Soft Comput
DOI 10.1007/s00500-013-1043-5

METHODOLOGIES AND APPLICATION

Population-based optimization of cytostatic/cytotoxic combination cancer chemotherapy

Gabriela Ochoa · Minaya Villasana

Abstract This article studies the suitability of modern population based algorithms for designing combination cancer chemotherapies. The problem of designing chemotherapy schedules is expressed as an optimization problem (an optimal control problem) where the objective is to minimize the tumor size without compromising the patient's health. Given the complexity of the underlying mathematical model describing the tumor's progression (considering two types of drugs, the cell cycle and the immune system response), analytical and classical optimization methods are not suitable, instead, stochastic heuristic optimization methods are the right tool to solve the optimal control problem. Considering several solution quality and performance metrics, we compared three powerful heuristic algorithms for real-parameter optimization, namely, CMA evolution strategy, differential evolution, and particle swarm pattern search method. The three algorithms were able to successfully solve the posed problem. However, differential evolution outperformed its counterparts both in quality of the obtained solutions and efficiency of search.

Keywords Evolutionary algorithms · Differential evolution · Evolution strategies · Particle swarm optimization · Numerical optimization · Optimal control · Cancer chemotherapy · Combination chemotherapy

Communicated by G. Acampora.

G. Ochoa (✉)
Computing Science and Mathematics, School of Natural Sciences, University of Stirling, Stirling, Scotland
e-mail: gabriela.ochoa@cs.stir.ac.uk

M. Villasana
Departamento de Cómputo Científico y Estadística, Universidad Simón Bolívar, Caracas, Venezuela
e-mail: mvillasa@usb.ve

1 Introduction

Chemotherapy is a type of cancer treatment where drugs are used to destroy tumor cells. Some cancer drugs are given on their own but often several drugs are given together, which is known as combination chemotherapy. Chemotherapy produces damaging side-effects, in consequence, it is normally given in rounds of treatment that alternate with resting (recovering) periods. Several of these cycles are normally required to reduce the tumor to a minimum level after which the organism can naturally eliminate the remaining tumor cells. The number and scheduling of the treatment rounds depends on the type and stage of the disease; and the patient's health, among other factors. An oncologist should plan treatments on a personal basis. Given the complexity of this task, the use of computer-based decision support systems is worth investigating.

The process of designing chemotherapy schedules is here expressed as an optimization problem (an optimal control problem) where the objective is to minimize the tumor size without compromising the patient's health. The goal in optimal control is to find a control scheme for an underlying dynamical system under a period of time to minimize a performance metric. In chemotherapy scheduling, the underlying dynamical system models the tumor progression; and the controls are the drug concentrations delivered into the system through time. Optimal control formulations of cancer chemotherapy can be found in the mathematical modeling literature (Panetta and Adam 1995; de Pillis 2003; de Pillis and Radunskaya 2001). When the

Published online: 07 April 2013

underlying mathematical formulation is sufficiently simple, the optimal control problem can be solved analytically or using classical numerical optimization methods. However, for increasingly complex and realistic cancer models, analytical or traditional numerical methods are no longer applicable, and some authors have turned to soft computing techniques, such as evolutionary and memetic algorithms to optimize chemotherapy schedules. Modern hybrid evolutionary and memetic algorithms (Acampora et al. 2011; Arnaldo et al. 2013, Ong et al. 2009, Payne et al. 2013) have increased their power and applicability in many complex domains such as knowledge acquisition (Acampora et al. 2011, 2012), data clustering (Li et al. 2013), logic circuit design (Houshmand et al. 2011), software engineering (Nunez et al. 2013), biological and chemical applications (Haupt et al. 2011).

In the realm of chemotherapy scheduling, Petrovski and McCall (1993) and Petrovski et al. (2004, 2006) have successfully used evolutionary algorithms and other modern heuristics. Villasana and Ochoa (2004), compared the performance of three meta-heuristics (genetic algorithms, evolution strategies, and simulated annealing) in a similar problem; and considered the effect of different terms in the objective function of the optimal control formulation (Ochoa et al. 2007). The main difference between the approaches of these two group of authors, lies in the underlying mathematical model of tumor growth. Petrovski et al. (2004, 2006) considered the Gompertz growth model with linear cell-loss effect (Wheldon 1998), without including interactions with the immune system. Villasana and Radunskaya (2003) employed a cancer model including the interactions between tumor cells and immune cells; and differentiating between cell phases for subsequent treatment with a cycle-phase-specific drug. In (2008), McCall et al. present a survey of approaches employing heuristic search methods to solve the cancer chemotherapy scheduling problem from an optimal control formulation. Examples of these approaches include the use of simulated annealing, a parallelized genetic algorithm, and multimodal optimization genetic algorithms (see McCall et al. 2008 and the references therein for further details). More recently, Liang and colleagues have applied several algorithms: an elitist genetic algorithm in Tse et al. (2007) and in Liang et al. (2006) they employ a memetic approach using the elitist genetic algorithm with the forward iterative dynamic programming as the local search. Some of these studies consider the application of multiple drugs; however, none of them considers a drug that is not cytotoxic nor do most models incorporate the tumor interaction with the body's natural defense system. These two features uniquely distinguish the cancer model considered in this article.

The incorporation of a cytostatic drug is only possible in a model that considers the cell cycle and the use of delay differential equations. This is the case of the model presented in Villasana and Radunskaya (2003). Although the original model considered only the effects of a cytotoxic cycle-phase-specific drug, it is extended in Villasana et al. (2010) by incorporating the action of a cytostatic agent in conjunction with the original cytotoxic agent. The idea behind combining these two agents is that the cytostatic drug can halt the rapid progression of the cancerous cells through their cell cycle at a certain phase so that upon release they are mostly agglutinated in the most vulnerable stage to the action of cytotoxic drugs. An example of a cycle-phase-specific cytotoxic drug is Taxol (paclitaxel), and an example of a cytostatic drug is Iressa (gefitinib). These two drugs were identified and included in the model in Villasana et al. (2010), where alternative objective function formulations and problem representations were also considered; and the stated optimal control-problem was solved using a single algorithm, the Covariance Matrix Adaptation (CMA) evolution strategy (Hansen and Ostermeier 1996, 2001).

The main contribution of the present study is to perform a comprehensive computational analysis comparing main stream population-based algorithms with their default parameter settings, when solving an optimal control formulation of cancer chemotherapy based on a realistic cancer model and multiple-type drugs. In particular three modern brands of actively studied algorithms are considered: the CMA evolution strategies (Hansen and Ostermeier 1996, 2001), differential evolution (Storn and Price 1995, 1997), and a hybrid version of a particle swarm optimization algorithm (Vaz and Vicente 2007). The comparison considers the algorithm's efficiency; and the quality and diversity of the obtained solutions.

The next section summarizes the problem formulation including the underlying mathematical model of cancer growth, the optimal control problem and the solution representation. Section 3 overviews the three algorithms tested. Section 4 details the experimental setup and performance metrics used. Section 5 outlines the results, and Sect. 6 summarizes and discusses the main findings.

2 Problem formulation

2.1 Biomedical background

Chemotherapy is a cancer treatment using powerful chemical agents to destroy rapidly dividing cells. Some of these drugs also interfere with the process of cell division,

reducing the tumor cells to a minimum level allowing the natural elimination of the remaining cancerous cells. There are several types of cancer drugs. The mathematical model used in this study incorporates two types of them: a cytotoxic and a cytostatic drug, which are both cycle-phase specific. Cycle-phase-specific drugs are those acting on a specific phase of the cell cycle. The cell cycle, is the process between two cell divisions or mitosis. It encompasses four stages: G_1, S, G_2, and M, where G_1 and G_2 are resting phases (or Gap periods), S is the synthetic period, and M or mitosis is the time during which cells segregate the duplicated DNA material between daughter cells.

A cytotoxic drug is a toxic agent that destroys cells. Taxol (paclitaxel) is one of such drugs, which acts by inhibiting mitosis and inducing programmed cell death. A cytostatic drug is not toxic to the cells, instead, it acts by halting their rapid cycling. The cytostatic drug modeled in this study is called Iressa (gefitinib). Combining paclitaxel and gefitinib has been found to be an effective treatment and produced better results than using the cytotoxic drug only (Park et al. 2004; Ciardiello et al. 2000). The idea of this type of combined therapy is that once a number of cells are captured at a certain phase by the cytostatic agent, they will be more sensitive to the effect of the toxic agent.

2.2 Mathematical model

The patient's model used is a competition model of tumor progression based on delay differential equations with the following main features:

1. The tumor cell population is divided into two compartments: tumor during interphase (T_I), where interphase is the period comprising G_1 through G_2; and tumor during mitosis (T_M). This division of the tumor throughout the cell cycle allows the modeling of cycle-phase specific drugs.
2. The immune system is modeled based on the cytotoxic T cells (CTL). The variable I represents the population of immune system cells (see Villasana and Radunskaya 2003 for a full discussion).
3. Two drugs of different types and effect are modeled: a cytotoxic drug and a cytostatic drug (Villasana et al. 2010). These drugs have different effects. The cytotoxic drug (u) aims at killing the tumor cells while affecting other normal and immune cells; while the cytostatic drug (v) arrests tumor cells during interphase so that after agglutination they may be targeted with a cytotoxic action.

Let τ be the resident time of cells in interphase. Figure 1 illustrates the patient model and the governing equations are:

$$
\begin{aligned}
T_I' &= 2a_4T_M - (c_1I + d_2)T_I - \max\left(0, -\frac{1}{330}v + 1\right)a_1T_I(t-\tau)\\
T_M' &= \max\left(0, -\frac{1}{330}v + 1\right)a_1T_I(t-\tau) - d_3T_M - a_4T_M\\
&\quad - c_3T_MI - k_1(1 - e^{-k_2u})T_M\\
I' &= k + \frac{\rho I(T_I + T_M)^3}{\alpha + (T_I + T_M)^3} - c_2IT_I - c_4T_MI - d_1I\\
&\quad - k_3(1 - e^{-k_4u})I\\
u_1' &= -\lambda_1u_1 + \mathbf{c}_u(t)\\
u_2' &= -\lambda_2u_2 + \mathbf{c}_u(t)\\
v' &= -\ln(\sqrt{2})v + \mathbf{c}_v(t)
\end{aligned}
\tag{1}
$$

where $'$ denotes derivatives with respect to time and with initial data given by:

$$
\begin{aligned}
&T_I(t) = \phi_1(t) && \text{for } t \in [-\tau, 0]\\
&T_M(t) = \phi_2(t) && \text{for } t \in [-\tau, 0]\\
&I(t) = \phi_3(t) && \text{for } t \in [-\tau, 0]\\
&u_1(0) = u_2(0) = v(0) = 0
\end{aligned}
$$

The fraction of cells that are targeted by the cytotoxic drug is modeled by a term of the form $-k_j(1 - e^{-h_iu})$, where u is the drug concentration and h_i and k_j are parameters that model the effectiveness. Linear competition terms are considered for tumor immune interactions ($I\,T_I$ and $I\,T_M$). The decay rate of paclitaxel is modeled as before (Villasana and Radunskaya 2003; Villasana and Ochoa 2004; Villasana et al. 2010) with two separate elimination terms (u_1 and u_2 such that the total drug concentration is their linear convex combination). Equations 4 and 5 of system (1) model this situation with multiple drug applications in time. The drug applications are identified with the function $\mathbf{c}_u(t)$, which is the concentration of paclitaxel that goes in the system at time t. The action of high concentrations of a cytostatic drug is to arrest cells in the interphase compartment. The equation that governs the dynamics for the concentration of the drug v is

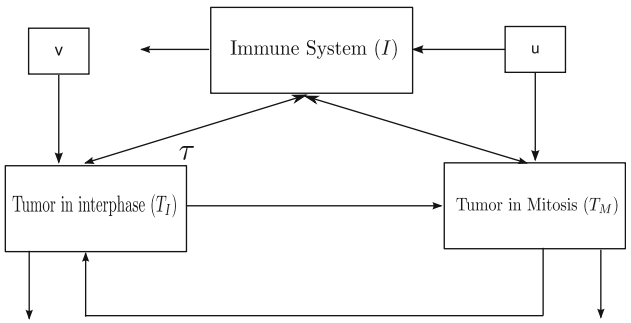

Fig. 1 Diagram of the patient model including the cytotoxic and cytostatic drugs

$$\frac{dv}{dt} = -\ln(\sqrt{2})v + c_v(t), \tag{2}$$

where c_v is the amount of drug administered at time t. This dynamics is identified after reports on Iressa's cytostasis and information on it's elimination half life (Piechocki et al. 2006). c_v is regarded in Eq. (2) as a continuous input and has been incorporated into the system dynamics. For more details on this dynamics the reader is referred to Villasana et al. (2010).

The first and second equation of system (1), have a modified cycling term. This term is inspired from the model in Swierniak et al. (1996) and reflects that the proliferating fraction of cells depends on the concentration level of the cytostatic drug. Parameter estimation was performed on the drug free system (Villasana and Radunskaya 2003). The information available for paclitaxel in Hardman and Linbird (1996), Zoli et al. (1995), Chuang et al. (1993) was used for estimating the paclitaxel drug terms. The system was re-scaled using the same non-dimensionalization as in Villasana and Radunskaya (2003). The maximum drug concentration after 3 days of treatment (330 mg) was used as the scaling factor for the drug concentration of v. The remaining model parameter values used are the same as in Villasana and Ochoa (2004).

The associated drug-free system for (1) can have up to five different fixed points depending on the parameter values (Villasana and Radunskaya 2003). One of this fixed points is always present, namely $(0, 0, k/d_1)$. This point represents the desirable scenario of a tumor-free environment with a positive immune population.

2.3 Tumor initial conditions

In order to simulate the progression of the disease and the course of the chemotherapy, the simulated treatment starts from a constant initial state: $(T_I(0), T_M(0), I(0)) = (1.3, 1.2, 0.9)$, where the values represent cell counts and are normalized by a factor of 10^6. These initial conditions simulate a patient with a tumor that is not controllable by the immune system alone. The objective of the computer-based approach is to devise a treatment conducting the system to the tumor-free basin of attraction, subject to the constraint of not reducing the immune system population below its initial critical level ($I_{thr} = 0.9$). The system's fixed points were numerically calculated and the desired state of the tumor-free attractor is reached with the following tumor cells values: $(T_I^*, T_M^*) = (0.3, 0.3)$. For these values (or below), the patient can be considered as cured.

2.4 Mathematical formulation and objective function

The problem of designing a chemotherapy protocol can be viewed as an optimal control problem where the goal is to eradicate the tumor while maintaining acceptable levels for the immune system that represents the patient's health. Mathematically this means using the drugs introduced into the system as control functions ($c_u(t)$ and $c_v(t)$) to drive the tumor into the tumor-free fixed point's basin of attraction, while keeping the immune system above a certain threshold.

The general mathematical formulation is:

$$\begin{array}{ll} \textbf{Min} & J(T_I(t), T_M(t), I(t), u(t), v(t)) \\ \textbf{s.t} & \text{Equations for patient dynamics} \end{array} \tag{3}$$

The objective function in this optimization process is a key component. In Villasana et al. (2010) a set of three objective functions (from a pool of twelve originally posed and preliminary analyzed) were studied and compared in terms of their effectiveness and solution quality. The objective function described below, rendered shorter and effective treatments while keeping low doses of cytotoxic drug, thus providing better patient health.

$$J(T_I(t), T_M(t), I(t), u(t), v(t)) = |T_I(t_f) - 0.3| + |T_M(t_f) - 0.3| + \frac{1}{T_f}\int_0^{T_f}(T_M + T_I)dt + F(I) + P$$

The terms in this function are:

$T_I(t_f)$: the tumor population in interphase at the end of treatment. The term $|T_I(t_f) - 0.3|$ minimizes the Euclidean distance form a specifically computed point inside the basin of attraction.

$T_M(t_f)$: the tumor population in mitosis at the end of treatment.

$\frac{1}{T_f}\int_0^{T_f}(T_M + T_I)dt$: integral term aggregating the average size of the tumor during the whole treatment. This term is included to avoid large oscillations in tumor size during treatment which can be detrimental.

$F(I)$: the constraint on the immune cell's population, which is calculated using Eq. 4.

P: the treatment length expressed as the number of cycles. Notice that a shorter treatment in terms of number of cycles do not necessarily represent a shorter treatment time in days. However, a smaller number of cycles is generally preferable.

$$F(I) = \begin{cases} 0 & \text{if } I(t) > I_{thr} \\ I_{thr} - I(t) & \text{if } I(t) \le I_{thr} \end{cases} \tag{4}$$

where $I_{thr} = 0.9$.

Pontryagin's Maximum Principle was used to obtain the necessary conditions for an analytical solution to this problem (Villasana and Ochoa 2004), yielding this as a singular problem. Since the amount of drug (the control variables) in this formulation are bounded below and above, the candidate solutions are *bang* − *bang*, which means that the optimal control switches from one extreme to the other at certain times (i.e. it is never strictly in between the bounds).

2.5 Problem encoding

The control variables are the drug concentrations delivered over time. These values determine the treatment dosing and scheduling. Given that we are dealing with bang-bang control, what needs to be determined are the optimal switching times (from *on* to *off*) between the drugs' application and the recovery periods. For a single drug, this can be encoded using real numbers representing switching times from application to resting (Villasana and Ochoa 2004). This representation can be easily extended to incorporate an additional drug. Four types of variables (two for each drug) are distinguished for encoding the drug application time lengths, and resting time lengths. Two additional variables are incorporated to account for the number of cycles of each drug, *PU* and *PV*. These are integer numbers in the range from 6 to 12 cycles, which proved sufficient to eradicate the simulated tumor in our experiments. This representation deals, then, with a variable length encoding. Figure 2 illustrates this encoding, which we termed *overlapping-permitting*, as it allows overlapping (simultaneous) applications of the cytostatic and cytotoxic drugs.

It is of interest, however, to find solutions that will have the same qualitative features as those found in Swierniak et al. (1996). For this purpose, a second encoding, that we termed *non-overlapping* representation was also considered, where the solutions always deliver non-overlapping treatment cycles, that is, the two drugs are never simultaneously applied. In this encoding, a treatment cycle consists of the application of the cytostatic drug, followed by the application of the cytotoxic drug, and a resting period. An additional variable, *P*, accounts for the number of cycles (allowed to vary from 6 to 12 cycles). Figure 3 illustrates the non-overlapping representation. Notice that this encoding induces a smaller search space, since a single resting time for each cycle is encoded. In Villasana et al. (2010), these two encodings were compared and the non-overlapping representation was found more effective and used smaller amounts of toxic agent. It was, therefore, selected for this study.

The range for the resting-time periods is [0, 30] (measured in days), where the lower bound indicates a continued treatment with no recovery period and the upper bound reflects the current chemotherapy standards. The maximum tolerated doses for paclitaxel and Iressa were considered to obtain bounds on the control variables whose ranges are [0.2, 5] and [1, 5] days, respectively.

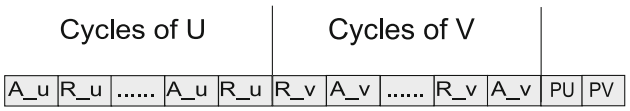

Fig. 2 Schematic view of a candidate solution (control variable). Solution encoding for *overlapping-permitting* treatments. The vector has two sections, the first corresponds to the cytotoxic drug (*u*) while the second to the cytostatic drug (*v*). The cytotoxic drug has cycles beginning with an application (*A*), while the cytostatic drug's cycle begins with a resting period (*R*). Application and resting times are real numbers representing days. Two additional variables are incorporated to account for the number of cycles of each drug, *PU* and *PV*

3 The evolutionary optimization algorithms

We consider two powerful evolutionary algorithms for numerical optimization, namely, Covariance Matrix Adaptation (CMA) evolution strategy (Hansen and Ostermeier 1996, 2001) and differential evolution (Storn and Price 1995, 1997). The third algorithm is a hybrid between a rigorous global optimization method (pattern search) and a heuristic population-based approach (particle swarm optimization). For the three algorithms, we used publicly available Matlab toolboxes provided by their respective authors. A description of these algorithms and their default parameter settings is outlined below.

3.1 CMA evolution strategy

Evolution strategies (ES) are a branch of evolutionary algorithms developed in Germany in the 1960s and first applied to hydrodynamical design problems (Rechenberg 1973). ES use mutation as the main operator, and propose a form of self-adaptive mutation, in which the mutation parameters are also part of the chromosome and, thus,

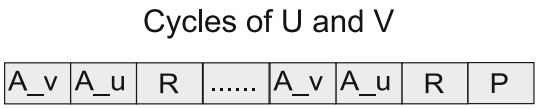

Fig. 3 The solution representation for *non-overlapping* treatments. The vector encodes cycles alternating applications (*A*) of the two drugs (*u* cytotoxic agent, *v* cytostatic agent,), finalizing with a resting period (*R*). The times are real numbers measuring days. The last variable, *P*, encodes the length of the treatment (i.e. the number of chemotherapy cycles)

subject to evolution). CMA-ES are a modern version of ES suited for real-parameter optimization (Hansen and Kern 2004; Hansen and Ostermeier 2001). They differ from the canonical ES mainly in the shape of mutation distribution, which is generated according to a covariance matrix. This matrix is modified during the evolutionary process and thus can adapt to the local shape of the search space, which increase the efficiency and convergence of the algorithm. The CMA-ES toolbox used in this study (Hansen 2006) suggests values for its main parameters: a the population size of $\lambda = 4 + \lfloor 3 \ln N \rfloor$ (where N is the problem size); and a initial mutation step size equal to one a third of the parameters range. It also incorporates weighted recombination, were the weights $(w_i, \ldots, w_\mu)$ are set to: $w_i = \ln \frac{\lambda+1}{2} - \ln i$, for $i = 1, \ldots, \mu$.

3.2 Differential evolution

The differential evolution (DE) algorithm (Price et al. 2005; Storn and Price 1995, 1997) is an evolutionary method that uses a more deterministic (greedy) approach to problem solving than other more traditional evolutionary and stochastic local search algorithms. DE incorporates a powerful and conceptually simple self-adapting mechanisms for the mutation strength based on a small population. The algorithm's key idea is a method for generating candidate solutions (parameter vectors), which consist of adding the weighted difference vector between two population members to a third member. DE has been successful in solving real-world engineering problems. Several variants have been developed, which are classified using the notation: $DE/\alpha/\beta/\gamma$. These three parameters describe the algorithm main design choices:

- α: method for selecting the parent solution to constitute the base vector.
- β: number of difference vectors used to mutate the base vector.
- γ: recombination mechanism used to create offsprings. The *bin* acronym indicates that a number of independent binomial trials are used within the recombination mechanism.

For this study, the most widely-known variants: namely $DE/rand/1/bin$, and $DE/local-to-best/1/bin$, were selected. Our preliminary experiments indicated that the second variant produced better results, therefore, we used it in our comparisons. The main DE parameters are the weighting factor F, and the crossover rate CR. We used the default parameters as suggested in the DE Matlab toolbox (Hansen 2008; Price et al. 2005): $F = 0.85$ and $CR = 0.8$.

3.3 Particle swarm pattern optimization

The particle swarm pattern optimization algorithm (PSwarm) (Vaz and Vicente 2007) combines the local optimality convergence property of pattern search with the more aggressive explorative property of particle swarm optimization. This hybrid algorithm proved to be very effective and more robust when compared to other powerful global optimizers (see Vaz and Vicente 2007 for details). Pattern search is a direct optimization method in which trial points within a mesh are computed iteratively following strict calculations until necessary conditions for local optimality are achieved (Kolda et al. 2003). Particle Swarm Optimization is a swarm intelligence method in which particles (representing solutions to the problem) fly through a multi-dimensional continuous space (Kennedy and Eberhart 1995). Each particle *id* has a current position x^{id} and a current velocity v^{id}. In addition, each particle knows its best position so far and the best overall position for the whole swarm. With this information, at each time step, the new particle's velocity and position are iteratively computed.

The hybrid PSwarm method is a pattern search method that incorporates particle swarm optimization to compute trial points belonging to the pattern search mesh. When this search fails to find an improved solution in an iteration, local optimization is carried out on the best position over all particles in the swarm. The PSwarm freeware Matlab toolbox used for the experiments is available in (Vaz and Vicente 2010) and the default parameters used are: $\mu = 0.5$ and $\nu = 0.5$. The inertia is computed as a linear decay function of the actual iteration with respect to the maximum iterations allowed. The default parameters for this decrement are the initial and final inertia set to 0.9 and 0.4 respectively and the maximum allowed iterations here set to 100.

4 Experimental setup

A candidate solution of the formulated optimal control problem consists of 37 real numbers that correspond to 12 applications of the cytotoxic drug, 12 applications of the cytostatic drug, and 12 resting times. The last value encodes the effective number of cycles used, P, out of the maximum possible 12 (see Fig. 3).

The algorithms' terminate after 100 iterations. This was chosen after observing a flattening of the performance curves at around 60 iterations in a set of preliminary runs with 300 iterations. In order to ensure that the computational effort of each algorithm was comparable, the population sizes of the three algorithms were set to the same

value. We followed the suggestion given by CMA-ES Toolbox. The offspring population was set to $\lambda = 4 + \lfloor 3 \ln N \rfloor$ (where N is the problem dimension). As discussed above, a candidate solution is represented with $N = 37$ real numbers, therefore, $\lambda = 14$. With this population size and considering 100 iterations, both the CMA-ES and the DE algorithms accumulated exactly 1,400 function evaluations per run. However, due to its characteristics, the hybrid PSwarm algorithm produced a variable number of evaluations in each run, despite having a population size of 14. Therefore, to ensure a comparable computational effort for this algorithm, an alternative stopping condition was set for it. Namely, the algorithm was allowed to run until a maximum of 1400 function evaluations. In this case, the number of PSwarm iterations varied with the following distribution: maximum = 100, minimum = 47, mean = 73.5 and standard deviation = 17.39.

We conducted 30 independent replicas for each algorithm and considered a number of metrics to asses the algorithms' performance and quality of the obtained solutions. The algorithms performance was also assessed using nonparametric statistical tests. In order to reduce the effects of random sampling, the initial population for each of the 30 replicas of each algorithm, was the same. In other words, the 3 algorithms were started from the same 30 initial populations.

4.1 Algorithm performance metrics and statistical analysis

In order to asses the performance and suitability of the algorithms, we considered the following metrics and statistical procedures:

4.1.1 Algorithm performance

The most direct metric for assessing the performance of the competing algorithms is the best objective function value obtained at the end of the run. Since 30 runs are conducted for each algorithm, these values are presented as box-plots and non-parametric tests are used to show significant statistical differences among the competing algorithms.

4.1.2 Nonparametric statistical tests

Statistical methods can be categorized into parametric and nonparametric. Parametric tests have been used when analyzing soft computing techniques. However, they are based on assumptions (i.e. independence, normality, and homoscedasticity) that are likely violated when considering stochastic search algorithms (Derrac et al. 2011). Nonparametric statistical procedures overcome this limitation and can be used for comparing this type of algorithms. In this article we used CONTROLTEST, a tool for nonparametric comparison between algorithms (Derrac et al. 2011).[1] Specifically, two of the proposed non-parametric tests were conducted: Friedman (1940) and Aligned Friedman (Hodges and Lehman 1962) tests. The Friedman test can be used to asses whether in a set of k samples ($k > 2$), at least two samples represent populations with different median values. Therefore, it is a multiple comparison test that aims to detect significant differences between the behavior of two or more algorithms. It ranks the algorithms for each problem (or run) separately; the best performing algorithms should have rank 1, the second best rank 2, and so forth. A drawback of the ranking scheme of the Friedman test is that it allows for intra-set comparisons only. In the Aligned Friedman test, a value of location is computed as the average performance achieved by all algorithms in each run (or problem). Then the difference between the performance obtained by an algorithm and the value of location is obtained. These differences (aligned observations) are then ranked. This procedure allows comparisons among runs (inter-set comparisons) which may be desirable when the number of algorithms for comparison is small. This is relevant in our study since 3 algorithms are compared and for each of the 30 runs, the algorithms start from the same initial conditions. In order to asses the statistical significance of the results, the *p-value* for each test is computed, which provides information about whether a statistical hypothesis test is significant or not. In our study the null hypothesis, H_0 represents no significant differences between the algorithms. The *p-values* also indicate how significant the results are: the smaller the *p-value* the stronger the evidence against H_0.

4.1.3 Algorithm convergence

In order to asses the algorithms' convergence rates, we visualized the curves of the objective function over the algorithm's iterations. We considered both the progression curves of the best run and the typical run (i.e. a run producing the median value of the objective function) for each algorithm.

4.1.4 Inter-run diversity

The diversity of the different solutions obtained by an algorithm across several runs, is an interesting property. To measure this, we calculated the *moment of Inertia* of the best 30 solutions obtained by each algorithm. This metric was used in Morrison and Jong (2002) in the context of

[1] The ControlTest package is available at http://sci2s.ugr.es/sicidm/.

evolutionary algorithms, but it is inspired by concepts from engineering where it measures how the mass of an object is distributed. Given a set of vectors p of dimension k, we first calculate the centroid of the set p: $c_i = \frac{\sum_{j=1}^{p} x_{i,j}}{p}$, for $i = 1, 2, \ldots, k$, where $x_{i,j}$ corresponds to the ith coordinate of the jth point. We can then calculate the moment of inertia of p by:

$$Inertia = \sum_{i=1}^{k} \sum_{j=1}^{p} (x_{i,j} - c_i)^2$$

A higher Inertia value, indicates a higher solution set diversity.

4.2 Solution quality metrics

Since the objective function aggregates a number of measurements, it is interesting to explore the quality of the obtained solutions according to the most relevant aspects. In order to do so, the following metrics were considered:

4.2.1 Area under the tumor curve (AUTC)

Accumulates the number tumor cells through the whole treatment process. Minimizing this metric clearly produces more effective treatments.

4.2.2 Total application time for the cytotoxic drug (AT)

Measures the total amount of the toxic drug applied to the system across the treatment process. Minimizing this metric will reduce the treatment's side effects and favor the patient health.

4.2.3 Duration of treatment (DT)

Calculates the total treatment duration in days. A shorter treatment is desirable over a longer one.

4.2.4 Immune system health (ISH)

Calculates the average deviation of the immune system population with respect to its critical threshold (I_{thr}). This measure represents the patient health and it is calculated by:

$$ISH = \int_0^{T_f} I(t)dt - I_{thr} * T_f.$$

To summarize, preferred treatments would have low *AUTC*, *AT*, and *DT* values, and high *ISH* values.

5 Results

Figure 4 illustrates for each algorithm the value and distribution of the best objective function values. DE produced both the best overall solution and the best typical performance, followed by CMA-ES and PSwarm in the third place.

In order to asses the statistical significance of the results, Table 1 reports the average ranks computed through the Friedman and Aligned Friedman tests. For both tests the smaller the average rank the better the algorithm. Since the study considers 3 algorithms and 30 runs per algorithm, the best possible Friedman rank for a given algorithm is 1 and the best Aligned Friedman rank is 30, but the figures in the table correspond to the average ranks. As can be seen in Table 1, the order between the algorithms is the same for the two tests and it is consistent with the box-plots in Fig. 4. The evidence supports that differential evolution is the best performing algorithm with a Friedman rank of 1.483 and Aligned Friedman rank of 30.433, followed by CMA-ES, with PSwarm in the last place. The *p-values* computed through the statistics of both tests (0.0005394 and 0.0000115) strongly suggest the existence of significant differences among the algorithms considered, i.e. the null hypothesis, H_0 suggesting no difference between algorithms is strongly rejected.

The dispersion or diversity, from the point of view of the solutions (drug schedules) themselves can be appreciated by the moment of inertia values (see Sect. 4.1), reported in Table 2. The inertia is larger for PSwarm and smaller for DE, with an intermediate figure for CMA-ES. This is an interesting observation if we consider that, for the three algorithms, the runs started from the same set of initial populations.

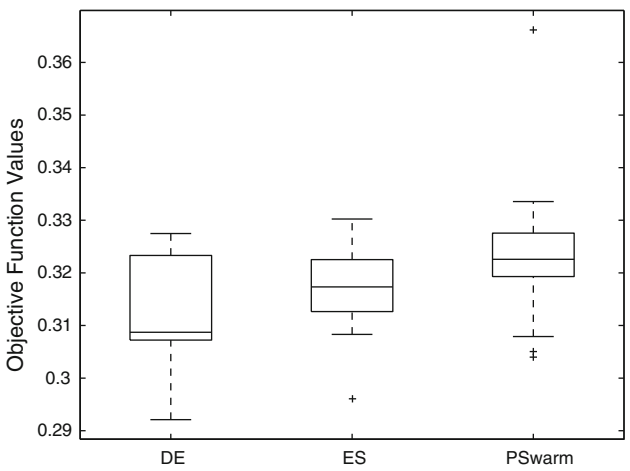

Fig. 4 Comparing algorithms' performance according to the objective function values of the best obtained solution at the end of each run. *DE* differential evolution, *ES* CMA-evolution strategy, *PSwarm* pattern particle swarm optimization

Table 1 Average Friedman and Aligned Friedman ranks

Algorithm	Friedman	A. Friedman
DE	1.483	30.433
ES	2.033	43.933
PSwarm	2.483	62.133

DE differential evolution, *ES* CMA-evolution strategy, *PSwarm* pattern particle swarm optimization

Table 2 Inter-run diversity (among the 30 runs) measured with the moment of inertia

	DE	ES	PSwarm
Inertia	1.7188	1.8392	2.3435

Table 3 Number of best solutions (out of the 30 runs) with a given number of cycles obtained by each algorithm

Algorithm	10 Cycles	11 Cycles	12 Cycles
DE	3	14	13
ES	7	16	7
PSwarm	0	6	24

Our encoding allows the design of treatments with different number of cycles. This number is subject to evolution and was allowed to vary between 6 and 12. As seen in Table 3, DE and ES were able to find a larger proportion of treatments with a shorter number of cycles, namely 10 and 11. PSwarm was not able to produce treatments of 10 cycles and produced only 6 treatments of 11 cycles.

Figure 5 illustrates the value and distribution of the solution quality metrics described in Sect. 4.2: (a) area under the tumor curve, (b) cytotoxic drug application time, (c) duration of treatment, and (d) immune system health. These metrics give more detailed information about the solutions and their various components aggregated in the objective function. Good solutions should have low values for metrics (a–c), keeping (d) as high as possible. We can see that DE obtained both the best overall and typical solutions with respect to the most relevant quality measurement (area under the tumor curve) which represents the tumor size across the whole treatment (plot (a) in Fig. 5). This is consistent with the results reported in Fig. 4 and Table 1. Moreover, DE was able produce the smaller overall tumor sizes with shorter drug application times [plot (b) in Fig. 5] relative to the second to best algorithm. DE also achieved higher immune system health [plot (c) in Fig. 5] with respect to CMA-ES, which seconded DE in all other measures.

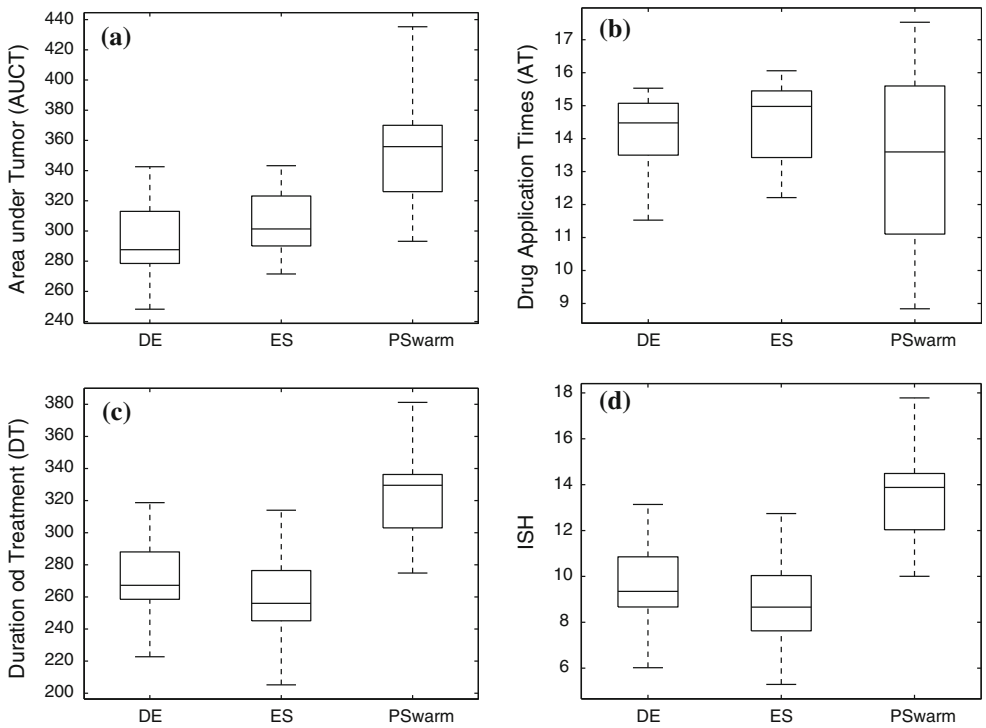

Fig. 5 Boxplots comparing the three competing algorithms, with respect to the performance measures: **a** area under the tumor, **b** drug application time, **c** total duration of the treatment and **d** immune system health. For the metrics **a–c** the smaller the better, whereas for metric **d** the larger the better. *DE* differential evolution, *CMA-ES* CMA-evolution strategy, *PSwarm* particle swarm pattern optimization

Figure 5 suggest that metrics (c) duration of treatment, and (d) immune system health are correlated, as the values for each algorithm have similar ranges. This is in accordance with the intuition that longer treatments can be more manageable for the immune system.

PSwarm produced the most benevolent treatments with higher immune system health values [plot (d)], longer durations [plot (c)], and shorter toxic drug application times [plot (b)]. However, this is achieved at the expense of a decreased treatment efficiency in terms of the overall tumor minimization [plot (a)], as compared to the other algorithms. CMA-ES is regarded as a competing algorithm but holds a middle ground in all of the performance measures between the two other algorithms considered.

Figure 6 illustrates the best protocols obtained by DE and PSwarm, which provide the two extreme solution scenarios found. We observe that DE produced a regular scheduling of the two drugs, while greater variation is observed for PSwarm with much smaller doses of toxic drug at the initial stages of the treatment.

Figure 7 illustrates the convergence behavior of the best and typical run of each algorithm on a semi-log scale. The best run for the ES corresponds to the outlier observed in Fig. 4, it shows the fastest initial convergence rate. However, DE is able to reach better objective values at later stages, and improves steadily during the search process. The typical runs show a faster initial convergence rate for ES and Pswarm, with DE maintaining an improved convergence across the whole search process and obtaining the best results. All the algorithms were run 30 times from the same 30 initial conditions. The curves in Fig. 4 are seen to start from different initial conditions, this is because the best and typical behavior of each algorithm correspond to different runs.

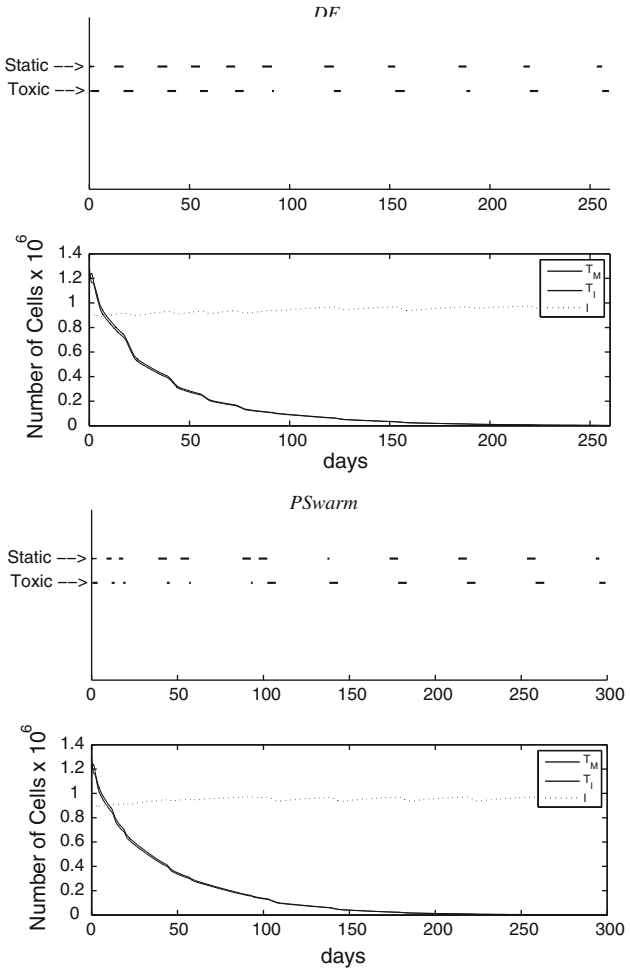

Fig. 6 The best combination therapy devised using *DE* differential evolution and *PSwarm* particle swarm pattern optimization

6 Conclusions

Two modern evolutionary algorithms for real-parameter optimization and a hybrid particle swarm pattern optimization algorithm are thoroughly compared for solving the problem of designing combined cancer chemotherapies, which is stated as an optimal control problem. The mathematical model of tumor progression displays several interesting features, distinguishing it from other models used in previous approaches to heuristic design of cancer chemotherapies. The model incorporates different stages of the

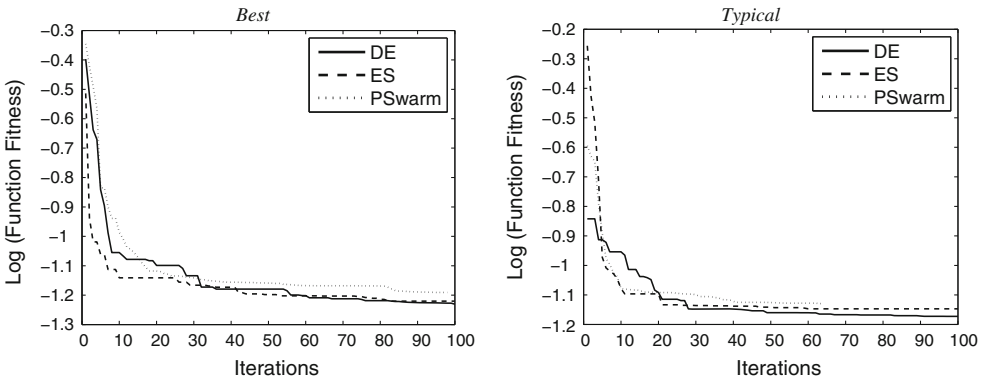

Fig. 7 Best-so-far objective value over time for the best (*top*) and typical (*bottom*) run. *DE* differential evolution, *CMA-ES* CMA-evolution strategy, *PSwarm* particle swarm pattern optimization

tumor cell's cycle and the immune system response. Modeling the cell-cycle allows the incorporation of so-called cyclic-specific drugs, and more importantly, of a different type of chemotherapy agent: a cytostatic drug. This agent acts by interfering with the rapid tumor cell cycling process, and arresting them at a certain phase at which they are most exposed to the effect of the toxic drugs.

The three algorithms compared were able to successfully solve the posed problem. It is worth noting that the algorithms were used without tuning their control parameters. With these default settings, Differential Evolution outperformed its counterparts both in quality of the obtained solutions and efficiency of search. The performance difference were statistically significant. The hybrid particle swarm pattern optimization algorithm was the least competitive of the algorithms with respect to the most important goal of reducing the tumor size. However, it produced more benevolent treatments (with respect to the immune system health) and a larger variability of alternative solutions, which can offer more options to the practitioner. Depending on the user's goal, one optimization algorithm may be preferred over another. For example, PSwarm produces larger treatments, and very similar resting-time lengths at the final stages the end of treatment. Upon inspection of the solutions, it was found that this algorithm produced the maximum possible length for the resting periods at the final stages of treatment. This explains the high immune system values observed in the protocols, as greater resting times allow for the recovery of the immune system. The immune system health is an important consideration, but the eradication of the tumor is likewise important and is not achieved as well by the PSwarm algorithm when compared to the solutions provided by the other two algorithms.

The proposed model may serve as a decision support system in the complex problem of designing personalized combination chemotherapies. Personalizing the model would require gathering each patient set of biological parameters and initial disease condition. This is a delicate task considering technical difficulties and ethical issues. The biological parameters and initial conditions used in this article served, however, as a case study to illustrate the effectiveness of evolutionary algorithms in optimal control formulations of cancer combination chemotherapies.

References

Acampora G, Cadenas J, Loia V, Ballester E (2011) Achieving memetic adaptability by means of agent-based machine learning. IEEE Trans Ind Inform 7(4):557–569

Acampora G, Cadenas J, Loia V, Ballester E (2011) A multi-agent memetic system for human-based knowledge selection. Syst Man Cybern Part A IEEE Trans Syst Hum 41(5):946–960

Acampora G, Loia V, Salerno S, Vitiello A (2012) A hybrid evolutionary approach for solving the ontology alignment problem. Int J Intell Syst 27(3):189–216

Arnaldo I, Contreras I, Millan-Ruiz D, Hidalgo J, Krasnogor N (2013) Matching island topologies to problem structure in parallel evolutionary algorithms. Soft Comput, pp 1–17. doi:10.1007/s00500-013-0995-9

Chuang L, Lotzova E, Cook K, Cristoforoni P, Morris M, Wharton T (1993) Effect of new investigational drug taxol on oncolytic activity and stimulation of human lymphocytes. Gynecol Oncol 49:291–298

Ciardiello F, Caputo R, Bianco R, Damiano V, Pomatico G, Placido SD, Bianco AR, Tortora G (2000) Antitumor effect and potentiation of cytotoxic drugs activity in human cancer cells by zd-1839 (Iressa), an epidermal growth factor receptor-selective tyrosine kinase inhibitor. Clin Cancer Res 6(5):2053–63

de Pillis LAR (2003) The dynamics of an optimally controlled tumnor model: a case study. Math Comput Model 37(11):1221–1244

de Pillis L, Radunskaya A (2001) A mathematical tumor model with immune resistance and drug therapy: an optimal control approach. J Theor Med 3:79–100

Derrac J, Garcia S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18

Friedman M (1940) A comparison of alternative tests of significance for the problem of m rankings. Ann Math Stat 11:86–92

Hansen N (2006) The CMA evolution strategy: a comparing review, towards a new evolutionary computation. In: Advances on estimation of distribution algorithms, vol 192, chap 4. Springer, Berlin, pp 75–102

Hansen N (2008) The CMA evolution strategy. http://www.lri.fr/hansen/cmaesintro.html

Hansen N, Kern S (2004) Evaluating the CMA evolution strategy on multimodal test functions. In: Yao X et al (eds) Parallel Problem solving from nature PPSN VIII. LNCS, vol 3242. Springer, Berlin, pp 282–291

Hansen N, Ostermeier A (1996) Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In: Proceedings of the 1996 IEEE international conference on evolutionary computation. IEEE Service Center, Piscataway, NJ, USA, pp 312–317

Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evol Comput 9(2):159–195

Hardman J, Linbird L (1996) Goodman and Gilman's the pharmacological basis of therapeutics, 9th edn. Mc Graw Hill, Health Profesion Division, New York

Haupt S, Haupt R, Young G (2011) A mixed integer genetic algorithm used in biological and chemical defense applications. Soft Comput 15(1):51–59

Hodges J, Lehman E (1962) Ranks methods for combination of independent experiments in analysis of variance. Ann Math Stat 33:482–497

Houshmand M, Saleh R, Houshmand M (2011) Logic minimization of QCA circuits using genetic algorithms. In: Gaspar-Cunha A, Takahashi R, Schaefer G, Costa L (eds) Soft computing in industrial applications. Advances in intelligent and soft computing, vol 96. Springer, Berlin, pp 393–403

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of the IEEE international conference on neural networks, vol 4. IEEE Press, Piscataway, NJ, USA, pp 1942–1948

Kolda TG, Lewis RM, Torczon V (2003) Optimization by direct search: new perspectives on some classical and modern methods. SIAM Rev 45(3):385–482

Li D, Hong G, Liyong Z (2013) A hybrid genetic algorithm-fuzzy c-means approach for incomplete data clustering based on

nearest-neighbor intervals. Soft Comput, pp 1–10. doi: 10.1007/s00500-013-0997-7
Liang Y, Leung KS, Mok TSK (2006) A novel evolutionary drug scheduling model in cancer chemotherapy. IEEE Trans Inf Technol Biomed 10(2):237–245
McCall J, Petrovski A, Shakya S (2008) Evolutionary algorithms for cancer chemotherapy optimization. In: Fogel GB, Corne DW, Pan Y (eds) Computational intelligence in bioinformatics. IEEE press series on computational intelligence, Chap. 12. Wiley Interscience, London, pp 265–296
Morrison R, Jong KD (2002) Measurement of population diversity. In: Artificial evolution: 5th international conference, evolution artificielle (EA 2001). LNCS, vol 2310. Springer, Berlin, pp 31–41
Nunez A, Merayo M, Hierons R, Nunez M (2013) Using genetic algorithms to generate test sequences for complex timed systems. Soft Comput 17(2):301–315
Ochoa G, Villasana M, Burke E (2007) An evolutionary approach to cancer chemotherapy. Genet Progr Evol Mach 8(4):301–318
Ong YS, Lim MH, Neri F, Ishibuchi H (2009) Special issue on emerging trends in soft computing: memetic algorithms. Soft Comput 13(8-9):739–740
Panetta J, Adam J (1995) A mathematical model of cycle-specific chemotherapy. Math Comput Model 22(2):67–82
Park JK, Lee SH, Kang JH, Nishio K, Saijo N, Kuh HJ (2004) Synergistic interaction between gefitinib (Iressa, ZD1839) and paclitaxel against human gastric carcinoma cells. Anticancer Drugs 15(8):809–19
Payne J, Giacobini M, Moore J (2013) Complex and dynamic population structures: synthesis, open questions, and future directions. Soft Comput, pp 1–12. doi:10.1007/s00500-013-0994-x
Petrovski A, McCall J (2001) Multi-objective optimisation of cancer chemotherapy using evolutionary algorithms. In: Zitzler E, Deb K, Thiele L, Coello CAC, Corne D (eds) Proceedings of the first international conference on evolutionary multi-criterion optimization, EMO 2001. LNCS, vol 1993. Springer, Berlin, pp 531–545
Petrovski A, Shakya S, McCall J (2006) Optimising cancer chemotherapy using an estimation of distribution algorithm and genetic algorithms. In: Cattolico M (ed) Genetic and evolutionary computation conference, GECCO 2006, vol 1. ACM Press, New York, pp 413–418
Petrovski A, Sudha B, McCall J (2004) Optimising cancer chemotherapy using particle swarm optimisation and genetic algorithms. In: Yao X, Burke EK, Lozano J, Smith J, Guervós JJM, Bullinaria JA, Rowe JE, Tiño P, Kabán A, Schwefel HP (eds) Parallel problem solving from nature—PPSN VIII. LNCS, vol 3242. Springer, Berlin, pp 633–641
Piechocki MP, Yoo GH, Dibbley SK, Amjad EH, Lonardo F (2006) Iressa induces cytostasis and augments fas-mediated apoptosis in acinic cell adenocarcinoma overexpressing HER2/neu. Int J Cancer 119:441–454
Price KV, Storn RM, Lampinen JA (2005) Differential evolution, a practical approach to global optimization. Natural computing series. Springer, Berlin
Rechenberg I (1973) Evolution strategy: optimization of technical systems by means of biological evolution. Fromman-Holzboog, Stuttgart
Storn R, Price K (1995) Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, International Computer Science Institute, Berkeley, CA, USA
Storn R, Price K (1997) Differential evolution a simple and efficient heuristic for global optimisation over continuous spaces. J Glob Optim 11(4):341–359
Swierniak A, Polanski A, Kimmel M (1996) Optimal control problem arising in cell-cycle- specific cancer chemotherapy. Cell Prolif 29:117–139
Tse SM, Liang Y, Leung KS, Lee KH, Mok TSK (2007) A memetic algorithm for multiple drugs cancer chemotherapy schedule optimization. IEEE Trans Syst Man Cybern Part B Cybern 37(1):84–91
Vaz AIF, Vicente LN (2007) A particle swarm pattern search method for bound constrained global optimization. J Glob Optim 39:197–219
Vaz AIF, Vicente LN (2010) Pswarm solver homepage. http://www.norg.uminho.pt/aivaz/pswarm/
Villasana M, Ochoa G (2004) Heuristic design of cancer chemotherapy. IEEE Trans Evol Comput 8(6):513–521
Villasana M, Ochoa G, Aguilar S (2010) Modeling and optimization of combined cytostatic and cytotoxic cancer chemotherapy. Artif Intell Med 50:163–173
Villasana M, Radunskaya A (2003) A delay differential equation model for tumor growth. J Math Biol 47:270–294
Wheldon T (1998) Mathematical models in cancer research. Adam Hilger, Bristol
Zoli W, Flamigni A, Frassineti G, Bajorko P, De Paola F, Milandri C, Amadori D, Gasperi-Campani A (1995) In vitro activitity of taxol and toxotere in comparison with doxorubicin and cisplatin on primary cell cultures of human breast cancers. Breast Cancer Res Treat 34:63–69