

METHODOLOGY ARTICLE

Open Access



Dynamic Optimization with Particle Swarms (DOPS): a meta-heuristic for parameter estimation in biochemical models

Adithya Sagar^{1†}, Rachel LeCover^{1†}, Christine Shoemaker² and Jeffrey Varner^{1*} 

Abstract

Background: Mathematical modeling is a powerful tool to analyze, and ultimately design biochemical networks. However, the estimation of the parameters that appear in biochemical models is a significant challenge. Parameter estimation typically involves expensive function evaluations and noisy data, making it difficult to quickly obtain optimal solutions. Further, biochemical models often have many local extrema which further complicates parameter estimation. Toward these challenges, we developed Dynamic Optimization with Particle Swarms (DOPS), a novel hybrid meta-heuristic that combined multi-swarm particle swarm optimization with dynamically dimensioned search (DDS). DOPS uses a multi-swarm particle swarm optimization technique to generate candidate solution vectors, the best of which is then greedily updated using dynamically dimensioned search.

Results: We tested DOPS using classic optimization test functions, biochemical benchmark problems and real-world biochemical models. We performed $\mathcal{T} = 25$ trials with $\mathcal{N} = 4000$ function evaluations per trial, and compared the performance of DOPS with other commonly used meta-heuristics such as differential evolution (DE), simulated annealing (SA) and dynamically dimensioned search (DDS). On average, DOPS outperformed other common meta-heuristics on the optimization test functions, benchmark problems and a real-world model of the human coagulation cascade.

Conclusions: DOPS is a promising meta-heuristic approach for the estimation of biochemical model parameters in relatively few function evaluations. DOPS source code is available for download under a MIT license at <http://www.varnerlab.org>.

Keywords: Biochemical engineering, Systems biology, Modeling, Optimization

Background

Mathematical modeling has evolved as a powerful paradigm to analyze, and ultimately design complex biochemical networks [1–5]. Mathematical modeling of biochemical networks is often an iterative process. First, models are formulated from existing biochemical knowledge, and then model parameters are estimated using experimental data [6–8]. Parameter estimation is typically framed as a non-linear optimization problem wherein the residual (or objective function) between experimental

measurements and model simulations is minimized using an optimization strategy [9]. Optimal parameter estimates are then used to predict unseen experimental data. If the validation studies fail, model construction and calibration are repeated iteratively until satisfactory results are obtained. As our biological knowledge increases, model formulation may not be as significant a challenge, but parameter estimation will likely remain difficult.

Parameter estimation is a major challenge to the development of biochemical models. Parameter estimation has been a well studied engineering problem for decades [10–13]. However, the complex dynamics of large biological systems and noisy, often incomplete experimental data sets pose a unique estimation challenge. Often optimization problems involving biological systems are

*Correspondence: jdv27@cornell.edu

[†]Adithya Sagar and Rachel LeCover contributed equally to this work.

¹Robert Fredrick Smith School of Chemical and Biomolecular Engineering, Cornell University, 244 Olin Hall, Ithaca, NY, USA

Full list of author information is available at the end of the article



non-linear and multi-modal i.e., typical models have multiple local minima or maxima [7, 9]. Non-linearity coupled with multi-modality renders local optimization techniques such as pattern search [14], Nelder-Mead simplex methods [15], steepest descent or Levenberg-Marquardt [16] incapable of reliably obtaining globally optimal solutions as these methods often terminate at local minimum. Though deterministic global optimization techniques (for example algorithms based on branch and bound) can handle non-linearity and multi-modality [17, 18], the absence of derivative information, discontinuous objective functions, non-smooth regions or the lack of knowledge about the objective function hampers these techniques.

Meta-heuristics like Genetic Algorithms (GAs) [19], Simulated Annealing (SA) [20], Evolutionary Programming [21] and Differential Evolution (DE) [22–25] have all shown promise on non-linear multi-modal problems [26]. These techniques do not make any assumptions nor do they require, *a priori* information about the structure of the objective function. Meta-heuristics are often very effective at finding globally optimal or near optimal solutions. For example, Mendes et al. used SA to estimate rate constants for the inhibition of HIV proteinase [27], while Modchang et al. used a GA to estimate parameters for a model of G-protein-coupled receptor (GPCR) activity [28]. Parameter estimates obtained using the GA stratified the effectiveness of two G-protein agonists, N⁶-cyclopentyladenosine (CPA) and 5'-N-ethylcarboxamidoadenosine (NECA). Tashkova et al. compared different meta-heuristics for parameter estimation on a dynamic model of endocytosis; DE was the most effective of the approaches tested [29]. Banga and co-workers have also successfully applied scatter-search to estimate model parameters [30–32]. Hybrid approaches, which combine meta-heuristics with local optimization techniques, have also become popular. For example, Egea et al. developed the enhanced scatter search (eSS) method [32], which combined scatter and local search methods, for parameter estimation in biological models [33]. However, despite these successes, a major drawback of most meta-heuristics remains the large number of function evaluations required to explore parameter space. Performing numerous potentially expensive function evaluations is not desirable (and perhaps not feasible) for many types of biochemical models. Alternatively, Tolson and Shoemaker found, using high-dimensional watershed models, that perturbing only a subset of parameters was an effective strategy for estimating parameters in expensive models [34]. Their approach, called Dynamically Dimensioned Search (DDS), is a simple stochastic single-solution heuristic that estimates nearly optimal solutions within a specified maximum number of function (or model) evaluations. Thus, while meta-heuristics are often effective at

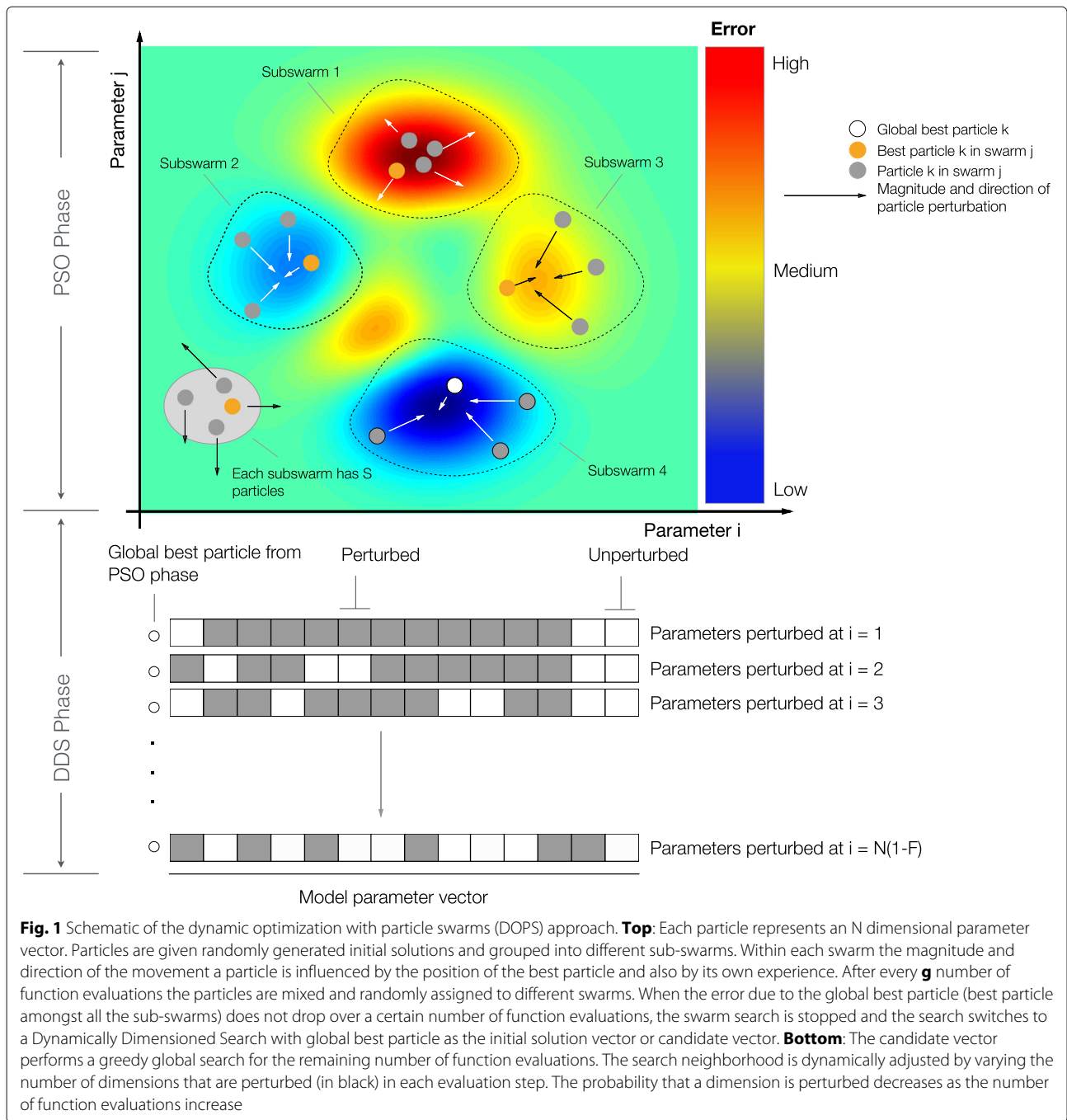
estimating globally optimal or nearly optimal solutions, they require a large number of function evaluations to converge to a solution.

In this study, we developed Dynamic Optimization with Particle Swarms (DOPS), a novel hybrid meta-heuristic that combines the global search capability of multi-swarm particle swarm optimization with the greedy refinement of dynamically dimensioned search (DDS). The objective of DOPS is to obtain near optimal parameter estimates for large biochemical models within a relatively few function evaluations. DOPS uses multi-swarm particle swarm optimization to generate nearly optimal candidate solutions, which are then greedily updated using dynamically dimensioned search. While particle swarm techniques are effective, they have the tendency to become stuck in small local regions and lose swarm diversity, so we combined multi-swarm particle optimization with DDS to escape these local regions and continue towards better solutions [35]. We tested DOPS using a combination of classic optimization test functions, biochemical benchmark problems and real-world biochemical models. First, we tested the performance of DOPS on the Ackley and Rosenbrock functions, and published biochemical benchmark problems. Next, we used DOPS to estimate the parameters of a model of the human coagulation cascade. On average, DOPS outperformed other common meta-heuristics like differential evolution, a genetic algorithm, CMA-ES (Covariance Matrix Adaptation Evolution Strategy), simulated annealing, single-swarm particle swarm optimization, and dynamically dimensioned search on the optimization test functions, benchmark problems and the coagulation model. For example, DOPS recovered the nominal parameters for the benchmark problems using an order of magnitude fewer function evaluations than eSS in all cases. It also produced parameter estimates for the coagulation model that predicted unseen coagulation data sets. Thus, DOPS is a promising hybrid meta-heuristic for the estimation of biochemical model parameters in relatively few function evaluations. However, the relative performance of DOPS should be evaluated cautiously; only naive implementations of the other approaches were tested. Thus, it is possible that other optimized meta-heuristics could outperform DOPS on both test and real-world problems.

Results

DOPS explores parameter space using a combination of global methods.

DOPS combines a multi-swarm particle swarm method with the dynamically dimensioned search approach of Shoemaker and colleagues (Fig. 1). The goal of DOPS is to estimate optimal or near optimal parameter vectors for high-dimensional biological models within a specified number of function evaluations. Toward this objective,



DOPS begins by using a multi-swarm particle swarm search and then dynamically switches, using an adaptive switching criteria, to the DDS approach. The particle swarm search uses multiple sub-swarms wherein the update to each particle (corresponding to a parameter vector estimate) is influenced by the best particle amongst the sub-swarm, and the current globally best particle. Particle updates occur within sub-swarms for a certain number of function evaluations, after which the sub-swarms are reorganized. This sub-swarm mixing is similar

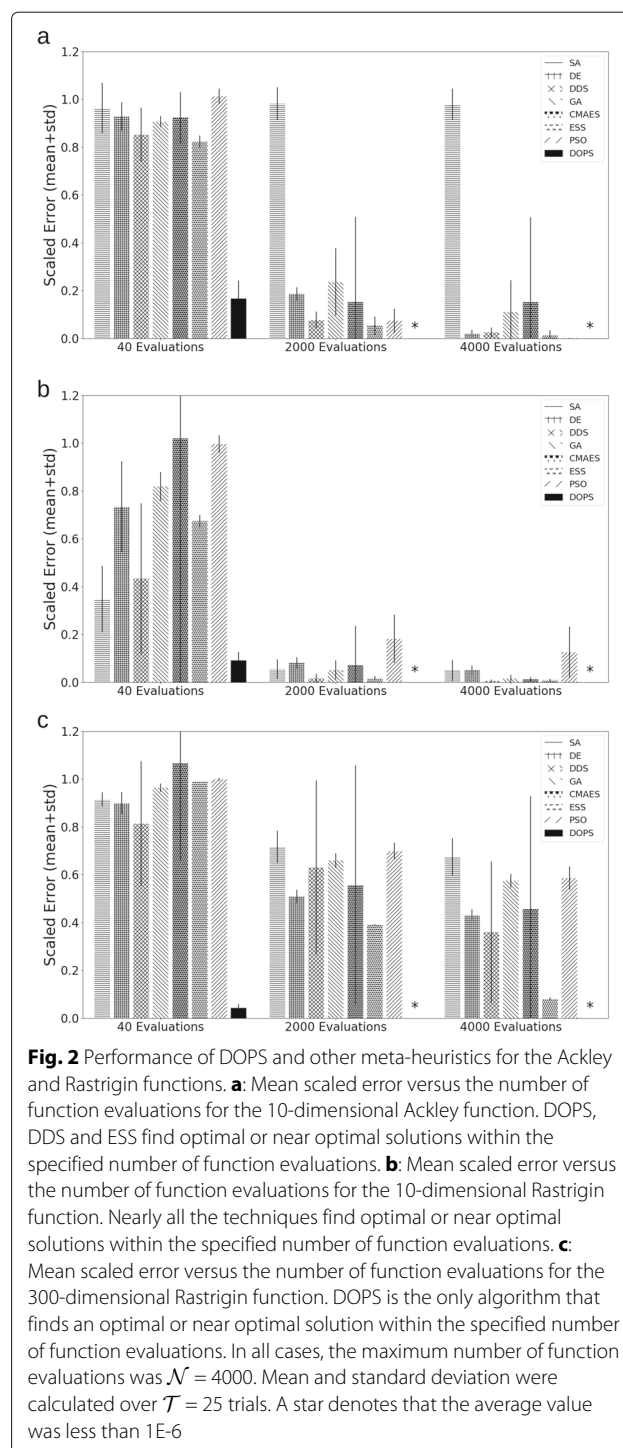
to the regrouping strategy described by Zhao et al. [36]. DOPS switches out of the particle swarm phase based upon an adaptive switching criteria that is a function of the rate of error convergence. If the error represented by the best particle does not decrease for a threshold number of function evaluations, DOPS switches automatically to the DDS search phase. The DDS search is initialized with the globally best particle from the particle swarm phase, thereafter, the particle is greedily updated by perturbing a subset of dimensions for the remaining number

of function evaluations. The identity of the parameters perturbed is chosen randomly, with fewer parameters perturbed the higher the number of function evaluations.

DOPS minimized benchmark problems using fewer function evaluations.

On average, DOPS performed similarly or outperformed four other meta-heuristics for the Ackley and Rastrigin test functions (Fig. 2). The Ackley and Rastrigin functions both have multiple local extrema and attain a global minimum value of zero. In each case, the maximum number of function evaluations was fixed at $\mathcal{N} = 4000$, and $\mathcal{T} = 25$ independent experiments were run with different initial parameter vectors. DOPS found optimal or near optimal solutions for both the 10-dimensional Ackley (Fig. 2a) and Rastrigin (Fig. 2b) functions within the budget of function evaluations. In each of the 10-dimensional cases, other meta-heuristics such as DDS and DE also performed well. However, DOPS consistently outperformed all other approaches tested. This performance difference was more pronounced as the dimension of the search problem increased; for a 300-dimensional Rastrigin function, DOPS was the only approach to find an optimal or near optimal solution within the function evaluation budget (Fig. 2b). Taken together, DOPS performed at least as well as other meta-heuristics on small dimensional test problems, but was especially suited to large dimensional search spaces. Next, we tested DOPS on benchmark biochemical models of varying complexity.

Villaverde and co-workers published a set of benchmark biochemical problems to evaluate parameter estimation methods [33]. They ranked the example problems by computational cost from most to least expensive. We evaluated the performance of DOPS on problems from the least and most expensive categories. The least expensive problem was a metabolic model of Chinese Hamster Ovary (CHO) with 35 metabolites, 32 reactions and 117 parameters [37]. The biochemical reactions were modeled using modular rate laws and generalized Michaelis–Menten kinetics. On the other hand, the expensive problem was a genome scale kinetic model of *Saccharomyces cerevisiae* with 261 reactions, 262 variables and 1759 parameters [38]. In both cases, synthetic time series data generated with known parameter values, was used as training data to estimate the model parameters. For the *Saccharomyces cerevisiae* model, the time series data consisted of 44 observables, while for the CHO metabolism problem the data corresponded to 13 different metabolite measurement sets. The number of function evaluations was fixed at $\mathcal{N} = 4000$, and we trained both models against the synthetic experimental data. DOPS produced good fits to the synthetic data (Additional file 1: Figure S1 and Additional file 2: Figure S2), and recapitulated the nominal parameter values using only $\mathcal{N} \leq 4000$ function



evaluations (Additional file 3: Figure S3). On the other hand, the enhanced scatter search (eSS) with a local optimizer method, took on order 10^5 function evaluations for the same problems. DOPS required a comparable amount of time (Additional file 4: Figure S4), faster convergence (Additional file 5: Figure S5 and Additional file 6: Figure S6), and also had lower variability in the best

amplify this initial coagulation signal. There are several control points in the cascade that inhibit thrombin formation, and eventually terminate thrombin generation. Tissue Factor Pathway Inhibitor (TFPI) inhibits upstream activation events, while antithrombin III (ATIII) neutralizes several of the proteases generated during coagulation, including thrombin. Thrombin itself also inadvertently plays a role in its own inhibition; thrombin, through interaction with thrombomodulin, protein C and endothelial cell protein C receptor (EPCR), converts protein C to activated protein C (APC) which attenuates the coagulation response by proteolytic cleavage of amplification complexes. Termination occurs after either prothrombin is consumed, or thrombin formation is neutralized by inhibitors such as APC or ATIII. Thus, the human coagulation cascade is an ideal test case; coagulation is challenging because it contains both fast and slow dynamics, but also accessible because of the availability of comprehensive data sets for model identification and validation. In this study, we used the coagulation model of Luan et al. [49], which is a coupled system of non-linear ordinary differential equations where biochemical interactions were modeled using mass action kinetics. The Luan model contained 148 parameters and 92 species and has been validated using 21 published experimental datasets.

DOPS estimated the parameters of a human coagulation model for TF/VIIa initiated coagulation without

anticoagulants (Fig. 4). The objective function was an unweighted linear combination of two error functions, representing coagulation initiated with different concentrations of TF/FVIIa (5pM, 5nM) [46]. The number of function evaluations was restricted to $\mathcal{N} = 4000$ for each algorithm we tested, and we performed $\mathcal{T} = 25$ trials of each experiment to collect average performance data (Table 1). DOPS converged faster and had a lower final error compared to the other algorithms (Fig. 5). Within the first 25% of function evaluations, DOPS produced a rapid drop in error followed by a slower but steady decline (Additional file 8: Figure S8b). Approximately between 500-1000 function evaluations DOPS switched to the dynamically dimensioned search phase, however this transition varied from trial to trial since the switch was based upon the local convergence rate. On average, DOPS minimized the coagulation model error to a greater extent than the other meta-heuristics. However, it was unclear if the parameters estimated by DOPS had predictive power on unseen data. To address this question, we used the final parameters estimated by DOPS to simulate data that was not used for training (coagulation initiated with 500pM, 50pM, and 10pM TF/VIIa). The optimal or near optimal parameters obtained by DOPS predicted unseen coagulation datasets (Fig. 6). The normalized standard error for the coagulation predictions was consistent with the training error, with the exception of the 50pM TF/VIIa case which was a factor 2.65 worse (Table 2). However, this

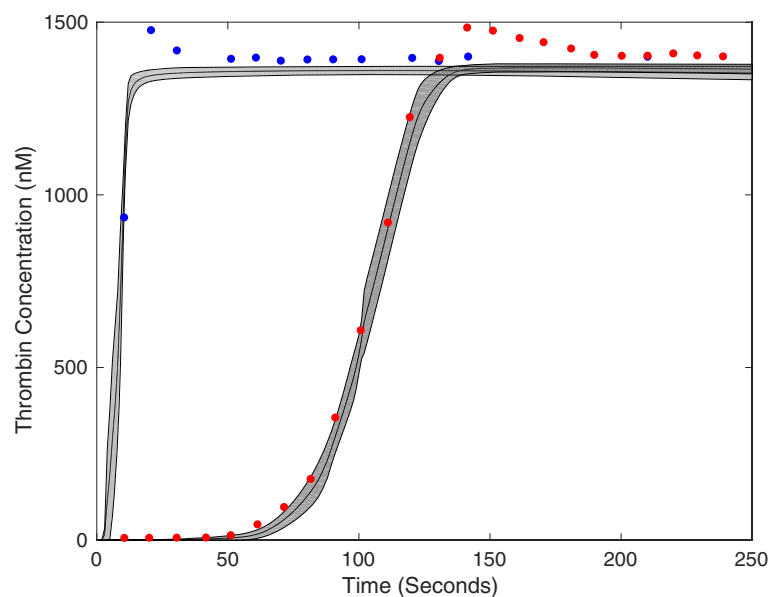


Fig. 4 Model fits on experimental data using DOPS. The model parameters were estimated using DOPS. Solid black lines indicate the simulated mean thrombin concentration using parameter vectors from 25 trials. The grey shaded region represents the 99% confidence estimate of the simulated thrombin concentration. The experimental data is reproduced from the synthetic plasma assays of Mann and co-workers. Thrombin generation is initiated by adding Factor TF/VIIa (5nM (blue) and 5pM (red)) to synthetic plasma containing 200 $\mu\text{mol/L}$ of phospholipid vesicles (PCPS) and a mixture of coagulation factors (II,V,VII,VIII,IX,X and XI) at their mean plasma concentrations

Table 1 Table with optimization settings and results for the coagulation problem, the benchmarks and test functions using DOPS

	Coagulation	B1	B4	Ackley	Rastrigin
Evaluations	4000	4000	4000	4000	4000
Lower Bound	0.001.pnom	0.2.pnom	0.2.pnom	-15	-5.12
Upper Bound	1000.pnom	5.pnom	5.pnom	30	5.12
CPU Time	10.1 hrs	38.3 hrs	6.2 min	2.8 s	2.6 s
Scaled initial error	1.0	1.0	1.0	1.0	1.0
Scaled final error	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01
Scaled nominal error	0.42	0.1	< 0.01	0	0

For each problem the bounds on the parameter vector, the total number of function evaluations, the best initial objective value and the best final objective value are specified. Here *pnom* indicates the nominal or true parameter vector of the model. Nominal objective value represents the objective value using the true parameter vector or the nominal parameter vector. The CPU time is the time taken for the problem on a 2.4GHz Intel Xeon Architecture running Matlab 2014b

might be expected as coagulation initiation with 50pM TF/FVIIa was the farthest away from the training conditions. Taken together, DOPS estimated parameter sets with predictive power on unseen coagulation data using fewer function iterations than other meta-heuristics. Next, we explored how the number of sub-swarms and

the switch to DDS influenced the performance of the approach.

Phase switching was critical to DOPS performance.

A differentiating feature of DOPS is the switch to dynamically dimensioned search following stagnation of the initial particle swarm phase. We quantified the influence of the number of sub-swarms and the switch to DDS on error convergence by comparing DOPS with and without DDS for different numbers of sub-swarms (Fig. 7). We considered multi swarm particle swarm optimization with and without the DDS phase for $\mathcal{N} = 4000$ function evaluations and $\mathcal{T} = 25$ trials on the coagulation model. We used one, two, four, five and eight sub-swarms, with a total of 40 particles divided evenly amongst the swarms. Hence, we did not consider swarm numbers of three and seven. All other algorithm parameters remained the same for all cases. Generally, the higher sub-swarm numbers converged in fewer function evaluations, where the optimum particle partitioning was in the neighborhood of five sub-swarms. However, the difference in convergence rate was qualitatively similar for four, five and eight sub-swarms, suggesting there was an optimal number of particles per swarm beyond which there was no significant advantage. The

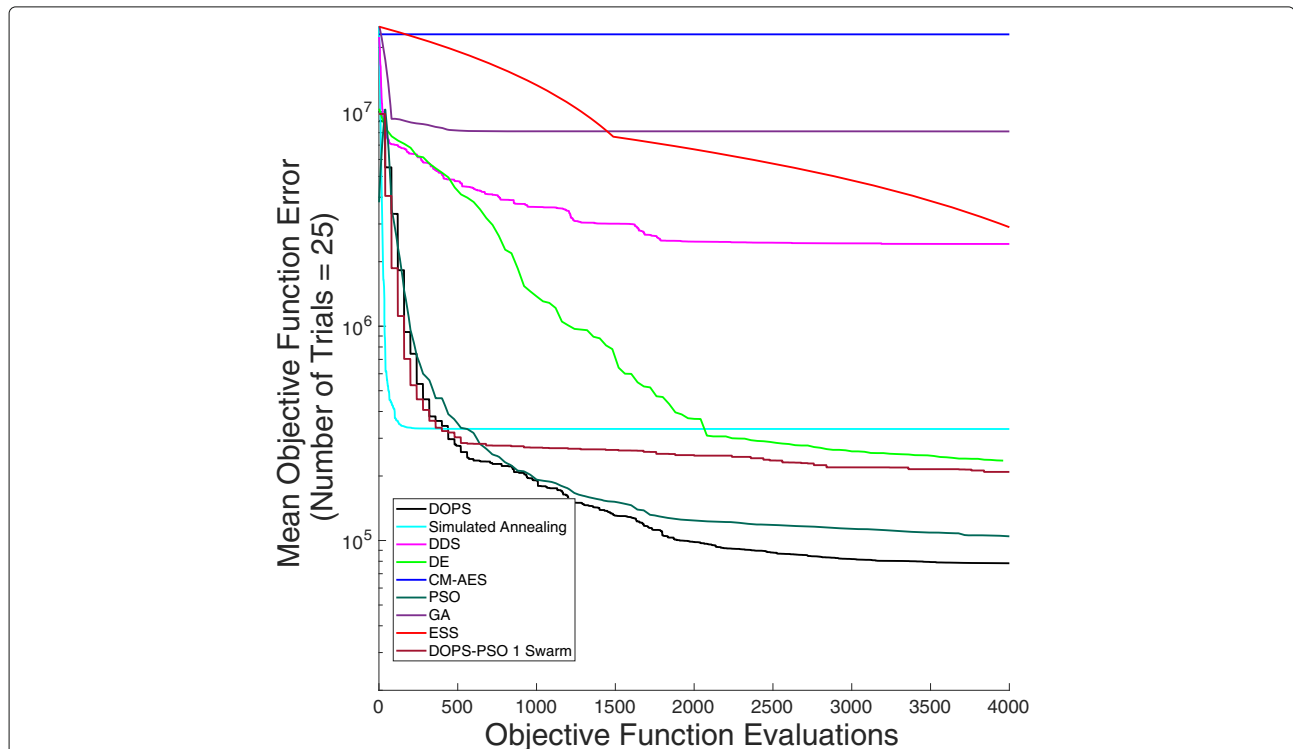


Fig. 5 Error convergence rates of the nine different algorithms on the coagulation model. The objective error is the mean over $\mathcal{T} = 25$ trials. DOPS, SA, PSO and DOPS-PSO have the steepest drop in error during first 300 function evaluations. Thereafter the error drop in DDS and SA remains nearly constant whereas DOPS continues to drops further. In the allotted budget of function evaluations ESS produces a modest reduction in error. At the end of 4000 function evaluations DOPS attains the lowest error

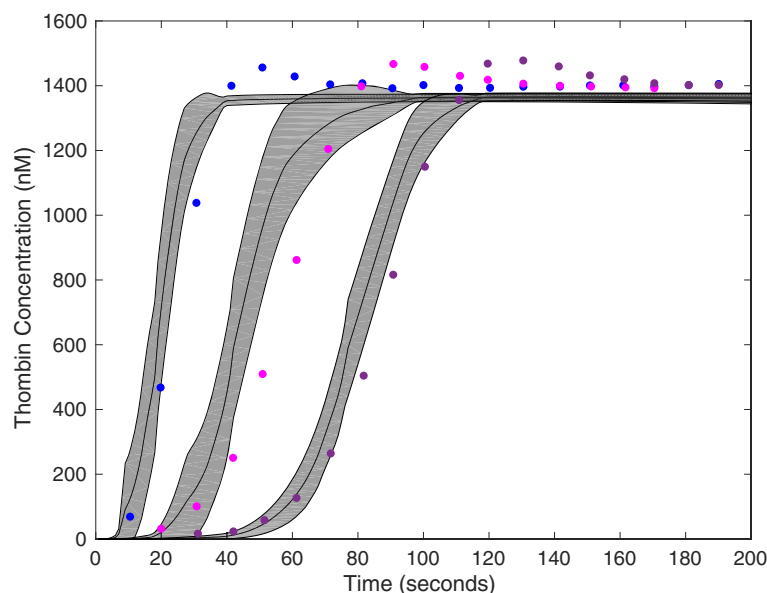


Fig. 6 Model predictions on unseen experimental data using parameters obtained from DOPS. The parameter estimates that were obtained using DOPS were tested against data that was not used in the model training. Solid black lines indicate the simulated mean thrombin concentration using parameter vectors from $\mathcal{T} = 25$ trials. The grey shaded region represents the 99% confidence estimate of the mean simulated thrombin concentration. The experimental data is reproduced from the synthetic plasma assays of Mann and co-workers. Thrombin generation is initiated by adding Factor VIIa-TF (500pM - blue, 50pM - pink and 10pM - purple, respectively) to synthetic plasma containing 200 $\mu\text{mol/L}$ of phospholipid vesicles (PCPS) and a mixture of coagulation factors (II,V,VII,VIII,IX,X and XI) at their mean plasma concentrations

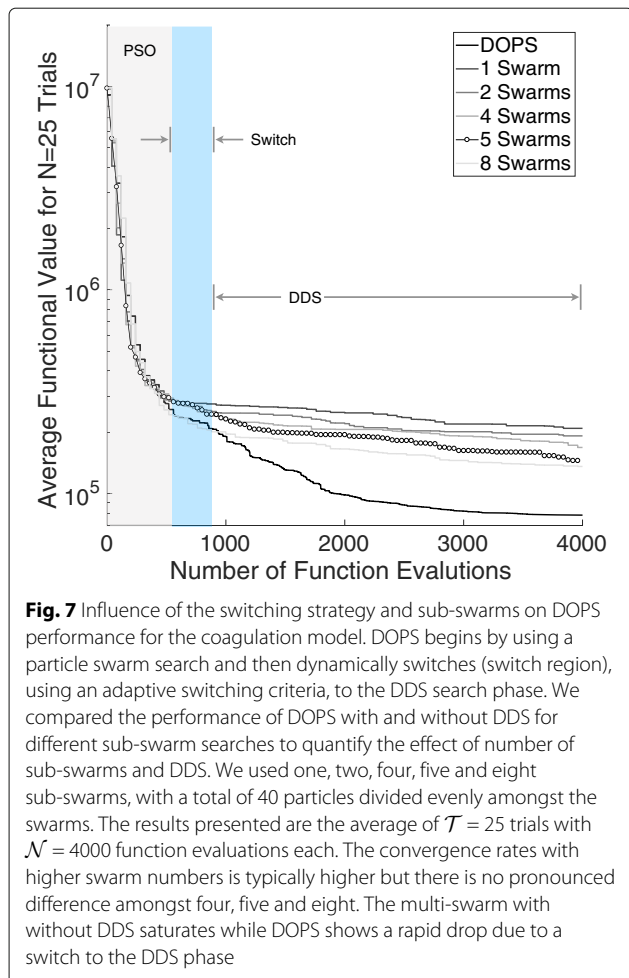
multi-swarm particle swarm optimization stagnated after 25% of the available function evaluations irrespective of the number of sub-swarms. However, DOPS (with five sub-swarms) switched to DDS after detecting the stagnation. The DDS phase refined the globally best particle to produce significantly lower error on average when compared to multi-swarm particle swarm optimization alone. Thus, the automated switching strategy was critical to the overall performance of DOPS. However, it was unclear if multiple strategy switches could further improve performance.

Table 2 Error analysis for the human coagulation model

TF/FVIIa concentration	Normalized S.E.	Category
5 nM	0.1336	Training
500 pM	0.2242	Prediction
50 pM	0.3109	Prediction
10 pM	0.2023	Prediction
5 pM	0.1170	Training

The coagulation model was trained on coagulation initiated with TF/FVIIa at 5 nM and the 5 pM to obtain the optimal parameters. Using these optimal parameters, coagulation dynamics were predicted for varying initiator concentrations (500 pM, 50 pM and 10 pM). Model agreement with measurements was quantified using normalized squared error. The normalized squared error is defined as $N.S.E. = (1/\max(\mathbf{X})) * (\|\mathbf{Y}, \mathbf{X}\|/\sqrt{N})$ where \mathbf{X} is the experimental data, \mathbf{Y} is the model simulation data interpolated onto the experimental time scale and N is the total number of experimental time points

We explored the performance of DOPS if it was permitted to switch between the PSO (Particle Swarm Optimization) and DDS modes multiple times. This mode (msDOPS) had comparable performance to DOPS on 10-d Ackley and Rastrigin functions, as well as on the 300-dimensional Rastrigin function. However, msDOPS performed better than DOPS on the CHO metabolism problem (Fig. 8a), with the average functional value being nearly half that of DOPS. To further distinguish DOPS from msDOPS, we compared the performance of each algorithm on the Eggholder function, a difficult function to optimize given its multiple minima [50]. msDOPS outperformed DOPS on the Eggholder function, however, neither version reached the true minimum at -959.6407 on any trial with a budget of $\mathcal{N} = 4000$ function evaluations (Fig. 8b). We also explored the performance of msDOPS and DOPS on the 100 dimensional Styblinski-Tang function [51] (Fig. 8c). In this comparison, msDOPS significantly outperformed DOPS, finding the true minimum before exhausting its function evaluation budget, while DOPS does not reach the minimum. Since the performance of msDOPS was promising on these problems, we measured its performance on the coagulation problem. Surprisingly, DOPS performed similarly to msDOPS on the coagulation problem (Fig. 8d); the final average objective value for DOPS reached 0.9413% of the initial functional value, compared to 0.9428% for msDOPS. Taken



together, these results indicate that switching plays a key role in DOPS's performance and that for some classes of problems, multiple switching between modes produces a faster drop in objective value. However, the coagulation model results suggested the advantage of msDOPS was problem specific.

Discussion

In this study, we developed dynamic optimization with particle swarms (DOPS), a novel meta-heuristic for parameter estimation. DOPS combined multi-swarm particle swarm optimization, a global search approach, with the greedy strategy of dynamically dimensioned search to estimate optimal or nearly optimal solutions in a fixed number of function evaluations. We tested the performance of DOPS and seven widely used meta-heuristics on the Ackley and Rastrigin test functions, a set of biochemical benchmark problems and a model of the human coagulation cascade. We also compared the performance of DOPS to enhanced Scatter Search (eSS), another widely used meta-heuristic approach. As the

number of parameters increased, DOPS outperformed the other meta-heuristics, generating optimal or nearly optimal solutions using significantly fewer function evaluations compared with the other methods. We tested the solutions generated by DOPS by comparing the estimated and true parameters in the benchmark studies, and by using the coagulation model to predict unseen experimental data. For both benchmark problems, DOPS retrieved the true parameters in significantly fewer function evaluations than other meta-heuristics. For the coagulation model, we used experimental coagulation measurements under two different conditions to estimate optimal or nearly optimal parameters. These parameters were then used to predict unseen coagulation data; the coagulation model parameters estimated by DOPS predicted the correct thrombin dynamics following TF/FVIIa induced coagulation without anticoagulants. Lastly, we showed the average performance of DOPS improved when combined with dynamically dimensioned search phase, compared to an identical multi-swarm approach alone, and that multiple mode switches could improve performance for some classes of problems. Taken together, DOPS is a promising meta-heuristic for the estimation of parameters in large biochemical models.

Meta-heuristics can be effective tools to estimate optimal or nearly optimal solutions for complex, multi-modal functions. However, meta-heuristics typically require a large number of function evaluations to converge to a solution compared with techniques that use derivative information. DOPS is a combination of particle swarm optimization, which is a global search method, and dynamically dimensioned search, which is a greedy evolutionary technique. Particle swarm optimization uses collective information shared amongst swarms of computational particles to search for global extrema. Several particle swarm variants have been proposed to improve the search ability and rate of convergence. These variations involve different neighborhood structures, multi-swarms or adaptive parameters. Multi-swarm particle swarm optimization with small particle neighborhoods has been shown to be better in searching on complex multi-modal solutions [36]. Multi-swarm methods generate diverse solutions, and avoid rapid convergence to local optima. However, at least for the coagulation problem used in this study, multi-swarm methods stagnated after approximately 25% of the available function evaluations; only the introduction of dynamically dimensioned search improved the rate of error convergence. Dynamically dimensioned search, which greedily perturbs only a subset of parameter dimensions in high dimensional parameter spaces, refined the globally best particle and produced significantly lower error on average when compared to multi-swarm particle swarm optimization alone. However, dynamically dimensioned search, starting from

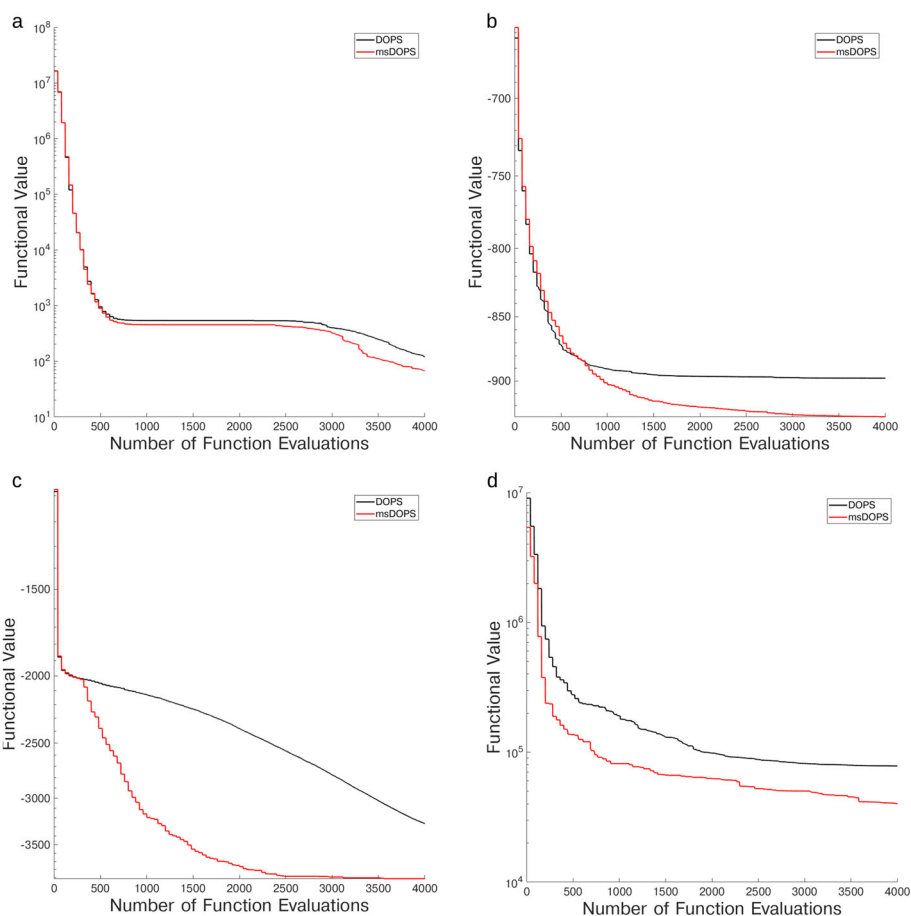


Fig. 8 Comparison of DOPS and Multiswitch DOPS Performance of DOPS and Multiswitch DOPS on the CHO metabolism problem (a), the Eggholder function (b), the 100 dimensional Styblinski-Tang function (c) and the coagulation problem (d). Both methods have the same initial decrease in error, but as the number of function evaluations increases, multiswitch DOPS produces a larger decrease in error. The results presented are the average of $\mathcal{T} = 250$ trials with for the CHO metabolism problem and $\mathcal{T} = 250$ trials on the Eggholder and Styblinski-Tang functions with $\mathcal{N} = 4000$ function evaluations each, and $\mathcal{T} = 25$ trials for the coagulation problem

a initial random parameter guess, was not as effective on average as DOPS. The initial solutions generated by the multi swarm search had a higher propensity to produce good parameter estimates when refined by dynamically dimensioned search. Thus, our hybrid combination of two meta-heuristics produced better results than either constituent approach, and better results than other meta-heuristic approaches on average. This was true of not only the convergence rate on the coagulation problem, but also the biochemical benchmark problems; DOPS required two-orders of magnitude fewer function evaluations compared with enhanced Scatter Search (eSS) to estimate the biochemical benchmark model parameters. What remains to be explored is the performance of DOPS compared to techniques that utilize derivative information, either on their own or in combination with other meta-heuristics, and the performance of DOPS in real-world applications compared with other meta-heuristics

such as hybrid genetic algorithms e.g., see [52]. Gradient methods perform well on smooth convex problems which have either a closed form of the gradient of the function being minimized, or a form that can be inexpensively estimated numerically. While the biological problems DOPS is intended for often do not have this form, perhaps the solutions could be further improved by following (or potentially replacing) the DDS phase with a gradient based technique when applicable. Taken together, the combination of particle swarm optimization and dynamically dimensioned search performed better than either of these constituent approaches alone, and required fewer function evaluations compared with other common meta-heuristics.

Conclusions

DOPS performed well on many different systems with no pre-optimization of algorithm parameters, however there

are many research questions that should be pursued further. DOPS comfortably outperformed existing, widely used meta-heuristics for high dimensional global optimization functions, biochemical benchmark models and a model of the human coagulation system. However, it is possible that highly optimized versions of common meta-heuristics could surpass DOPS; we should compare the performance of DOPS with optimized versions of the other common meta-heuristics on both test and real-world problems to determine if a performance advantage exists in practice. Next, DOPS has a hybrid architecture, thus the particle swarm phase could be combined with other search strategies such as local derivative based approaches to improve convergence rates. We could also consider multiple phases beyond particle swarm and dynamically dimensioned search, for example switching to a gradient based search following the dynamically dimensioned search phase. Lastly, we should update DOPS to treat multi-objective problems. The identification of large biochemical models sometimes requires training using qualitative, conflicting or even contradictory data sets. One strategy to address this challenge is to estimate experimentally constrained model ensembles using multiobjective optimization. Previously, we developed Pareto Optimal Ensemble Techniques (POETs) which integrates simulated annealing with Pareto optimality to identify models near the optimal tradeoff surface between competing training objectives [53]. Since DOPS consistently outperformed simulated annealing on both test and real-world problems, we expect a multi-objective form of DOPS would more quickly estimate solutions which lie along high dimensional trade-off surfaces.

Methods

Optimization problem formulation.

Model parameters were estimated by minimizing the difference between model simulations and \mathcal{E} experimental measurements. Simulation error is quantified by an objective function $K(\mathbf{p})$ (typically the Euclidean norm of the difference between simulations and measurements) subject to problem and parameter constraints:

$$\begin{aligned} \min_{\mathbf{p}} K(\mathbf{p}) &= \sum_{i=1}^{\mathcal{E}} (g_i(t_i, \mathbf{x}, \mathbf{p}, \mathbf{u}) - y_i)^2 \\ \text{subject to } \dot{\mathbf{x}} &= \mathbf{f}(t, \mathbf{x}(t), \mathbf{p}, \mathbf{u}(t), \mathbf{p}) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \\ \mathbf{c}(t, \mathbf{x}, \mathbf{p}, \mathbf{u}) &\geq \mathbf{0} \\ \mathbf{p}^L &\leq \mathbf{p} \leq \mathbf{p}^U \end{aligned} \quad (1)$$

The term $K(\mathbf{p})$ denotes the objective function (sum of squared error), t denotes time, $g_i(t_i, \mathbf{x}, \mathbf{p}, \mathbf{u})$ is the model output for experiment i , while y_i denotes the measured value for experiment i . The quantity $\mathbf{x}(t, \mathbf{p})$ denotes the

state variable vector with an initial state \mathbf{x}_0 , $\mathbf{u}(t)$ is a model input vector, $\mathbf{f}(t, \mathbf{x}(t), \mathbf{p}, \mathbf{u}(t), \mathbf{p})$ is the system of model equations (e.g., differential equations or algebraic constraints) and \mathbf{p} denotes the model parameter vector (quantity to be estimated). The parameter search (or model simulations) can be subject to $\mathbf{c}(t, \mathbf{x}, \mathbf{p}, \mathbf{u})$ linear or non-linear constraints, and parameter bound constraints where \mathbf{p}^L and \mathbf{p}^U denote the lower and upper parameter bounds, respectively. Optimal model parameters are then given by:

$$\mathbf{p}^* = \arg \min_{\mathbf{p}} K(\mathbf{p}) \quad (2)$$

In this study, we considered only parameter bound constraints, and did not include the $\mathbf{c}(t, \mathbf{x}, \mathbf{p}, \mathbf{u})$ linear or non-linear problem constraints. However, additional these constraints can be handled, without changing the approach, using a penalty function method.

Dynamic optimization with particle swarms (DOPS).

DOPS combines multi-swarm particle swarm optimization with dynamically dimensioned search (Fig. 1) and (Algorithm 1). The goal of DOPS is to estimate optimal or near optimal parameter vectors for high-dimensional biological models within a specified number of function evaluations. Toward this objective, DOPS begins by using a particle swarm search and then dynamically switches, using an adaptive switching criteria, to a DDS search phase.

Phase 1: Particle swarm phase.

Particle swarm optimization is an evolutionary algorithm that uses a population of particles (solutions) to find an optimal solution [54, 55]. Each particle is updated based on its experience (particle best) and the experience of all other particles within the swarm (global best). The particle swarm phase of DOPS begins by randomly initializing a swarm of \mathcal{K} -dimensional particles (represented as z_i), wherein each particle corresponded to a \mathcal{K} -dimensional parameter vector. After initialization, particles were randomly partitioned into k equally sized sub-swarms $\mathcal{S}_1, \dots, \mathcal{S}_k$. Particles within each sub-swarm \mathcal{S}_k were updated according to the rule:

$$z_{i,j} = \theta_{1,j-1} z_{i,j-1} + \theta_{2,j-1} (\mathcal{L}_i - z_{i,j-1}) + \theta_{3,j-1} (\mathcal{G}_k - z_{i,j-1}) \quad (3)$$

where $(\theta_1, \theta_2, \theta_3)$ were adjustable parameters, \mathcal{L}_i denotes the best solution found by particle i within sub-swarm \mathcal{S}_k for function evaluation $1 \rightarrow j-1$, and \mathcal{G}_k denotes the best solution found over all particles within sub-swarm \mathcal{S}_k . The quantities r_1 and r_2 denote uniform random vectors with the same dimension as the number of unknown model parameters ($\mathcal{K} \times 1$). Equation 3 is similar to the general particle swarm update rule, however, it does not

Algorithm 1: Pseudo code for the dynamic optimization with particle swarms (DOPS) method

input : A randomized swarm of particles of size $NP \times K$ and fixed number of function evaluations N

output: Optimized parameter vector of size $1 \times K$

- 1 Initialize the particles randomly and assign particles randomly to various sub-swarms;
- 2 **while** $j \leq N$ **do**
- 3 **if** $\text{mod}(j, G) = 0$ **then**
- 4 | Reassign particles to different sub-swarms;
- 5 **end**
- 6 **for** $i \leftarrow 1$ to NS **do**
- 7 | Update particles within sub-swarms according to Eq. 3;
- 8 **end**
- 9 Find best particle \mathcal{G} amongst all sub-swarms;
- 10 **if** $\text{besterror}(j) \geq 0.99 * \text{besterror}(j + 1)$ **then**
- 11 | $\text{failurecounter} \leftarrow \text{failurecounter} + 1$;
- 12 **else**
- 13 | $\text{failurecounter} \leftarrow 0$;
- 14 **end**
- 15 **if** $\text{failurecounter} \geq \text{threshold}$ **then**
- 16 | $\mathcal{G} \leftarrow \text{DDS}(\mathcal{G}, N - j)$;
- 17 | **return** \mathcal{G}
- 18 **else**
- 19 | $j \leftarrow j + 1$;
- 20 **end**
- 21 **return** \mathcal{G}
- 22 **end**

contain velocity terms. In DOPS, the parameter $\theta_{1,j-1}$ is similar to the inertia weight parameter for the velocity term described by Shi and Eberhart [56]; Shi and Eberhart proposed a linearly decreasing inertia weight to improve convergence properties of particle swarm optimization. Our implementation of $\theta_{1,j-1}$ is inspired by this and the decreasing perturbation probability proposed by Tolson and Shoemaker [34]. It is an analogous equivalent to inertia weight on velocity. However $\theta_{1,j-1}$ places inertia on the position rather than velocity and uses the same rule described by Shi and Eberhart to adaptively change with the number of function evaluations:

$$\theta_{1,j} = \frac{(\mathcal{N} - j) * (w_{max} - w_{min})}{(\mathcal{N} - 1)} + w_{min} \quad (4)$$

where \mathcal{N} represents the total number of function evaluations, w_{max} and w_{min} are the maximum and minimum inertia weights, respectively. In this study, we used $w_{max} = 0.9$ and $w_{min} = 0.4$, however, these values are user configurable and could be changed depending upon the problem being explored. Similarly, θ_2 and θ_3 were treated

as constants, where $\theta_2 = \theta_3 = 1.5$; the values of θ_2 and θ_3 control how heavily the particle swarm weighs the previous solutions it has found when generating a new candidate solution. If $\theta_2 \gg \theta_3$ the new parameter solution will resemble the best *local* solution found by particle i (\mathcal{L}_i), while $\theta_3 \gg \theta_2$ suggests the new parameter solution will resemble the best *global* solution found so far. While updating the particles, parameter bounds were enforced using reflection boundary conditions (Algorithm 2).

Algorithm 2: Pseudo code for the reflective boundary conditions used by the dynamic optimization with particle swarms (DOPS) method

- 1 **if** $z_{i,j}^{old} < z_i^{min}$ **then**
- 2 | $z_{i,j}^{new} = z_{i,j}^{old} + (z_i^{min} - z_{i,j}^{old})$ **if** $z_{i,j}^{new} > z_i^{max}$ **then**
- 3 | $z_{i,j}^{new} = z_i^{max}$
- 4 **end**
- 5 **end**
- 6 **if** $z_{i,j}^{old} > z_i^{max}$ **then**
- 7 | $z_{i,j}^{new} = z_{i,j}^{old} + (z_{i,j}^{old} - z_i^{max})$ **if** $z_{i,j}^{new} < z_i^{min}$ **then**
- 8 | $z_{i,j}^{new} = z_i^{min}$
- 9 **end**
- 10 **end**

After every \mathcal{M} function evaluations, particles were randomly redistributed to a new sub-swarm, and updated according to Eq. (3). This process continued for a maximum of \mathcal{FN} functions evaluations, where \mathcal{F} denotes the fraction of function evaluations used during the particle swarm phase of DOPS:

$$\mathcal{F} = \left(\frac{NP}{\mathcal{N}} \right) j \quad (5)$$

The quantity NP denotes the total number of particles in the swarm, \mathcal{N} denotes the total possible number of function evaluations, while the counter j denotes the number of successful particle swarm iterations (each costing NP function evaluations). If the simulation error stagnated e.g., did not change by more than 1% for a specified number of evaluations (default value of 4), the swarm phase was terminated and DOPS switched to exploring parameter space using the DDS approach using the remaining $(1 - \mathcal{F})\mathcal{N}$ function evaluations.

Phase 2: DDS phase.

Dynamically Dimensioned Search (DDS) is a single solution based search algorithm. DDS is used to obtain good solutions to high-dimensional search problems within a fixed number of function evaluations. DDS starts as a global search algorithm by perturbing all the dimensions.

Algorithm 3: Pseudo code for the Dynamically Dimensioned Search (DDS) method

input : Candidate vector \mathcal{G} from swarm search and $(1 - \mathcal{F})\mathcal{N}$ evaluations
output: Optimized parameter vector of size $1 \times K$

- 1 **while** $j \leq (1 - \mathcal{F})\mathcal{N}$ **do**
- 2 Assign probability of perturbation to each dimension \mathcal{P}_i according to Eq. 7;
- 3 Select a subset of dimensions based on a threshold value for perturbation;
- 4 Update candidate solution $\mathcal{G}(J)$ according to Eq. 5;
- 5 Ensure updated solution $\mathcal{G}_{new}(J)$ is within bounds using Algorithm 2;
- 6 **end**

Later the number of dimensions that are perturbed is decreased with a certain probability. The probability that a certain dimension is perturbed reduces (a minimum of one dimension is always perturbed) as the iterations increase. This causes the algorithm to behave as a local search algorithm as the number of iterations increase. The perturbation magnitude of each dimension is from normal distribution with zero mean. The standard deviation that was used in the original DDS paper and the current study is 0.2. DDS performs a greedy search where the solution is updated only if it is better than the previous solution. The combination of perturbing a subset of dimensions along with greedy search indirectly relies on model sensitivity to a specific parameter combination. The reader is requested to refer to the original paper by Tolson and Shoemaker for further detail [34].

At the conclusion of the swarm phase, the overall best particle, \mathcal{G}_k , over the k sub-swarms was used to initialize the DDS phase. DOPS takes at least $(1 - \mathcal{F})\mathcal{N}$ function evaluations during the DDS phase and then terminates the search. For the DDS phase, the best parameter estimate was updated using the rule:

$$\mathcal{G}_{new}(J) = \begin{cases} \mathcal{G}(J) + \mathbf{r}_{normal}(J)\sigma(J), & \text{if } \mathcal{G}_{new}(J) < \mathcal{G}(J). \\ \mathcal{G}(J), & \text{otherwise.} \end{cases} \quad (6)$$

where \mathbf{J} is a vector representing the subset of dimensions that are being perturbed, \mathbf{r}_{normal} denotes a normal random vector of the same dimensions as \mathcal{G} , and σ denotes the perturbation amplitude:

$$\sigma = R(\mathbf{p}^U - \mathbf{p}^L) \quad (7)$$

where R is the scalar perturbation size parameter, \mathbf{p}^U and \mathbf{p}^L are $(K \times 1)$ vectors that represent the maximum and

minimum bounds on each dimension. The set \mathbf{J} was constructed using a probability function \mathcal{P}_i that represents a threshold for determining whether a specific dimension j was perturbed or not; \mathcal{P}_i is monotonically decreasing function of function evaluations:

$$\mathcal{P}_i = 1 - \log \left[\frac{i}{(1 - \mathcal{F})\mathcal{N}} \right] \quad (8)$$

where i is the current iteration. After \mathcal{P}_i is determined, we drew \mathcal{P}_j from a uniform distribution for each dimension j . If $\mathcal{P}_j < \mathcal{P}_i$ was included in \mathbf{J} . Thus, the probability that a dimension j was perturbed was inversely proportional to the number of function evaluations. DDS updates are greedy; \mathcal{G}_{new} becomes the new solution vector only if it is better than \mathcal{G} .

Multiswitch DOPS

We investigated whether switching search methods more than once would result in better performance; this DOPS variant is referred to as multiswitch DOPS or msDOPS. msDOPS begins with the PSO phase and uses the same criteria as DOPS to switch to the DDS phase. However, msDOPS can switch back to a PSO search when the DDS phase has reduced the functional value to 90% of its initial value. Should the DDS phase fail to improve the functional value sufficiently, this version is identical to DOPS. When the switch from DDS to PSO occurs, we use the best solution from DDS to seed the particle swarm. DOPS and msDOPS source code is available for download under a MIT license at <http://www.varnerlab.org>.

Comparison techniques

The implementations of particle swarm optimization, simulated annealing, and genetic algorithms are the ones given in Matlab R2017A (particleswarm, simulannealbnd and ga). The implementation of DE used was developed by R.Storn and available at <http://www1.icsi.berkeley.edu/~storn/code.html>. The version of eSS used was Release 2014B - AMIGO2014bench VERSION WITH eSS MAY-2014-BUGS FIXED - JRB, released by the Process Engineering Group IIM-CSIC. The genetic algorithm, particle swarm, and differential evolution algorithms were run with a 40 particles to be directly comparable to the number of particles used in the PSO phase of DOPS. For comparison, the version of CM-AES used was cmaes.m, Version 3.61.beta from http://cma.gforge.inria.fr/cmaes_sourcecode_page.html. The scripts used to run the comparison methods are also available at <http://www.varnerlab.org>.

Additional files

Additional file 1: Figure S1. Data fits for CHO metabolism problem. (PDF 130 kb)

Additional file 2: Figure S2. Data fits for *S.cerevisiae* metabolism problem. (PDF 22 kb)

Additional file 3: Figure S3. Comparison of states and parameters. (PDF 192 kb)

Additional file 4: Figure S4. Time Comparison. (PNG 28 kb)

Additional file 5: Figure S5. Convergence Curves. (PDF 116 kb)

Additional file 6: Figure S6. Comparison of DOPS to ESS. (PDF 73 kb)

Additional file 7: Figure S7. Comparison of functional values. (PDF 123 kb)

Additional file 8: Figure S8. Dispersion Curves. (PDF 89 kb)

Funding

This study was supported by an award from the US Army and Systems Biology of Trauma Induced Coagulopathy (W911NF-10-1-0376).

Availability of data and materials

DOPS is open source, available under an MIT software license from <http://www.varnerlab.org>. All scripts used in this study are included in the `utilities` directory. The optimization problems used are located in the `CoagulationFiles` and `testFunctions` directories.

Authors' contributions

AS developed original DOPS algorithm. RL developed msDOPS and generated figures in the manuscript. CS developed the dynamic dimensional search approach, and assisted in editing the manuscript. JV directed the study, prepared and edited the manuscript. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

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

Author details

¹Robert Fredrick Smith School of Chemical and Biomolecular Engineering, Cornell University, 244 Olin Hall, Ithaca, NY, USA. ²School of Civil and Environmental Engineering, Cornell University, Ithaca, NY, USA.

Received: 29 December 2017 Accepted: 17 September 2018

Published online: 12 October 2018

References

- Asmus HE, Herwig R, Cho K-H, Wolkenhauer O. Dynamics of biological systems: role of systems biology in medical research. *Expert Rev Mol Diagn.* 2006;6:891–902.
- van Riel NAW. Dynamic modelling and analysis of biochemical networks: mechanism-based models and model-based experiments. *Brief Bioinform.* 2006;7(4):364–74. <https://doi.org/10.1093/bib/bbl040>.
- Jaqaman K, Danuser G. Linking data to models: data regression. *Nat Rev Mol Cell Biol.* 2006;7(11):813–9. <https://doi.org/10.1038/nrm2030>.
- Kitano H. Systems biology: a brief overview. *Science.* 2002;295(5560):1662–4.
- Hood L, Heath JR, Phelps ME, Lin B. Systems biology and new technologies enable predictive and preventative medicine. *Science.* 2004;306(5696):640–3.
- Aldridge BB, Burke JM, Lauffenburger DA, Sorger PK. Physicochemical modelling of cell signalling pathways. *Nat Cell Biol.* 2006;8(11):1195–203. <https://doi.org/10.1038/ncb1497>.
- Banga JR. Optimization in computational systems biology. *BMC Syst Biol.* 2008;2(1):47.
- Ashyraliyev M, Fomekong-Nanfack Y, Kaandorp JA, Blom JG. Systems biology: parameter estimation for biochemical models. *Febs J.* 2009;276(4):886–902.
- Moles CG, Mendes P, Banga JR. Parameter estimation in biochemical pathways: a comparison of global optimization methods. *Genome Res.* 2003;13(11):2467–74.
- Nieman R, Fisher D, Seborg D. A review of process identification and parameter estimation techniques†. *Int J Control.* 1971;13(2):209–64.
- Beck JV, Arnold KJ. *Parameter Estimation in Engineering and Science.* New York: Wiley; 1977.
- Young P. Parameter estimation for continuous-time models—a survey. *Automatica.* 1981;17(1):23–39.
- Beck JV, Woodbury KA. Inverse problems and parameter estimation: integration of measurements and analysis. *Meas Sci Technol.* 1998;9(6):839.
- Hooke R, Jeeves TA. “direct search” solution of numerical and statistical problems. *J ACM.* 1961;8(2):212–29.
- Nelder JA, Mead R. A simplex method for function minimization. *Comput J.* 1965;7(4):308–13.
- Moré JJ. The Levenberg-Marquardt algorithm: implementation and theory. In: *Numerical Analysis.* Berlin: Springer; 1978. p. 105–116.
- Esposito WR, Floudas CA. Deterministic global optimization in nonlinear optimal control problems. *J Glob Optim.* 2000;17(1-4):97–126.
- Horst R, Tuy H. *Global Optimization Approaches.* Berlin: Springer-Verlag; 2013.
- Goldberg DE. *Genetic Algorithms.* India: Pearson Education; 2006.
- Kirkpatrick S, Gelatt CD, Vecchi MP, et al. Optimization by simulated annealing. *Science.* 1983;220(4598):671–80.
- Fogel D. *Artificial intelligence through simulated evolution.* New York: Wiley; 2009.
- Storn R, Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim.* 1997;11(4):341–59.
- Tsai K-Y, Wang F-S. Evolutionary optimization with data collocation for reverse engineering of biological networks. *Bioinformatics.* 2005;21(7):1180–8.
- Wang F-S, Su T-L, Jang H-J. Hybrid differential evolution for problems of kinetic parameter estimation and dynamic optimization of an ethanol fermentation process. *Ind Eng Chem Res.* 2001;40(13):2876–85.
- Noman N, Iba H. Inferring gene regulatory networks using differential evolution with local search heuristics. *IEEE/ACM Trans Comput Biol Bioinforma.* 2007;4(4):634–47.
- Sun J, Garibaldi JM, Hodgman C. Parameter estimation using metaheuristics in systems biology: a comprehensive review. *Comput Biol Bioinforma IEEE/ACM Trans.* 2012;9(1):185–202.
- Mendes P, Kell D. Non-linear optimization of biochemical pathways: applications to metabolic engineering and parameter estimation. *Bioinformatics.* 1998;14(10):869–83.
- Modchang C, Triampo W, Lenbury Y. Mathematical modeling and application of genetic algorithm to parameter estimation in signal transduction: Trafficking and promiscuous coupling of g-protein coupled receptors. *Comput Biol Med.* 2008;38(5):574–82.
- Tashkova K, Korošec P, Šilc J, Todorovski L, Džeroski S. Parameter estimation with bio-inspired meta-heuristic optimization: modeling the dynamics of endocytosis. *BMC Syst Biol.* 2011;5(1):159.
- Villaverde AF, Egea JA, Banga JR. A cooperative strategy for parameter estimation in large scale systems biology models. *BMC Syst Biol.* 2012;6(1):75.
- Rodríguez-Fernández M, Egea JA, Banga JR. Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. *BMC Bioinformatics.* 2006;7(1):483.
- Egea JA, Rodríguez-Fernández M, Banga JR, Martí R. Scatter search for chemical and bio-process optimization. *J Glob Optim.* 2007;37(3):481–503.
- Villaverde AF, Henriques D, Smallbone K, Bongard S, Schmid J, Cicin-Sain D, Crombach A, Saez-Rodríguez J, Mauch K, Balsa-Canto E, et al. Biopredyn-bench: a suite of benchmark problems for dynamic modelling in systems biology. *BMC Syst Biol.* 2015;9(1):8.

34. Tolson BA, Shoemaker CA. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resour Res.* 2007;43(1):W01413.
35. Cheung NJ, Ding X-M, Shen H-B. Optifel: A convergent heterogeneous particle swarm optimization algorithm for takagi-sugeno fuzzy modeling. *IEEE Trans Fuzzy Syst.* 2014;22(4):919–33.
36. Zhao S-Z, Liang JJ, Suganthan PN, Tasgetiren MF. Dynamic multi-swarm particle swarm optimizer with local search for large scale global optimization. 2008 IEEE Congress on Evolutionary Computation (CEC 2008). 2008;3845–3852. <https://doi.org/10.1109/CEC.2008.4631320>.
37. Villaverde AF, et al. High-confidence predictions in systems biology dynamic models. In: Saez-Rodriguez J, Rocha M, Fdez-Riverola F, De Paz Santana J, editors. 8th International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB 2014). *Advances in Intelligent Systems and Computing*, vol 294. Cham: Springer; 2014. https://doi.org/10.1007/978-3-319-07581-5_20.
38. Smallbone K, Mendes P. Large-scale metabolic models: From reconstruction to differential equations. *Ind Biotechnol.* 2013;9(4):179–84.
39. Mann KG, Butenas S, Brummel K. The dynamics of thrombin formation. *Arterioscler Thromb Vasc Biol.* 2003;23(1):17–25.
40. Mann K, Brummel K, Butenas S. What is all that thrombin for? *J Thromb Haemost.* 2003;1(7):1504–14.
41. Mann KG. Thrombin formation. *Chest.* 2003;124(3):4–10.
42. Vogler EA, Siedlecki CA. Contact activation of blood-plasma coagulation. *Biomaterials.* 2009;30(10):1857–69.
43. Diamond SL. Systems biology of coagulation. *J Thromb Haemost.* 2013;11(s1):224–32.
44. Fogelson AL, Tania N. Coagulation under flow: the influence of flow-mediated transport on the initiation and inhibition of coagulation. *Pathophysiol Haemost Thromb.* 2005;34(2-3):91–108.
45. Anand M, Rajagopal K, Rajagopal K. A model incorporating some of the mechanical and biochemical factors underlying clot formation and dissolution in flowing blood: review article. *J Theor Med.* 2003;5(3-4):183–218.
46. Hockin MF, Jones KC, Everse SJ, Mann KG. A model for the stoichiometric regulation of blood coagulation. *J Biol Chem.* 2002;277(21):18322–33.
47. Chatterjee MS, Denney WS, Jing H, Diamond SL. Systems biology of coagulation initiation: kinetics of thrombin generation in resting and activated human blood. *PLoS Comput Biol.* 2010;6(9):e1000950.
48. Mann KG, Brummel-Ziedins K, Orfeo T, Butenas S. Models of blood coagulation. *Blood Cells Mol Dis.* 2006;36(2):108–17.
49. Luan D, Zai M, Varner JD. Computationally derived points of fragility of a human cascade are consistent with current therapeutic strategies. *PLoS Comput Biol.* 2007;3(7):142.
50. Jamil M, Yang X-S. A literature survey of benchmark functions for global optimization problems. *Int J Math Model Numer Optimisation.* 2013;4(2):150–94.
51. Jamil M, Yang X-S. A literature survey of benchmark functions for global optimisation problems. *Int J Math Model Numer Optimisation.* 2013;4(2):150–94.
52. Montefusco F, Akman OE, Soyer OS, Bates DG. Ultrasensitive negative feedback control: A natural approach for the design of synthetic controllers. *PLoS ONE.* 2016;11(8):0161605. <https://doi.org/10.1371/journal.pone.0161605>.
53. Bassen DM, Vilkhovoy M, Minot M, Butcher JT, Varner JD. Jupoets: a constrained multiobjective optimization approach to estimate biochemical model ensembles in the julia programming language. *BMC Syst Biol.* 2017;11(1):10. <https://doi.org/10.1186/s12918-016-0380-2>.
54. Clerc M. Particle Swarm Optimization. London: ISTE; 2006. <http://www.loc.gov/catdir/toc/ecip065/2005037211.html>.
55. Abraham A, Guo H, Liu H. Swarm intelligence: foundations, perspectives and applications. Berlin: Springer; 2006. pp. 3–25.
56. Shi Y, Eberhart RC. Empirical study of particle swarm optimization. In: *Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99*, vol. 3. Piscataway: IEEE; 1999. p. 1950.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

