Computer Methods and Programs in Biomedicine 243 (2024) 107933

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Uncovering personal circadian responses to light through particle swarm optimization

Jesús Vicente-Martínez [a,b], María Ángeles Bonmatí-Carrión [a,b,*], Juan Antonio Madrid [a,b], Maria Angeles Rol [a,b]

[a] Chronobiology Laboratory, Department of Physiology, College of Biology, University of Murcia, Mare Nostrum Campus, IUIE, IMIB-Arrixaca, Murcia 30100, Spain
[b] Ciber Fragilidad y Envejecimiento Saludable, Instituto de Salud Carlos III, Madrid 28029, Spain

ARTICLE INFO

Keywords:
Particle swarm optimization
Kronauer's oscillator model
Circadian personalization
Circadian response to light
Parameter optimization of ordinary differential equations
Heuristic algorithms

ABSTRACT

Background and objectives: Kronauer's oscillator model of the human central pacemaker is one of the most commonly used approaches to study the human circadian response to light. Two sources of error when applying it to a personal light exposure have been identified: (1) as a populational model, it does not consider inter-individual variability, and (2) the initial conditions needed to integrate the model are usually unknown, and thus subjectively estimated. In this work, we evaluate the ability of particle swarm optimization (PSO) algorithms to simultaneously uncover the optimal initial conditions and individual parameters of a pre-defined Kronauer's oscillator model.
Methods: A Canonical PSO, a Dynamic Multi-Swarm PSO and a novel modification of the latter, namely Hierarchical Dynamic Multi-Swarm PSO, are evaluated. Two different target models (under a regular and an irregular schedule) are defined, and the same realistic light profile is fed to them. Based on their output, a fitness function is proposed, which is minimized by the algorithms to find the optimum set of parameters and initial conditions of the model.
Results: We demonstrate that Dynamic Multi-Swarm and Hierarchical Dynamic Multi-Swarm algorithms can accurately uncover personal circadian parameters under both regular and irregular schedules, but as expected, optimization is easier under a regular schedule. Circadian parameters play the most important role in the optimization process and should be prioritized over initial conditions, although assessment of the impact of misestimating the latter is recommended. The log-log linear relationship between mean absolute error and computational cost shows that the number of particles to use is at the discretion of the user.
Conclusions: The robustness and low errors achieved by the algorithms support their further testing, validation and systematic application to empirical data under a regular or irregular schedule. Uncovering personal circadian parameters can improve the assessment of the circadian status of a person and the applicability of personalized light therapies, as well as help to discover other factors that may lie behind the interindividual variability in the circadian response to light.

1. Introduction

Many human physiological processes oscillate spontaneously with a period close to 24 h, in a clear adaptation to the Earth's rotational period and the day/night cycle. The synchronization of these processes is regulated and fine-tuned by the central pacemaker located in the suprachiasmatic nucleus, the main Zeitgeber of which is the light/dark cycle [1]. In order to study the dynamic effects of light on the central pacemaker, one of the most commonly used models is a modified Van der Pol oscillator, first proposed by Kronauer [2] and refined several times [3–5] based on data provided by new laboratory-based experiments. Although numerous studies [6–16] have demonstrated the

Abbreviations: IVP, initial value problem; CBT, core body temperature; PSO, particle swarm optimizers/optimization; ODE, ordinary differential equation(s); CBTmin, minimum of core body temperature; DMS-PSO, dynamic multi-swarm PSO; H-DMSPSO, hierarchical dynamic multi-swarm PSO; CC, computational cost.

* Corresponding author at: Chronobiology Laboratory, Department of Physiology, College of Biology, University of Murcia, Mare Nostrum Campus, IUIE, IMIB-Arrixaca, Murcia 30100, Spain.
E-mail address: mabonmati@um.es (M.Á. Bonmatí-Carrión).

https://doi.org/10.1016/j.cmpb.2023.107933
Received 23 May 2023; Received in revised form 2 October 2023; Accepted 17 November 2023
Available online 19 November 2023

relevance of Kronauer's model, two main concerns inherent to the modeling process itself affect its direct applicability: inter-individual variability and initial conditions estimation.

As with any populational model with fixed parameters, Kronauer's model does not consider the inter-individual variability of the population, which may result in a deviation of some of the default parameters from their real value, with a subsequent error in the results yielded. It would therefore be desirable to fit one or more of its parameters to each person, so that the model represents more accurately the particular physiology of each subject. Regarding which parameters are the best candidates for this fitting, certain factors must be considered, including: the biological relevance of the parameter in relation to the interpretability of the results obtained and their experimental validation, the variability of the results explained by the parameter within its biologically-coherent limits or the total number of parameters to be fitted, as this directly affects the computational cost and the number of experiments to be conducted to determine their real value.

The integration of Kronauer's model requires establishing and solving an initial value problem (IVP). The usual approach when working with an IVP consists of simulating the evolution of the system to make predictions about its behavior, and comparing the results to the empirical data. The opposite problem −finding the optimal model parameters from available empirical data− is relatively uncommon and has not been studied in much depth [17]. This approach is particularly problematic when a limited number of empirical measurements are available [18], as is the case in Kronauer's model, the standard output of which is the time of day when the minimum core body temperature (CBT) occurs. Moreover, the problem becomes even more challenging, given the strongly non-linear nature of the Van der Pol oscillator. As a result, personalization of the circadian response to light by means of the parameters of Kronauer's model has only been attempted on a few occasions, and for a limited number of parameters. In the work by Bonarius et al. [6], the model –combined with the Phillips-Robinson sleep model [19]– was transformed into a Bayesian stochastic framework to optimize the internal period (τ). Stone et al. [15] selected and simultaneously optimized three of its parameters (τ - internal period, p - light sensitivity and k - balance of the response curve to light) by means of least-squares fitting. Based on their results, the same three parameters have been chosen to be optimized in the present work.

On the other hand, considering that Kronauer's is a biological model, it is impossible to factually know the initial conditions of the state variables without recurring to their experimental measurement. Thus, when available, they may be estimated from an experimental phase marker by rough estimation [15] or by fitting the system output to an experimental waveform [9]. However, experimental measurements are not usually available, and thus the most common procedure is to estimate them by making other certain reasonable assumptions. In simulations under a regular illuminance pattern, one option is to assume that the system outputs are also stable day by day, as the authors of the model did in their original published papers [3–5,20]. Similarly, stable values can be achieved by allowing them to converge after several days of integration [12], once again assuming that daily illuminance is stable over time. The same procedure can be also applied to simulations or recordings over several days, repeating the integration over the entire period as many times as necessary, until convergence is reached [16]. However, this approach is not feasible when stability cannot be guaranteed; indeed, the importance of the correct choice of initial conditions becomes critical if we consider situations with strong phase misalignments, such as jet-lag or shift work [7]. Therefore, it may be appropriate to find their optimum value rather than subjectively estimating them, even at the cost of a higher computational effort. In turn, in high dimensional and computationally demanding problems like the one we are posing, heuristic algorithms are prospectively a good optimization methodology, since they favor speed over other evaluation metrics, sacrificing better accuracy or precision for lower computational cost.

Among the many meta-heuristic optimization schemes developed [21], Particle Swarm Optimizers (PSO) were chosen because of their simplicity and robustness. They are inspired by the collective behavior of organisms, where each individual (particle) in the swarm adjusts its position based on its own experience and the information shared with neighboring particles. PSO algorithms are relatively simple to understand and implement, as compared to other evolutionary algorithms like genetic algorithms or differential evolution, making them more accessible for researchers and practitioners to implement and modify according to their own needs. Moreover, applications for PSO algorithms have recently been found in various biomedical fields, including feature and gene selection [22,23], chemotherapy [24] and even the parameter optimization of biomedical models [25], which is the type of application described in this paper. The original PSO algorithm was published in 1995 by Eberhart and Kennedy [26], but many variants have been developed thereafter. Some are focusing on dividing the swarm into smaller sub-swarms for better search efficiency and to avoid premature convergence on local minima, one of the main drawbacks of these algorithms. For a general overview of PSO and its variants and applications, see [27,28].

In our study, a realistic one-week light profile is used as the input to Kronauer's oscillator model to evaluate the ability of different particle swarm optimization algorithms to uncover three pre-set circadian parameters: τ (internal period), p (light sensitivity) and k (balance of the response curve to light). Initial conditions are treated as free independent variables, grouping them together with the circadian parameters into a 6-dimensional optimization scheme and thus avoiding subjective estimations when applying the model. Due to the high non-linearities of the model and the dimensionality of the optimization problem, we hypothesize that particle-based heuristic algorithms like PSO may be an efficient answer to such difficult tasks. Among the PSO variants already developed, we evaluate the Canonical PSO [29] and a Dynamic Multi-Swarm PSO (DMS-PSO) [30]. Furthermore, we propose and evaluate a novel generalization of the latter, namely, Hierarchical Dynamic Multi-Swarm PSO.

2. Methodology

2.1. Oscillator circadian model

Kronauer's dynamic model is a system of three ordinary differential equations (ODE) based on two coupled processes that transform a light input into a signal that entrains a modified Van der Pol limit cycle oscillator [31], which represents the human central pacemaker. In this study, the St. Hilaire's version of the model is used [5], in which the light processing was modified to improve its behavior in low-light conditions. The output of the model, which is used to evaluate the fitness of the proposed solution by the algorithm, consists of the time at which the minimum of the core body temperature (CBTmin) occurs each day. The full equations, details and values of the parameters are given in the Appendix A. Parameter values specified in Table A.1 define the 'default' model.

2.2. Target parameters

The three circadian parameters to be optimized were set to reasonable extreme values far from the default populational model, testing the search capabilities of the algorithms with physiologically-rare cases. The

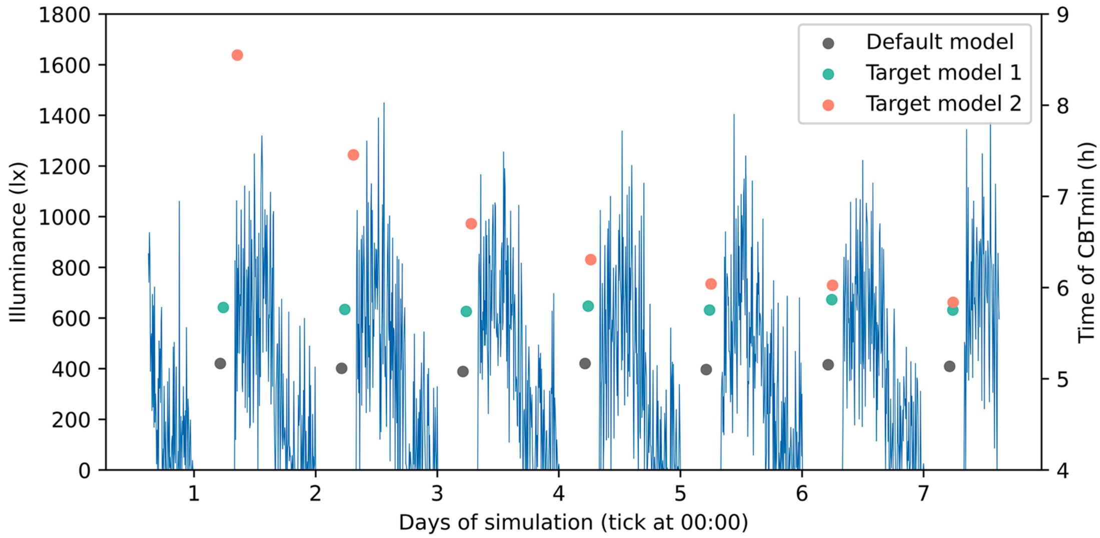


Fig. 1. Left y-axis: Illuminance (blue) used as the input for the models, consisting of 7 consecutive days of a modified realistic light profile proposed in [35]. Right y-axis: Time of CBTmin of default model (gray dots), target model 1 (light-green dots) and target model 2 (light-red dots). The default model and target model 1 were previously entrained to the illuminance profile so their time of CBTmin were stable starting from the beginning of the simulation. Target model 2 was not previously entrained, and therefore great phase advances were observed throughout the simulation. Simulation starts and ends at 15:00 h.

internal circadian period was set to $\tau = 23.8$ h, representing a rather extreme, but still frequent period at the population level [32]; the light sensitivity was set to $p = 0.25$, representing those individuals with lower, but still functional circadian response to light (considering the great variability in human sensitivity to light [33] and that setting it to values close to zero would represent a circadian-blind individual, and thus light would not have any entrainment effect); and the balance of the response curve was set to $k = 0$, representing an undefined mammal in terms of diurnality/nocturnality, based on the analysis in [15] and Aschoff's rule [34].

2.3. Light input and entrainment of the models

The input of the models consists of a realistic illuminance profile proposed in [35], with added Gaussian noise and a standard deviation of 300 lx (Eq.(1) and Fig. 1, left y-axis). Following the same procedure as in [6], complete darkness (0 lx) was set from 0:00 to 8:00. Negative illuminance values due to the Gaussian noise were also set to 0. The same light profile was applied for a period of seven consecutive days to all simulations, starting at 15:00.

$$I(t) = 40 + 330 \cdot \{\tanh[0.6 \cdot (t - 7.5)] - \tanh[0.6 \cdot (t - 16.5)]\} + N(0, 300)\ \ lx \tag{1}$$

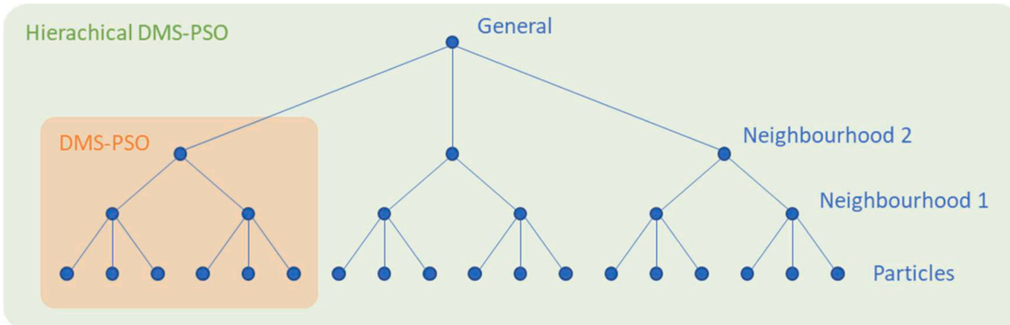
where $\boldsymbol{t}$ is the time of the day, in hours. To establish the initial conditions, two different cases were considered, depending on whether the models were previously entrained or unentrained to the light input (target model 1 and target model 2, respectively). The entrainment of default and target model 1 was performed by iteratively adjusting the initial conditions until a stable rhythm of CBTmin was achieved, which would happen when there was less than 60 s of difference in the CBTmin prediction for all days over the course of two consecutive weeks. From this stable state, initial conditions were obtained (default model: $x_0 = 1.069$, $y_0 = 0.119$, $n_0 = 0.731$; target model 1: $x_0 = 1.062$, $y_0 = 0.232$, $n_0 = 0.844$). Initial conditions for target model 2 were set to ($x_0 = 0.3$, $y_0 = 0.9$, $n_0 = 0.5$) to achieve a delay of several hours in its initial phase (e.g., shift work schedule or jet-lag). Once the parameters and initial conditions were defined for the three models (Table 1), the illuminance profile (blue line, left y-axis in Fig. 1) was fed to them. As a result, three different IVPs were set and integrated, and three arrays of Times of CBTmin were obtained as an output (gray, light-green and light-red dots in Fig. 1, right y-axis). These arrays represent the weekly trend of the internal phase of each model, which is stable throughout the week for the default and target model 1. The only difference between them arises from their different circadian parameters (τ, p, k), which cause the target model 1 to be phase-delayed by around 45 min with respect to the default. On the other hand, target model 2 starts the simulation with a marked phase delay. However, due to the entrainment effect of light, its internal phase becomes closer day by day to that of target model 1, to the extent that, by the end of the week, they are both almost in synchrony.

2.4. Fitness function and minimization problem

The fitness function $F(x_0, y_0, n_0, \tau, p, k)$ to be minimized by the PSO algorithms is defined as the average of the absolute difference between the array of daily times of CBTmin predicted by the optimizer and that of the target model, and it depends on the three circadian parameters (τ, p, k), as well as the three initial conditions (x_0, y_0, n_0) of the model, resulting in a 6-dimensional function to minimize. Parameters are constrained according to their biologically-coherent range (as proposed in [16]), and initial conditions are constrained according to the values of

Table 1
Default and target parameters of the models. The default model corresponds to the version published in [5]. The target models' parameters were modified to represent an individual with different physiology than the default model, either under a regular schedule (target model 1) or an irregular schedule (target model 2). Notice the very similar initial conditions of the two previously entrained models, despite the differences in their circadian parameters (τ, p, k).

Parameters	Default model (previously entrained)	Target model 1 (previously entrained)	Target model 2 (previously not entrained)
τ	24.2 h	23.8 h	23.8 h
p	0.5	0.25	0.25
k	0.55	0	0
x_0	1.069	1.062	0.3
y_0	0.119	0.232	0.9
n_0	0.731	0.844	0.5



Hierarchical DMS-PSO hyperparameters and pseudocode

Control hyperparameters

N: number of particles	M: max iterations
N_1: number of level 1 neighborhoods	N_2: number of level 2 neighborhoods
R_1: regrouping period of particles	R_2: regrouping period of N_1 neighborhoods.

Pseudocode

Start

1: Integrate target model using Eqs. (A.1) to (A.6).
2: Calculate and store target array of times of CBTmin using Eq. (A.7) and (A.8).
3: Initialize the position P and velocity V of all particles as specified in Section 2.5.1.
4: Group the swarm randomly in N_1 neighborhoods.
5: Group the N_1 neighborhoods randomly in N_2 neighborhoods.
6: **For** t = 1:M, **do:**
7:     Update inertia using Eq. (5).
8:     **For** j = 1:N, **do:**
9:         Integrate the model using Eqs. (A.1) to (A.6).
10:         Calculate array of times of CBTmin using Eq. (A.7) and (A.8).
11:         Evaluate Fitness [Eq. (2)].
12:         Check and update Pbest, if necessary.
13:         Check and update N_1best, if necessary.
14:         Update velocity and position of the particle using Eqs. (3) and (4).
15:     **If** remainder(t/R_1) = 0, **then:** Regroup particles randomly inside each N_2 neighborhood.
16:     **If** remainder(t/R_2) = 0, **then:** Regroup full N_1 neighborhoods randomly.
17:     **If** convergence is achieved, **then:** Go to step 18.
18: Set $N_1 = N_2 = 1$.
19: **For** t = 1:100, **do:**
20:     **For** j = 1:N, **do:**
21:         Integrate the model using Eqs. (A.1) to (A.6).
22:         Calculate array of times of CBTmin using Eqs. (A.7) and (A.8).
23:         Evaluate Fitness [Eq. (2)].
24:         Check and update Pbest, if necessary.

End

Fig. 2. Structure (top) and hyperparameters and algorithm (bottom) of the Hierarchical Dynamic Multi-Swarm PSO (H-DMSPSO), a generalization of the DMS-PSO [30]. Steps 18–24 are optional, since the final solution may be controlled through different convergence criteria and maximum number of iterations.

the state variables when an stable solution is achieved (see [4], for instance). The minimization problem is proposed as:

$$\left.\begin{array}{l} Minimize\ F(x_0, y_0, n_0, \tau, p, k) = \dfrac{\sum_{i=1}^{D} \left| TimeofCBTmin_{pred,i} - TimeofCBTmin_{targ,i} \right|}{D} \\ Subject\ to:\ -1.5 \leq x_0 \leq 1.5;\ -1.5 \leq y_0 \leq 1.5;\ 0 \leq n_0 \leq 1 \\ 23.4 \leq \tau \leq 25;\ 0 \leq p \leq 1;\ -1 \leq k \leq 1 \end{array}\right\} \quad (2)$$

where D is the total number of days with a CBTmin output.

2.5. Particle swarm optimizers

Particle swarm optimizers are stochastic optimization techniques that consist of a group of particles that simultaneously search for the solution space of a function, aiming to find its global minimum. Drawing inspiration from the social behavior of certain animals when searching for food sources, each member of the swarm (particle) is defined by its position $\boldsymbol{X}$ and velocity $\boldsymbol{V}$, and moves stochastically towards its own best position discovered so far ($\boldsymbol{P_{best}}$) as well as the best position discovered so far by the rest of the members of the swarm ($\boldsymbol{G_{best}}$), thus combining independent and collaborative searches. The stochastic nature of the algorithm is represented by the terms $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$, two random numbers in the range [0, 1] selected every iteration. The balance between independent and collaborative search is given by $\boldsymbol{c_1}$ and $\boldsymbol{c_2}$, and an inertia term $\boldsymbol{w}$ is added to maintain some momentum of the particle in its search direction. The velocity and position of each particle are then updated every iteration $\boldsymbol{t}$, as follows:

$$V(t+1) = w(t)\cdot V(t) + c_1\cdot r_1(t)\cdot[P_{best} - X(t)] + c_2\cdot r_2(t)\cdot[G_{best} - X(t)] \quad (3)$$

$$X(t+1) = X(t) + V(t+1) \quad (4)$$

In our study, the ability of different PSOs to find the parameters of the two target models specified in Table 1 is evaluated by optimizing the minimization problem Eq. (2)). That is, given only the output of the models (array of times of CBTmin) as a target, the algorithms should uncover the underlying parameters (τ, p, k) and initial conditions (x_0, y_0, n_0) that generate that output. The algorithms evaluated were the Canonical PSO [29], a Dynamic Multi-Swarm PSO (DMS-PSO) [30] in which the swarm is divided into N_1 sub-swarms of P particles each, and a modification of the latter, namely Hierarchical Dynamic Multi-Swarm PSO (H-DMSPSO). This proposed modification generalizes the concept of sub-swarms to a hierarchical structure, grouping the N_1 neighborhoods into higher-ranked N_2 neighborhoods (Fig. 2). Particle velocities and positions are then updated according to Eqs. (3) and ((4), but in which for each N_2 neighborhood, $G_{best} = N_2 = N_{1,best}$; the collaborative term then depends on the best-known position within its N_2 neighborhood and *not* on the best-known position of the swarm as a whole. Thus, N_2 neighborhoods search the solution space completely independently from one another until they are allowed to exchange information, which happens every R_2 generations when all the N_1 neighborhoods are randomly shuffled and regrouped. As in DMS-PSO, once a convergence criterion is satisfied or a maximum number of iterations is reached, a global PSO search ($N_1 = N_2 = 1$) can be performed so that all particles reunite and try to further improve upon the best solution found so far.

2.5.1. Particles initialization and encoding strategy

Initial sampling functions used for all parameters and initial conditions are specified in Table 2. Circadian parameters were sampled from a uniform distribution in their respective biologically-coherent range (as proposed in [15]) for all target models. However, to sample initial conditions, two different approaches were followed, depending on the pre-entrainment of the model. If we assume a regular schedule, the initial phase of a model will be very close to the initial phase of the entrained default model, and thus sampling from a normal distribution centered on the latter would help the algorithm to find the real initial phase of the former. Under irregular schedules, however, an insufficiently-covered initial search space could lead to rapid convergence on local minima, yielding suboptimal results. Therefore, in case of target model 1, initial conditions were sampled from a normal distribution, with the mean equal to the initial conditions of the default model and with a standard deviation equal to 0.1, whereas in case of target model 2, they were sampled from a uniform distribution in the range [-1, 1] for x_0 and y_0, and [0, 1] for n_0. Initial velocities were set to 0 for all particles.

Considering that particles are defined by a tuple of 6 real numbers and that the search space is continuous, the encoding strategy chosen was to keep the values of the particles, directly representing their positions on the search space with the same real numbers that define them (real value encoding). Real value encoding is more computationally expensive, since it requires floating-point arithmetic operations. Thus,

Table 2
Probability distributions used to sample the initial position of the particles. *U(a, b)* denotes a uniform distribution in the range [a, b]; *N*(μ, σ) denotes a normal distribution with mean μ and standard deviation σ.

Parameters	Target model 1 (previously entrained)	Target model 2 (not previously entrained)
τ	$\sim U(23.4,\ 25)$	
p	$\sim U(0,\ 1)$	
k	$\sim U(-1,\ 1)$	
x_0	$\sim N(1.069,\ 0.1)$	$\sim U(-1,\ 1)$
y_0	$\sim N(0.119,\ 0.1)$	$\sim U(-1,\ 1)$
n_0	$\sim N(0.731,\ 0.1)$	$\sim U(0,\ 1)$

Table 3
Swarm structures evaluated. Each algorithm underwent a 10-fold test from 30 to 480 particles, varying the structure of the swarm accordingly. Notice the absence of any grouping in the Canonical PSO ($N_2 = N_1 = 1$) and the absence of N_2 neighborhood groupings in the DMS-PSO ($N_2 = 1$).

	Swarm structure (N_2 x N_1 x P) for N particles				
	$N = 30$	$N = 60$	$N = 120$	$N = 240$	$N = 480$
Canonical PSO	1 × 1 × 30	1 × 1 × 60	1 × 1 × 120	1 × 1 × 240	1 × 1 × 480
DMS-PSO	1 × 10 × 3	1 × 20 × 3	1 × 40 × 3	1 × 80 × 3	1 × 160 × 3
H-DMSPSO 5x	5 × 2 × 3	5 × 4 × 3	5 × 8 × 3	5 × 16 × 3	5 × 32 × 3
H-DMSPSO 10x	10 × 1 × 3	10 × 2 × 3	10 × 4 × 3	10 × 8 × 3	10 × 16 × 3

the computational bottleneck in this particular problem lies in integrating the system of the ODE, which takes up around 99.94% of the total computation time (Table B.1). Moreover, real-value encoding is simpler to implement in this particular case, and any intermediate encoding/decoding step is also avoided.

2.5.2. Algorithm parameters and inertia

For the purposes of this study, two different N_2 groupings were tested ($N_2 = 5$ and $N_2 = 10$). As recommended in [30], particles were grouped into threes ($P = 3$) and N_1 neighborhoods were regrouped every 5 iterations ($R_1 = 5$), leaving the regrouping period of the N_2 neighborhoods as $R_2 = 40$, as we empirically found to be a good compromise

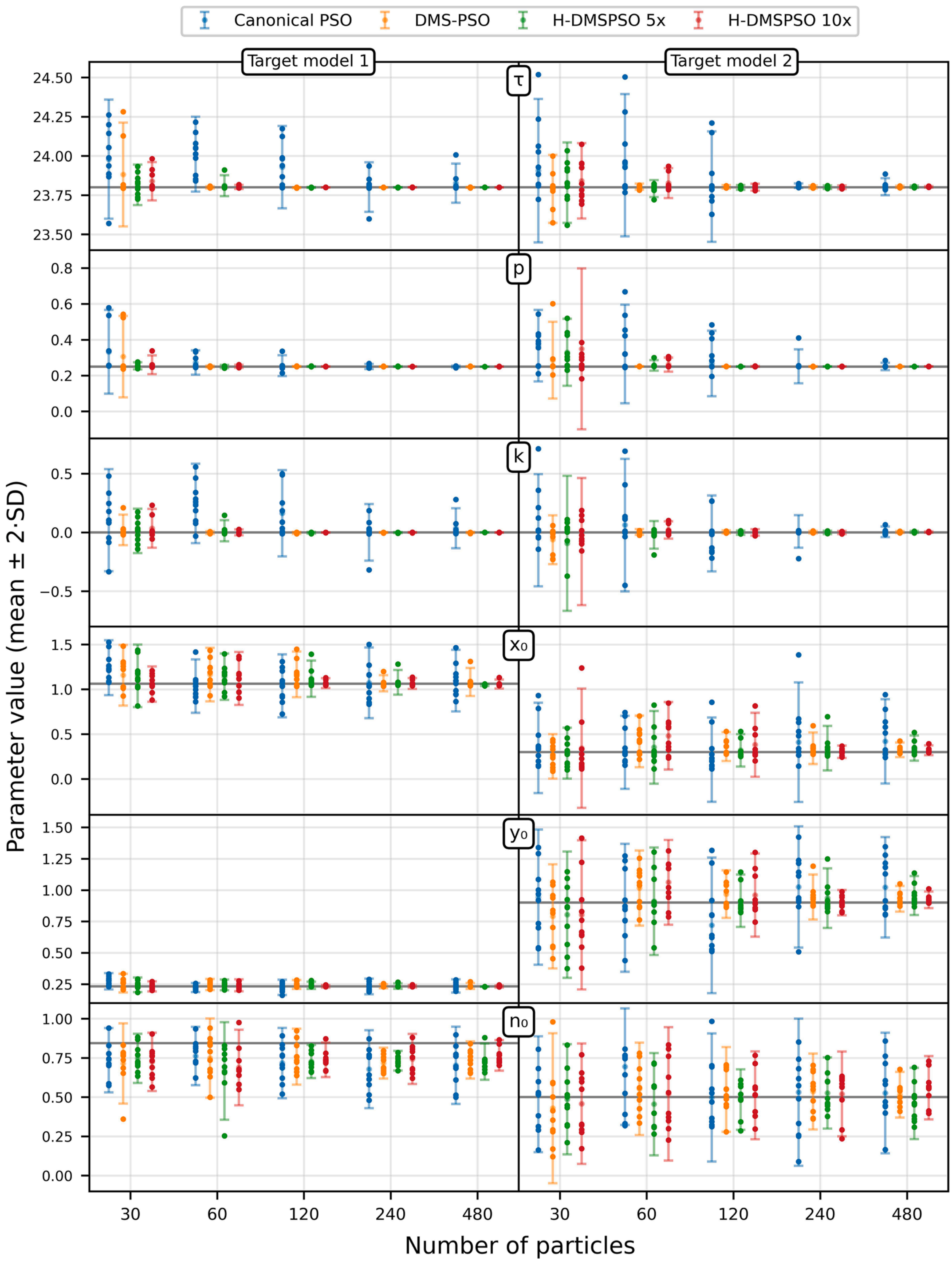


Fig. 3. Uncovering of the three circadian parameters (τ, p, k) and the three initial conditions (x_0, y_0, n_0) for all algorithms over ten runs, represented as single points and parameter mean ± 2·SD intervals. Target values are plotted as horizontal gray lines. Algorithms are colored by type (blue: Canonical PSO, orange: DMS-PSO, green: H-DMSPSO 5x, red: H-DMSPSO 10x) and grouped by number of particles. Left: target model 1; right: target model 2. DMS-PSO: Dynamic Multi-Swarm PSO; H-DMSPSO: Hierarchical Dynamic Multi-Swarm PSO.

between mixing and the independent exploration behaviors of the N_2 neighborhoods. Five different swarm structures per algorithm were evaluated, using 30, 60, 120, 240 and 480 particles (Table 3). A linearly declining inertia was set, limited by a minimum value w_{min}:

$$w(t) = \max(w_0 - t \cdot \Delta w, w_{\min}) \tag{5}$$

Algorithm parameters were set to $c_1 = c_2 = 1.49445$, $w_0 = 0.9$, $w_{min} = 0.1$ and $\Delta w = 0.002$. Moreover, particle positions were restricted to their biologically-coherent ranges (see Eq. (2) and Section 2.5.1) and velocities were restricted to a maximum of 5% of that same range. Each structure was run 10 times upon convergence, which was considered to have been achieved when the best solution found so far was not updated for 100 iterations.

3. Code, calculation and statistical tests

PSO algorithms, circadian models, numerical solvers, light profile data, entrainment of models, simulations, analysis of results and plotting were coded entirely in Python v3.10.4. In addition to the standard libraries, third-party libraries were used, including: numpy v1.22.3, pandas v1.5.1, matplotlib v3.5.2, numba v0.45.4, numbalsoda v0.3.5, scipy v1.8.0 and all their dependencies. All simulations were run in a 64-bit Windows®10 PC, equipped with an Intel® Core™ i5–11400 processor and 32 GB RAM. Pairwise Dunn tests were performed for fitness error and mean absolute errors (MAE) of the parameters and initial conditions, between algorithms grouped by number of particles used. Additionally, algorithms were sorted into significance groups (p-value < 0.01) by means of compact letter display analysis, calculated following the algorithm in [36].

3. Results and discussion

3.1. Revealing personal circadian parameters

The revealing of circadian parameters through the use of algorithms is shown in Fig. 3. Canonical PSO is unable to confidently reveal all three internal circadian parameters (τ, p, k), even with a $1 \times 1 \times 480$ structure of 480 particles, in either of the two target models. The concept of dividing the swarm into smaller neighborhoods, first introduced in 1999 [37], is a common strategy to improve the search behavior of the swarm. In concordance, Dunn pairwise tests show significant differences (p-value < 0.01) in fitness error levels and MAE between Canonical PSO and Multi-Swarm PSO algorithms (Figs. 4 and 5) for almost all groupings, and the circadian parameters can be successfully optimized using any of the latter. For example, using 120 or more particles, H-DMSPSO 10x algorithms achieve MAE below 0.001 for target model 1 and DMS-PSO or H-DMSPSO 5x achieve MAE below 0.01 for target model 2 for all three parameters simultaneously (Supplementary Table 1). Indeed, convergence plots (Fig. 6) show that Canonical PSO tends to converge earlier, having greater chances of falling into local minima than their grouped counterparts. To the extent of our knowledge, the particle swarm methodology has never been applied to optimize the parameters of a human central pacemaker, but the original PSO [26] algorithm has been successfully applied to optimize the parameters of a model of the transcriptional regulation of the rice circadian clock, composed also by a system of ODE [38] and to accurately compute parameters in an arterial blood flow model [25]. In view of the results presented here, Dynamic Multi-Swarm algorithms may prove to be a better approach to the problem of optimizing parameters in the biomedical field, considering that the complexity of the algorithm is not much greater. In this regard, Dunn pairwise tests show that in general there are no significant differences (p-value < 0.01) among DMS-PSO, H-DMSPSO 5x and H-DMSPSO 10x, except in some particular parameters, in which the algorithm overlaps the Canonical PSO.

3.2. Optimum number of particles and computational cost

Log-Log plots of MAE as a function of computational cost (CC, represented as number of function evaluations) for each parameter are shown in Fig. 7. An exponential relationship can be derived from them (dashed gray lines), in the form of:

$$MAE = a \cdot CC^b \leftrightarrow \log_{10} MAE = \log_{10} a + b \cdot \log_{10} CC \tag{6}$$

where a and b are constants. This log-log linear relationship and the negative slopes achieved in all regressions show the robustness of the algorithms (more computational power leads to lower errors). In principle, the results show that there is no clear optimum number of particles to use, and it is up to the user to decide, considering the desired error and the computational cost.

In this particular problem, the computational bottleneck lies in integrating the system of ODE and not in the algorithms themselves. For a 1-week simulation, the integration of the ODE can take up to 99.94% of the total computational time (Table B.1). Thus, the computational effort depends essentially on the number of times the ODE system is integrated. Different algorithms can then be compared according to the number of integrations they need to reach accurate solutions. In the

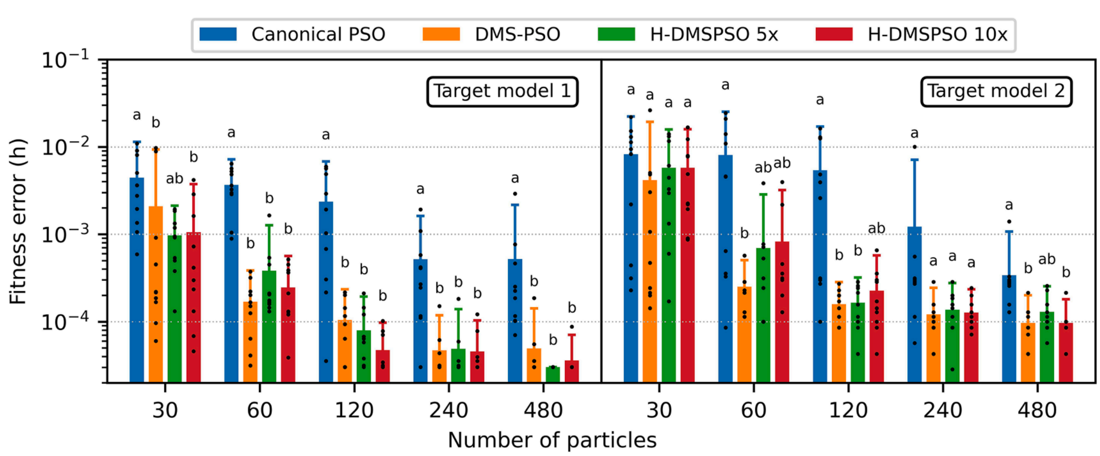


Fig. 4. Average fitness error (mean + 2·SD hours) achieved by all algorithms over 10 runs. Algorithms are colored by type (blue: Canonical PSO, orange: DMS-PSO, green: H-DMSPSO 5x, red: H-DMSPSO 10x) and grouped by number of particles. Pairwise Dunn tests were performed between algorithms grouped according to number of particles. Different letters above bars indicate significant differences between algorithms (p-value < 0.01) grouped by number of particles, and were calculated following the Compact Letter Display classification algorithm in [36]. Left: target model 1; right: target model 2. DMS-PSO: Dynamic Multi-Swarm PSO; H-DMSPSO: Hierarchical Dynamic Multi-Swarm PSO.

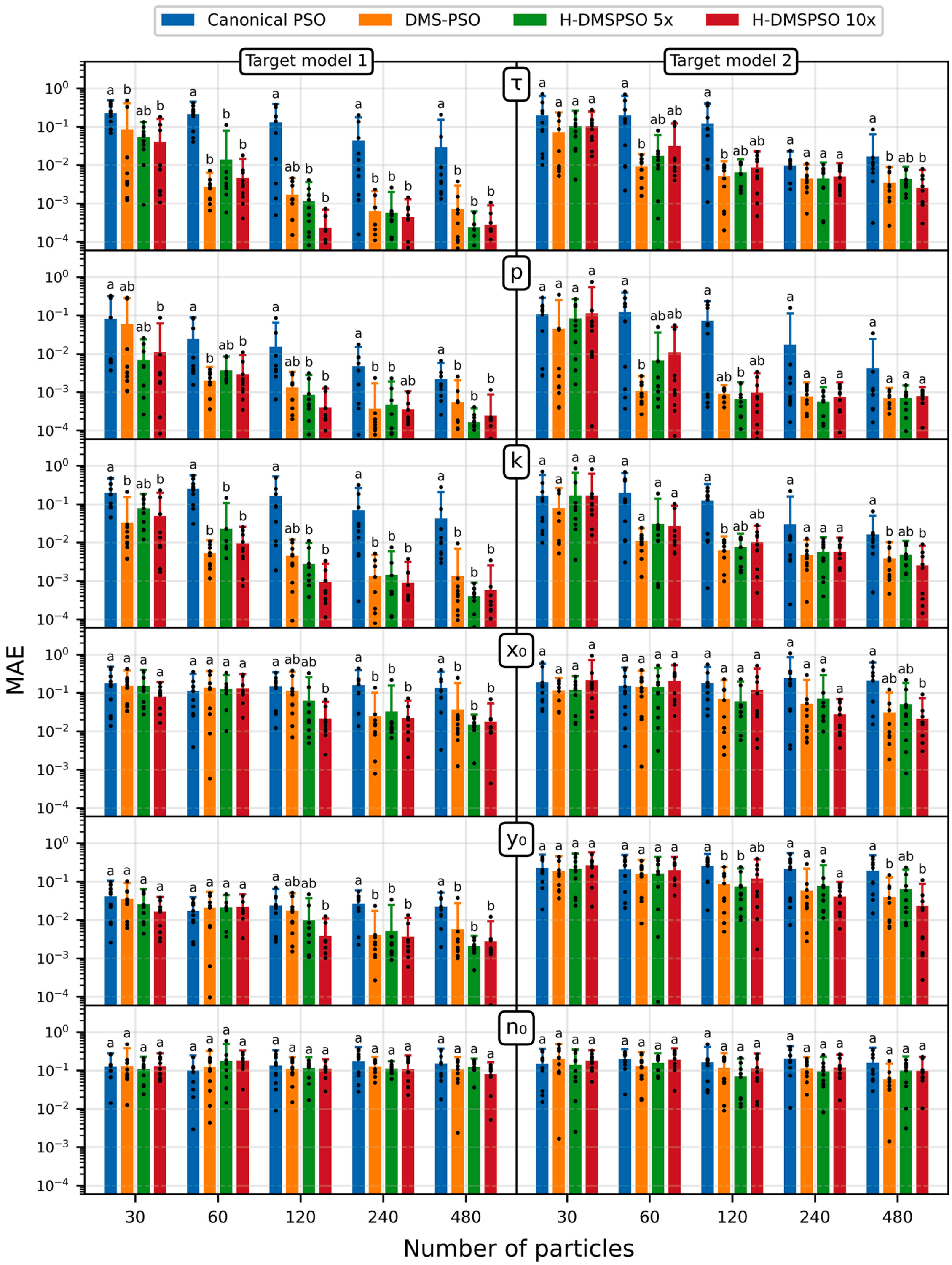


Fig. 5. Mean absolute error (MAE)+2·SD of the three circadian parameters (τ, p, k) and the three initial conditions (x_0, y_0, n_0) for all algorithms over ten runs. MAE is calculated as the mean difference between the final value achieved by the algorithm and the target value over the ten runs. Algorithms are colored by type (blue: Canonical PSO, orange: DMS-PSO, green: H-DMSPSO 5x, red: H-DMSPSO 10x) and grouped by number of particles. Pairwise Dunn tests were performed for each parameter between algorithms grouped by number of particles. Different letters above bars indicate significant differences (*p*-value < 0.01) for each parameter between algorithms grouped by number of particles, and were calculated following the Compact Letter Display classification algorithm in [36]. Left: target model 1; right: target model 2. DMS-PSO: Dynamic Multi-Swarm PSO; H-DMSPSO: Hierarchical Dynamic Multi-Swarm PSO.

work by Stone et al. [15], only 3 variables were optimized and therefore, a brute-force search could be applied by splitting the range of each parameter into segments with a width of 0.1 and solving the problem at each point of the resulting grid. The number of integrations performed (13 possible values for τ, 11 for p and 21 for k) added up to roughly $3 \cdot 10^3$ solutions. Applying this brute-force approach to 3 parameters + 3 initial conditions (our problem), with 20 possible values per parameter (precision of approximately 0.05–0.1 per parameter), would lead to $20^6 \approx 6.4 \cdot 10^7$ integrations of the ODE system, which is two to three orders of magnitude higher than the results achieved by the PSO algorithms

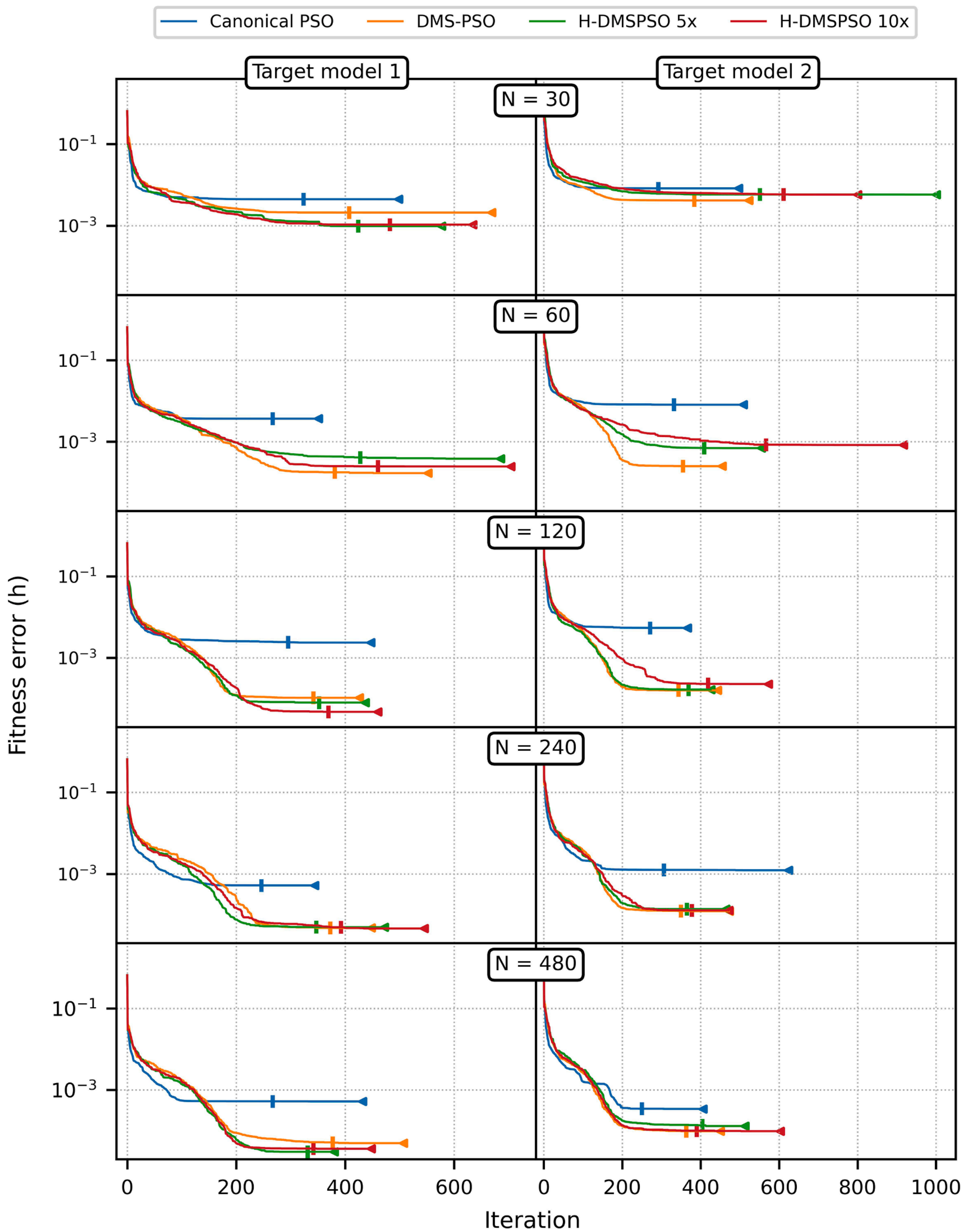


Fig. 6. Average convergence plots for all algorithms over ten runs. Triangles indicate the maximum iterations needed to converge by any of the ten runs; vertical bars indicate the average iterations needed over ten runs. Plots are divided by number of particles of the algorithms, from $N = 30$ (top) to $N = 480$ (bottom). Algorithms are colored by type (blue: Canonical PSO, orange: DMS-PSO, green: H-DMSPSO 5x, red: H-DMSPSO 10x). Left: target model 1; right: target model 2. DMS-PSO: Dynamic Multi-Swarm PSO; H-DMSPSO: Hierarchical Dynamic Multi-Swarm PSO.

and with a much lower precision. This analysis denotes the *curse of dimensionality* that the brute-force approach usually suffers from, which makes it impractical for solving the problem posed here. Another procedure was followed in the work of Bonarius et al. [6], in which a Bayesian particle filter was applied to the Kronauer + Phillips-Robinson combined model. Although both particle filters and PSO algorithms rely on a group of particles to simultaneously evaluate the state of the system throughout the solution space, they solve the problem from very different approaches, and it is therefore difficult to compare them. Indeed, the proposed Bayesian approach first transforms the deterministic ODE system into a stochastic one, and then the particle filter is applied to the noisy evolution of the system, sampling dynamically every time a partial empirical observation is available. Only one parameter (τ) is optimized along the integration. Conversely, the PSO approach maintains the original deterministic nature of the ODE, and thus the integration of the ODE system is performed in one go, but it needs to be done iteratively until convergence. Moreover, with the former algorithm, initial conditions need to be defined as prior probability distributions and therefore cannot be optimized. In terms of computational efficiency, the particle filter included a total of 7 free

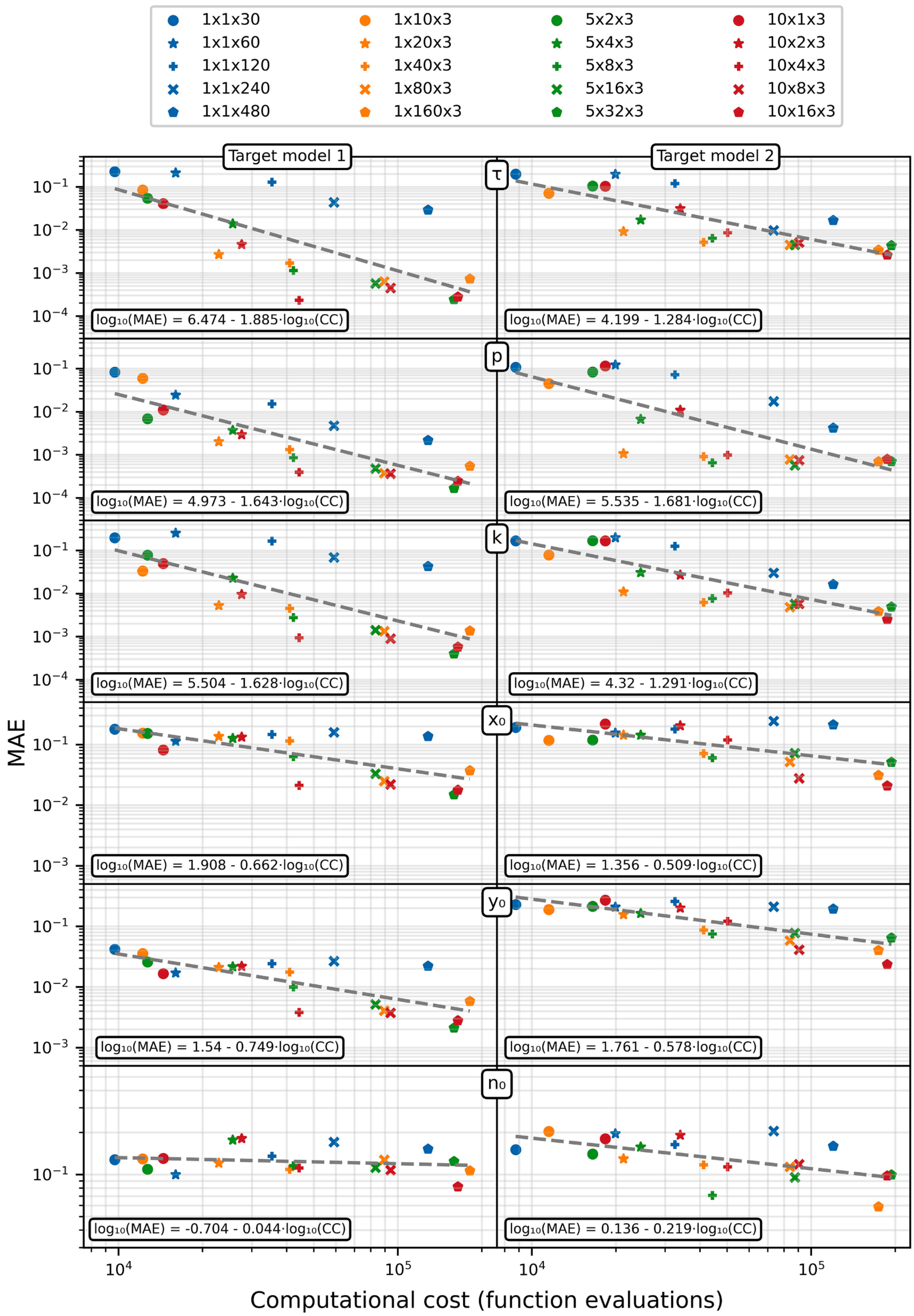


Fig. 7. Mean absolute error (MAE) versus mean computational cost of the three circadian parameters (τ, p, k) and the three initial conditions (x_0, y_0, n_0) for all structures over ten runs. The computational cost (function evaluations) is defined as the product of the number of particles of the structure and the number of iterations needed to converge (Function evaluations = N · Iterations). Log-Log linear regressions are plotted as dashed gray lines, and correspondent regression equations are shown inside each subplot. Structures are colored by algorithm type (blue: Canonical PSO, orange: DMS-PSO, green: H-DMSPSO 5x, red: H-DMSPSO 10x). DMS-PSO: Dynamic Multi-Swarm PSO; H-DMS-PSO: Hierarchical Dynamic Multi-Swarm PSO. See graphic legend for symbol details.

variables to optimize (6 state variables + parameter τ) and required 800 particles to achieve confident results. The algorithm was based on a previous work by Mott et al. [10], who suggested that 240 particles were sufficient to estimate 3 variables. According to the authors, $240^{7/3} \approx 3.6 \cdot 10^5$ particles would have been needed to optimize 7 variables, but in reality, little effort was required to estimate the state variables and thus most of it seemed to be dedicated to optimizing τ. Without optimizing initial conditions, for 3 personal circadian variables plus 3 state variables, $240^{6/3} \approx 5.7 \cdot 10^4$ particles would be needed at most, which would in principle be equivalent to using any of the Dynamic Multi-Swarm PSO algorithms and 120 to 240 particles. Therefore, Dynamic Multi-Swarm PSO algorithms may almost certainly suit even higher-dimensional optimizations of ODE-based circadian and/or sleep regulation model. Besides, the brute-force method results unfeasible in order to solve the 6-dimensional problem posed here, whereas PSO algorithms can successfully address it.

3.3. Initial conditions optimization

In all the structures tested, the initial conditions show greater MAE than their respective circadian parameters. The initial phase of the model, determined by x_0 and y_0, is more easily optimized than the initial depletion level of the photoreceptors (n_0), but still, none of the algorithms is able to achieve MAE below 0.01 for both of them simultaneously. The less acute slopes of initial conditions with respect to circadian parameters on the regression lines in Fig. 7 also confirm that the former ones are harder to optimize, in both models. Since the simulation starts at 15:00 and the model is being entrained along it, small deviations in its initial estimated phase have little influence on the results, because the model has enough time to align to the light profile before the time of the first CBTmin. Thus, the closer the start of the simulation is to the time of the first CBTmin, the greater impact is found on the final error achieved. In consequence, if the simulation is long enough (i.e., >10 days), it may be reasonable to make a rough estimate of the initial phase and ignore the first point of CBTmin yielded, which would reduce the computational load. Indeed, in the work of Stone et al. [15] on 12 participants, rough estimations of the initial conditions allowed them to optimize the same three parameters (τ, p, k) by brute-force search. Results showed that after the continuous recording of a 3-week period, the MAE of the default model was greatly improved from 1.02 h to 0.28 h In this particular case, although the impact of estimating the initial conditions was not assessed, the results shown here indicate that are the errors due to that estimation would be negligible, because the circadian phase was assessed after each week. However, misestimating the initial phase could also be an important source of error. In the study by Rea et al. [13], Kronauer's model was initialized assuming an initial time of CBTmin = 04:00 across all four datasets tested, but simply optimizing it in the range of 03:00–09:00 in increments of 1 h led to an overall decrease in MAE from 0.79 to 0.61 h Moreover, more extreme chronotypes would probably lead to greater errors. For example, for a particular dataset of night-owl young adults [39] analyzed in that same study, the optimum initial time of CBTmin found was 09:00 and MAE decreased from 1.21 to 0.59 h Thus, results suggest that optimization of initial conditions and/or assessment of the impact of misestimating them for each particular case is recommended, but they may not need to be so precisely optimized as circadian parameters. Further research is needed in this direction.

3.4. Optimization under a regular/irregular schedule

As expected, optimization is easier and algorithms achieve lower fitness errors and lower MAE of the circadian parameters (τ, p, k) under regular as opposed to irregular schedules. Since the only difference between the two target models is that the algorithm's starting point in target model 1 is closer to the actual circadian phase of the model, being able to estimate it has a significant impact on estimating the circadian parameters. However, the MAE achieved by Dynamic Multi-Swarm PSO algorithms are in the range of 10^{-2}–10^{-3} for target model 2 when using 120 particles or more, and negative slopes in regression lines in Fig. 7 suggest that given enough computational power, the algorithm is indeed robust enough to locate the global minimum for individuals under irregular schedules. A review on individual circadian phase prediction [40] found that only the oscillator model used in this work has been successfully validated under more challenging conditions like rotating night-shift workers [14], yielding better results than neural networks or statistical approaches, which supports the direct application of Dynamic Multi-Swarm PSO algorithms to uncover personal circadian parameters involved in the human response to light under either regular or irregular schedules.

3.5. Limitations and applicability

Integrating the system of ODE requires a significant computational effort (Appendix B), and thus fine optimization of all the hyperparameters of the algorithm has not been considered in this work. Moreover, exhaustive review, testing and optimization of other strategies to split the swarm into sub-populations (see [28], Section 4.1), as well as other heuristic and evolutionary algorithms [21] or other optimization techniques could also improve the results presented. Considering the robustness of the Multi-Swarm strategy and the order of magnitude of the error yielded ($< 10^{-3}$ h), we believe that its application to empirical recordings would not add any significant error to the entire process of optimizing personal circadian parameters, which mostly depends on the empirical measurement of individual circadian phases. Currently, even when using dim light melatonin onset time as the phase marker, the best prediction errors yielded by the models are in the order of a magnitude of 0.1–1 h [40], much higher than the errors due to optimizers.

4. Conclusions

In this study, we have demonstrated that Dynamic Multi-Swarm PSO algorithms can accurately and simultaneously uncover three personal circadian parameters and three initial conditions of Kronauer's oscillator model, and may prove to be a more efficient approach than Global PSO algorithms in the biomedical field. Circadian parameters play the most important role in the applicability of the model and should be prioritized, but optimization of the initial conditions or assessment of the impact of misestimating them is recommended. Considering the low errors achieved by the algorithms and their robustness, we believe that their testing, validation and systematic application to empirical data (under regular or irregular schedules) is promising, and that uncovering personal circadian parameters could shed some light into what other factors (genetic, environmental, or others) may lie behind the interindividual variability in the circadian response to light.

Declaration of Competing Interest

M.A.R. and J.A.M. are founding partners of Kronohealth, S.L., a spin-off company, also participated by and co-founded by the University of Murcia. Kronohealth has not contributed to financing this study nor will obtain commercial profit from them. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Acknowledgments

We would like to thank Salvador Bará for his greatly appreciated comments on a previous version of this manuscript. This paper has been funded by the Ministry of Economy and Competitiveness, the Instituto de Salud Carlos III through a CIBERFES grant (CB16/10/00239) and

Diabfrail LatAm (European Union Horizon 2020 research and innovation program No. 825546) awarded to MAR (all co-financed by FEDER). Grant RTI2018-093528-B-I00, funded by MCIN/AEI/ 10.13039/501100011033 and by "ERDF A way of making Europe", by the "European Union" and by the "European Union NextGenerationEU/PRTR". JVM predoctoral contract has been funded by grant '21610/FPI/21'. Fundación Séneca. Región de Murcia (Spain).

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.cmpb.2023.107933.

Appendix A. Oscillator circadian model

The model used in this paper consists of a system of Ordinary Differential Equations (ODE), which describes the transformation of a light input received through the retina into a drive that reaches the human central pacemaker and modifies its state (phase and amplitude). Eq. (A.1) and (A.2) represent the physiological response of the retinal photoreceptors to the light stimulus $\boldsymbol{I}$. The activation $\boldsymbol{\alpha}$ of the photopigments is transformed into the drive $\widehat{\boldsymbol{B}}$ by the scaling factor **G**. The proportion $\boldsymbol{n}$ of ready-to-use photopigments in the retina is dynamically controlled by Eq. (A.3). This process is also controlled by parameters $\boldsymbol{p}$, $\boldsymbol{I_0}$, $\boldsymbol{\alpha_0}$ and $\boldsymbol{\beta}$. The drive $\widehat{\boldsymbol{B}}$ is subsequently modulated (Eq. (A.4)), depending on the state of the oscillator. This modulated drive $\boldsymbol{B}$ reaches the central pacemaker (Eq. (A.5) and (A.6)), the phase and amplitude of which are defined by the state variables $\boldsymbol{x}$ and $\boldsymbol{y}$. The dynamic properties of the oscillator also depend on the parameters **τ, μ, k** and **q**. The default values of the parameters are taken from the revised version of the model [5] and are specified in Table A.1. A more detailed explanation can be found in [41] or in the original papers [2–5].

$$\alpha = \alpha_0 \cdot \left(\frac{I}{I_0}\right)^p \cdot \left(\frac{I}{I+100}\right) \tag{A.1}$$

$$\widehat{B} = G \cdot \alpha(1-n) \tag{A.2}$$

$$\dot{n} = 60 \cdot [\alpha(1-n) - \beta n] \tag{A.3}$$

$$B = \widehat{B} \cdot (1-0.4x)(1-0.4y) \tag{A.4}$$

$$\dot{x} = \frac{\pi}{12} \cdot \left[y + \mu\left(\frac{1}{3}x + \frac{4}{3}x^3 - \frac{256}{105}x^7\right) + B\right] \tag{A.5}$$

$$\dot{y} = \frac{\pi}{12} \cdot \left\{qBy - \left[\left(\frac{24}{0.99729\tau}\right)^2 + kB\right] \cdot x\right\} \tag{A.6}$$

Once a light profile $\boldsymbol{I}$ and initial conditions of the state variables $(\boldsymbol{x_0}, \boldsymbol{y_0}, \boldsymbol{n_0})$ have been defined, the initial value problem can be solved. The output consists of the time at which the minimum core body temperature (CBTmin) occurs each day:

$$Time\ of\ CBTmin = Time\ of\left(\varphi_{yx}\right) + \varphi_c \tag{A.7}$$

where $\boldsymbol{Timeof\varphi_{yx}}$is the phase angle between the state variables $\boldsymbol{x}$ and $\boldsymbol{y}$ such that:

$$Time\ \ of\ \ \varphi_{yx} = Time\ \ at\ \ which\ \ \left[arctan\left(\frac{y}{x}\right) = -170.7^o\right] \tag{A.8}$$

and $\boldsymbol{\varphi_c}$ is a constant. Thus, for a given light profile, the integration and solving of Eqs. (A.1)–(A.8) will output an array of times whose length equals the number of days that are comprised. This array is used in Eq. (2) to evaluate the fitness of the solution predicted by the PSO algorithms, by comparing it to the target solution.

Table A.1
Default values of the model parameters used in this paper, as specified in [5].

Parameter	Parameter value
α_0 (min^{-1})	0.1
I_0	9500
p	0.5
G	37
β (min^{-1})	0.007
τ (h)	24.2
μ	0.13
Q	1/3
k	0.55
φ_c (h)	0.97

Appendix B. Computational cost of integrating the system of ODE

Table B.1 shows that the computational bottleneck of the algorithms lies in the time dedicated by the machine to solve the system of ODE. Even though only this integration takes advantage of the optimized numba/numbalsoda libraries (the rest of the code is run in a not-compiled python interpreter), the time spent by the ODE solver accounts for 99.94% of the total computational time. Total duration refers to the time spent by the machine solving all the steps of the algorithm in each iteration (lines 7 to 17 in the Pseudocode in Fig. 2), and ODE solver duration refers to the time dedicated to integrate the system of ODE (line 9 in the Pseudocode in Fig. 2). All simulations were run on a 64-bit Windows®10 PC equipped with an Intel® Core™ i5–11400 processor and 32 GB RAM.

Table B.1
Total and ODE solver durations per iteration of all algorithms, over 100 iterations. Total duration refers to the time spent by the machine solving all the steps of the algorithm in each iteration (lines 7 to 17 in the Pseudocode in Fig. 2); ODE solver duration refers to the time dedicated to integrating the system of ODE (line 9 in the Pseudocode in Fig. 2). N: number of particles of the algorithm; SD: standard deviation. DMSPSO: Dynamic Multi-Swarm PSO; H-DMSPSO: Hierarchical Dynamic Multi-Swarm PSO.

N	Algorithm	Total duration per iteration (s)		ODE solver duration per iteration (s)		ODE solver percentage of total duration
		Mean	SD	Mean	SD	
30	Canonical PSO	18.3931	0.2889	18.3820	0.2888	99.9394
	DMSPSO	18.0167	0.1007	18.0057	0.1008	99.9389
	H-DMSPSO 5x	18.0040	0.0872	17.9929	0.0873	99.9383
	H-DMSPSO 10x	18.0133	0.0985	18.0022	0.0985	99.9387
60	Canonical PSO	30.1261	0.8574	30.1086	0.8572	99.9419
	DMSPSO	29.3352	0.3017	29.3179	0.3017	99.9411
	H-DMSPSO 5x	29.2518	0.1743	29.2345	0.1743	99.9409
	H-DMSPSO 10x	29.2188	0.1072	29.2015	0.1072	99.9407
120	Canonical PSO	52.6465	0.8061	52.6165	0.8061	99.9429
	DMSPSO	51.8565	0.1581	51.8266	0.1580	99.9423
	H-DMSPSO 5x	51.8577	0.1794	51.8278	0.1796	99.9424
	H-DMSPSO 10x	51.8808	0.1654	51.8511	0.1654	99.9426
240	Canonical PSO	97.0401	0.5674	96.9852	0.5674	99.9434
	DMSPSO	97.0550	0.6578	96.9999	0.6579	99.9432
	H-DMSPSO 5x	97.1122	0.5410	97.0573	0.5412	99.9434
	H-DMSPSO 10x	96.9299	0.2092	96.8753	0.2092	99.9436
480	Canonical PSO	187.4386	0.9182	187.3343	0.9182	99.9443
	DMSPSO	187.5196	1.5754	187.4145	1.5751	99.9439
	H-DMSPSO 5x	186.8407	0.4194	186.7357	0.4189	99.9438
	H-DMSPSO 10x	186.8992	0.5551	186.7942	0.5551	99.9438

References

[1] C.A. Czeisler, The effect of light on the human circadian pacemaker, Ciba Found. Symp. 183 (1995) 254–290, https://doi.org/10.1002/9780470514597.ch14, discussion 290-302.

[2] R.E. Kronauer, A quantitative model for the effects of light on the amplitude and phase of the deep circadian pacemaker, based on human data, in: Proceedings of the 10th European Congress on Sleep Research, Düsseldorf, Pontenagel Press, 1990, p. 306.

[3] M.E. Jewett, R.E. Kronauer, Refinement of limit cycle oscillator model of the effects of light on the human circadian pacemaker, J. Theor. Biol. 192 (1998) 455–465, https://doi.org/10.1006/jtbi.1998.0667.

[4] M.E. Jewett, D.B. Forger, R.E. Kronauer, Revised limit cycle oscillator model of human circadian pacemaker, J. Biol. Rhythms 14 (1999) 493–500, https://doi.org/10.1177/074873049901400608.

[5] M.A. St. Hilaire, E.B. Klerman, S.B.S. Khalsa, K.P. Wright, C.A. Czeisler, R. E Kronauer, Addition of a non-photic component to a light-based mathematical model of the human circadian pacemaker, J. Theor. Biol. 247 (2007) 583–599, https://doi.org/10.1016/j.jtbi.2007.04.001.

[6] J. Bonarius, C. Papatsimpa, J.P. Linnartz, Parameter estimation in a model of the human circadian pacemaker using a particle filter, IEEE Trans. Biomed. Eng. 68 (2021) 1305–1316, https://doi.org/10.1109/TBME.2020.3026538.

[7] J.L. Creaser, C.O. Diekman, K.C.A. Wedgwood, Entrainment dynamics organised by global manifolds in a circadian pacemaker model, Front. Appl. Math. Stat. 7 (2021), 703359, https://doi.org/10.3389/fams.2021.703359.

[8] C.O. Diekman, A. Bose, Reentrainment of the circadian pacemaker during jet lag: east-west asymmetry and the effects of north-south travel, J. Theor. Biol. 437 (2018) 261–285, https://doi.org/10.1016/j.jtbi.2017.10.002.

[9] P. Indic, D.B. Forger, M.A. St. Hilaire, D.A. Dean, E.N. Brown, R.E. Kronauer, E. B. Klerman, M.E Jewett, Comparison of amplitude recovery dynamics of two limit cycle oscillator models of the human circadian pacemaker, Chronobiol. Int. 22 (2005) 613–629, https://doi.org/10.1080/07420520500180371.

[10] C. Mott, G. Dumont, D.B. Boivin, D. Mollicone, Model-based human circadian phase estimation using a particle filter, IEEE Trans. Biomed. Eng. 58 (2011) 1325–1336, https://doi.org/10.1109/TBME.2011.2107321.

[11] C. Papatsimpa, J.P. Linnartz, Personalized circadian light: a digital siblings approach. Ambient Intelligence and Smart Environments, IOS Press, 2021. Bashir, E., Luštrek, M.ISBN 978-1-64368-186-3.

[12] C. Papatsimpa, L.J.M. Schlangen, K.C.H.J. Smolders, J.P.M.G. Linnartz, Y.A.W. de Kort, The interindividual variability of sleep timing and circadian phase in humans is influenced by daytime and evening light conditions, Sci. Rep. 11 (2021) 13709, https://doi.org/10.1038/s41598-021-92863-z.

[13] M.S. Rea, R. Nagare, A. Bierman, M.G. Figueiro, The circadian stimulus-oscillator model: improvements to Kronauer's model of the human circadian pacemaker, Front. Neurosci. 16 (2022), 965525, https://doi.org/10.3389/fnins.2022.965525.

[14] J.E. Stone, X.L. Aubert, H. Maass, A.J.K. Phillips, M. Magee, M.E. Howard, S. W. Lockley, S.M.W. Rajaratnam, T.L. Sletten, Application of a limit-cycle oscillator model for prediction of circadian phase in rotating night shift workers, Sci. Rep. 9 (2019) 11032, https://doi.org/10.1038/s41598-019-47290-6.

[15] J.E. Stone, E.M. McGlashan, N. Quin, K. Skinner, J.J. Stephenson, S.W. Cain, A.J. K. Phillips, The role of light sensitivity and intrinsic circadian period in predicting individual circadian timing, J. Biol. Rhythms 35 (2020) 628–640, https://doi.org/10.1177/0748730420962598.

[16] T. Woelders, D.G.M. Beersma, M.C.M. Gordijn, R.A. Hut, E.J. Wams, Daily light exposure patterns reveal phase and period of the human circadian clock, J. Biol. Rhythms 32 (2017) 274–286, https://doi.org/10.1177/0748730417696787.

[17] H. Liang, H. Wu, Parameter estimation for differential equation models using a framework of measurement error in regression models, J. Am. Stat. Assoc. 103 (2008) 1570–1583, https://doi.org/10.1198/016214508000000797.

[18] D. Brewer, M. Barenco, R. Callard, M. Hubank, J. Stark, Fitting ordinary differential equations to short time course data, Philos. Trans. R. Soc. Math. Phys. Eng. Sci. 366 (2008) 519–544, https://doi.org/10.1098/rsta.2007.2108.

[19] A.J.K. Phillips, P.A. Robinson, A quantitative model of sleep-wake dynamics based on the physiology of the brainstem ascending arousal system, J. Biol. Rhythms 22 (2007) 167–179, https://doi.org/10.1177/0748730406297512.

[20] D.B. Forger, M.E. Jewett, R.E. Kronauer, A simpler model of the human circadian pacemaker, J. Biol. Rhythms 14 (1999) 533–538, https://doi.org/10.1177/074873099129000867.

[21] O. Bozorg-Haddad, M. Solgi, H.A. Loaiciga, Meta-heuristic and evolutionary algorithms for engineering optimization. Wiley Series in Operations Research and Management Science, John Wiley & Sons, Hoboken, NJ, 2017. ISBN 978-1-119-38699-5.

[22] C.M. Lai, C.C. Chiu, Y.C. Shih, H.P. Huang, A hybrid feature selection algorithm using simplified swarm optimization for body fat prediction, Comput. Methods Programs Biomed. 226 (2022), 107183, https://doi.org/10.1016/j.cmpb.2022.107183.
[23] E. Pashaei, E. Pashaei, N. Aydin, Gene selection using hybrid binary black hole algorithm and modified binary particle swarm optimization, Genomics 111 (2019) 669–686, https://doi.org/10.1016/j.ygeno.2018.04.004.
[24] O. Shindi, J. Kanesan, G. Kendall, A. Ramanathan, The combined effect of optimal control and swarm intelligence on optimization of cancer chemotherapy, Comput. Methods Programs Biomed. 189 (2020), 105327, https://doi.org/10.1016/j.cmpb.2020.105327.
[25] Y. Aboelkassem, D. Savic, Particle swarm optimizer for arterial blood flow models, Comput. Methods Programs Biomed. 201 (2021), 105933, https://doi.org/10.1016/j.cmpb.2021.105933.
[26] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, Proceedings of the MHS'95, in: Proceedings of the 6th International Symposium on Micro Machine and Human Science, Nagoya, Japan, IEEE, 1995, pp. 39–43.
[27] A.G. Gad, Particle swarm optimization algorithm and its applications: a systematic review, Arch. Comput. Methods Eng. 29 (2022) 2531–2561, https://doi.org/10.1007/s11831-021-09694-4.
[28] D. Wang, D. Tan, L. Liu, Particle swarm optimization algorithm: an overview, Soft Comput. 22 (2018) 387–408, https://doi.org/10.1007/s00500-016-2474-6.
[29] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: Proceedings of the IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), Anchorage, AK, USA, IEEE, 1998, pp. 69–73.
[30] W.D. Chang, A modified particle swarm optimization with multiple subpopulations for multimodal function optimization problems, Appl. Soft Comput. 33 (2015) 170–182, https://doi.org/10.1016/j.asoc.2015.04.002.
[31] van der Pol, B. LXXXVIII. On "Relaxation-Oscillations." Lond. Edinb. Dublin Philos. Mag. J. Sci. 1926, 2, 978–992, doi:10.1080/14786442608564127.
[32] J.F. Duffy, S.W. Cain, A.M. Chang, A.J.K. Phillips, M.Y. Münch, C. Gronfier, J. K. Wyatt, D.J. Dijk, K.P. Wright, C.A. Czeisler, Sex difference in the near-24-hour intrinsic period of the human circadian timing system, Proc. Natl. Acad. Sci. 108 (2011) 15602–15608, https://doi.org/10.1073/pnas.1010666108.
[33] A.J.K. Phillips, P. Vidafar, A.C. Burns, E.M. McGlashan, C. Anderson, S.M. W. Rajaratnam, S.W. Lockley, S.W. Cain, High sensitivity and interindividual variability in the response of the human circadian system to evening light, Proc. Natl. Acad. Sci. 116 (2019) 12019–12024, https://doi.org/10.1073/pnas.1901824116.
[34] Aschoff, J. Exogenous and endogenous components in circadian rhythms. Cold Spring Harbor Symposia on Quantitative Biology 1960, 25, 11–28, doi:10.1101/SQB.1960.025.01.004.
[35] A.C. Skeldon, A.J.K. Phillips, D.J. Dijk, The effects of self-selected light-dark cycles and social constraints on human sleep and circadian timing: a modeling approach, Sci. Rep. 7 (2017) 45158, https://doi.org/10.1038/srep45158.
[36] H.P. Piepho, An algorithm for a letter-based representation of all-pairwise comparisons, J. Comput. Graph. Stat. 13 (2004) 456–466, https://doi.org/10.1198/1061860043515.
[37] P.N. Suganthan, Particle swarm optimiser with neighbourhood operator, Proceedings of the, in: Proceedings of the Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), Washington, DC, USA, IEEE, 1999, pp. 1958–1962.
[38] J. Matsuzaki, Y. Kawahara, T. Izawa, Punctual transcriptional regulation by the rice circadian clock under fluctuating field conditions, Plant Cell 27 (2015) 633–648, https://doi.org/10.1105/tpc.114.135582.
[39] K.M. Sharkey, M.A. Carskadon, M.G. Figueiro, Y. Zhu, M.S. Rea, Effects of an advanced sleep schedule and morning short wavelength light exposure on circadian phase in young adults with late sleep schedules, Sleep Med. 12 (2011) 685–692, https://doi.org/10.1016/j.sleep.2011.01.016.
[40] J.E. Stone, S. Postnova, T.L. Sletten, S.M.W. Rajaratnam, A.J.K. Phillips, Computational approaches for individual circadian phase prediction in field settings, Curr. Opin. Syst. Biol. 22 (2020) 39–51, https://doi.org/10.1016/j.coisb.2020.07.011.
[41] J. Vicente-Martínez, P.F. Almaida-Pagan, A. Martinez-Nicolas, J.A. Madrid, M. A. Rol, M.Á. Bonmatí-Carrión, Integral analysis of circadian rhythms. Statistical Methods at the Forefront of Biomedical Advances, Springer International Publishing, Cham, 2023, pp. 185–236. Larriba, Y.ISBN 978-3-031-32728-5.