (2020) Update on intraocular lens power calculation study protocols: The better way to design and report clinical trials. *Ophthalmology*, 9, S0161-6420(20)30638-2. Available from: <https://doi.org/10.1016/j.ophtha.2020.07.005>
- Holladay, J.T., Prager, T.C., Chandler, T.Y., Musgrave, K.H., Lewis, J.W. & Ruiz, R.S. (1988) A three-part system for refining intraocular lens power calculations. *Journal of Cataract and Refractive Surgery*, 14(1), 17–24. Available from: [https://doi.org/10.1016/s0886-3350\(88\)80059-2](https://doi.org/10.1016/s0886-3350(88)80059-2)
- Kanzow, C., Yamashita, N. & Fukushima, M. (2004) Levenberg–Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints. *Journal of Computational and Applied Mathematics*, 172(2), 375–397. Available from: <https://doi.org/10.1016/j.cam.2004.02.013>
- Karmarkar, N. (1984) A new polynomial-time algorithm for linear programming. *Combinatorica*, 4, 373–395. Available from: <https://doi.org/10.1145/800057.808695>
- Kennedy, J. & Eberhart, R.C. (1995) Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks*, 4, 1942–1948. Available from: <https://doi.org/10.1109/ICNN.1995.488968>
- Langenbacher, A., Szentmáry, N., Cayless, A., Müller, M., Eppig, T., Schröder, S. et al. (2021) IOL formula constants: Strategies for optimization and defining standards for presenting data. *Ophthalmic Research*, 64(6), 1055–1067. Available from: <https://doi.org/10.1159/000514916>
- Langenbacher, A., Szentmáry, N., Cayless, A., Weisensee, J., Fabian, E., Wendelstein, J. et al. (2021) Considerations on the Castrop formula for calculation of intraocular lens power. *PLoS One*, 16(6), e0252102. Available from: <https://doi.org/10.1371/journal.pone.0252102>
- Langenbacher, A., Szentmáry, N., Cayless, A., Wendelstein, J. & Hoffmann, P. (2022) Strategies for formula constant optimisation for intraocular lens power calculation. *PLoS One*, 17(5), e0267352. Available from: <https://doi.org/10.1371/journal.pone.0267352>
- Liu, M.H.M., Ruili, X.J. & Zhou, W.H. (2017) A damping factor based particle swarm optimization approach. The 9th International Conference on Modelling, Identification and Control (ICMIC 2017), Kunming, China, July 10–12, 2017. Available from: <https://doi.org/10.1109/ICMIC.2017.8321632>
- Mezura-Montes, E. & Coello Coello, C.A. (2011) Constraint-handling in nature-inspired numerical optimization: Past, present and future. *Swarm and Evolutionary Computation*, 1, 173–194. Available from: <https://doi.org/10.1016/j.swevo.2011.10.001>
- Olsen, T. & Hoffmann, P. (2014) C constant: New concept for ray tracing-assisted intraocular lens power calculation. *Journal of Cataract and Refractive Surgery*, 40(5), 764–773. Available from: <https://doi.org/10.1016/j.jcrs.2013.10.037>
- Pedersen, M.E.H. (2010) *Good parameters for particle swarm optimization*. Luxembourg: Hvas Laboratories. Technical report no. HL1001.
- Pedersen, M.E.H. & Chipperfield, A.J. (2010) Simplifying particle swarm optimization. *Applied Soft Computing*, 10(2), 618–628. Available from: <https://doi.org/10.1016/j.asoc.2009.08.029>
- Press, W.H., Teukolsky, S.A., Vetterling, W.T. & Flannery, B.P. (2007) Section 10.11. Linear programming: Interior-point methods. In: *Numerical recipes: The art of scientific computing*, 3rd edition. New York: Cambridge University Press. ISBN: 978-0-521-88068-8.
- Retzlaff, J.A., Sanders, D.R. & Kraff, M.C. (1990) Development of the SRK/T intraocular lens implant power calculation formula. *Journal of cataract and refractive surgery*, 16(3), 333–340. Available from: [https://doi.org/10.1016/s0886-3350\(13\)80705-5](https://doi.org/10.1016/s0886-3350(13)80705-5). Erratum in: *J Cataract Refract Surg* 1990 Jul;16(4):528.
- Schröder, S., Leydolt, C., Menapace, R., Eppig, T. & Langenbacher, A. (2016) Determination of personalized IOL-constants for the Haigis formula under consideration of measurement precision. *PLoS One*, 11(7), e0158988. Available from: <https://doi.org/10.1371/journal.pone.0158988>
- Shi, Y. & Obayyanahatti, B. (1998) A modified particle swarm optimizer. *Proceedings of the IEEE Conference on Evolutionary Computation, ICEC*. 6: 69–73. <https://doi.org/10.1109/ICEC.1998.699146>
- Taroni, L., Hoffer, K.J., Pellegrini, M., Lupardi, E. & Savini, G. (2023) Comparison of the new Hoffer QST with 4 modern accurate formulas. *Journal of Cataract and Refractive Surgery*, 49(4), 114–118. Available from: <https://doi.org/10.1097/j.jcrs.0000000000001126>
- Wang, S., Zhou, F. & Wang, F. (2013) Effect of inertia weight w on PSO-SA algorithm. *International Journal of Online and Biomedical*, 9, 87–91. Available from: <https://doi.org/10.3991/ijoe.v9iS6.2923>
- Wendelstein, J., Hoffmann, P., Hirnschall, N., Fischinger, I.R., Mariacher, S., Wingert, T. et al. (2021) Project hyperopic power prediction: Accuracy of 13 different concepts for intraocular lens calculation in short eyes. *The British Journal of Ophthalmology*, 27, bjophthalmol-2020-318272. Available from: <https://doi.org/10.1136/bjophthalmol-2020-318272>
- Zhang, J.Q., Zou, X.Y., Zheng, D.Y., Chen, W.R., Sun, A. & Luo, L.X. (2019) Effect of lens constants optimization on the accuracy of intraocular lens power calculation formulas for highly myopic eyes. *International Journal of Ophthalmology*, 12(6), 943–948. Available from: <https://doi.org/10.18240/ijo.2019.06.10>

**How to cite this article:** Langenbacher, A., Szentmáry, N., Cayless, A., Wendelstein, J. & Hoffmann, P. (2023) Particle swarm optimisation strategies for IOL formula constant optimisation. *Acta Ophthalmologica*, 101, 775–782. Available from: <https://doi.org/10.1111/aos.15664>