The Author(s) *BMC Bioinformatics* 2017, **18**(Suppl 4):116
DOI 10.1186/s12859-017-1523-1

BMC Bioinformatics

RESEARCH Open Access

Combination therapy design for maximizing sensitivity and minimizing toxicity

Kevin Matlock[1], Noah Berlow[2], Charles Keller[2] and Ranadip Pal[1*]

From Third International Workshop on Computational Network Biology: Modeling, Analysis, and Control (CNB-MAC 2016)
Seattle, WA, USA. 02-Oct-16

Abstract

Background: Design of personalized targeted therapies involve modeling of patient sensitivity to various drugs and drug combinations. Majority of studies evaluate the sensitivity of tumor cells to targeted drugs without modeling the effect of the drugs on normal cells. In this article, we consider the individual modeling of drug responses to tumor and normal cells and utilize them to design targeted combination therapies that maximize sensitivity over tumor cells and minimize toxicity over normal cells.

Results: The problem is formulated as maximizing sensitivity over tumor cell models while maintaining sensitivity below a threshold over normal cell models. We utilize the constrained structure of tumor proliferation models to design an accelerated lexicographic search algorithm for generating the optimal solution. For comparison purposes, we also designed two suboptimal search algorithms based on evolutionary algorithms and hill-climbing based techniques. Results over synthetic models and models generated from Genomics of Drug Sensitivity in Cancer database shows the ability of the proposed algorithms to arrive at optimal or close to optimal solutions in significantly lower number of steps as compared to exhaustive search. We also present the theoretical analysis of the expected number of comparisons required for the proposed Lexicographic search that compare favorably with the observed number of computations.

Conclusions: The proposed algorithms provide a framework for design of combination therapy that tackles tumor heterogeneity while satisfying toxicity constraints.

Keywords: Combination drug design, Lexicographic search, Toxicity constraints

Background

Design of drug therapies for cancer have primarily been considered from the perspective of sensitivity prediction using genetic characterizations as the predictor variables [1–3]. The genetic characterization based methodologies have severe limitations when the cancer type shows numerous aberrations among the samples and consequently predicting sensitivity based on similar steady state genetic characterizations provide limited accuracy. We have recently considered the modeling of tumor sensitivity using functional drug response data [4, 5], along with a functional and genetic characterization based integrated modeling [6]. Models were designed based on the in vitro tumor response to a set of drugs with known targets. However, the combination therapy design was based on the model reflecting the average behavior of the tumor tissue [7].

In this article, we incorporate heterogeneity by considering the effect of a drug on various parts of the tumors and incorporate toxicity by considering the effect of the drug on normal cell types. Consider a solid tumor tissue where the biopsy sample can be divided into separate samples to explore the heterogeneity. We can pass each biopsy sample through a drug screen to create a probabilistic target inhibition map (PTIM) [5] model. Let $MT_1, MT_2, \cdots, MT_k$ denote the k models corresponding to the k spatial tumor biopsies.

*Correspondence: ranadip.pal@ttu.edu
[1]Department of Electrical and Computer Engineering, Texas Tech University, 1012 Boston Ave, 79409 Lubbock, TX, USA
Full list of author information is available at the end of the article

For toxicity the affect of the drug is not limited to the same organ that the tumor resides in. To solve this we can pass normal cell cultures from different organs of the body through drug screens to create separate models of different organs to assess toxicity of the drugs. Note that, normal cells from kidney, lungs etc. of a specific cancer patient may not be readily available and thus, the response to drugs can be approximated by using drug screens on normal human based cell lines of kidney, lungs and other organs. The assumption is that variations in normal cell response to different drugs over different patients are smaller compared to tumor cell response over different patients. The normal cell response to different drugs can vary significantly for cells belonging to different organs in the body. Let $MN_1, MN_2, \cdots, MN_p$ denote p models corresponding to p different normal cell types.

The goal of the combination therapy design will be to select a set of drugs that will maximize the sensitivity over heterogeneous tumor models $MT_1, MT_2, \cdots, MT_k$ and minimize sensitivity over normal cell type models $MN_1, MN_2, \cdots, MN_p$. Note that currently available combination therapy design techniques are model free and require multiple experimental iterations to arrive at the optimal strategy [8–15]. This article considers model based combination therapy design over multiple models of tumor and normal cell lines.

We utilize the constrained structure of tumor proliferation models to design an accelerated lexicographic search algorithm for generating the optimal solution. For comparison purposes, we also designed two suboptimal search algorithms based on evolutionary algorithms and hill-climbing based techniques. We test the performance of our algorithms on synthetic models and models generated from Genomics of Drug Sensitivity in Cancer (GDSC) database [16]. Utilizing the model structure in the search process allows us to arrive at optimal or close to optimal solutions in significantly lower number of steps as compared to exhaustive search. The article also presents the theoretical analysis of the expected number of comparisons required for the proposed optimal Lexicographic search that compare favorably with the observed number of computations.

The paper is organized as follows: The model representation is discussed in "Methods" section. "Algorithms" section discusses the proposed lexicographic search algorithm along with suboptimal Genetic algorithm and Hill climbing approaches. "Results and discussion" section presents the results followed by Conclusions in "Conclusions" section.

Methods

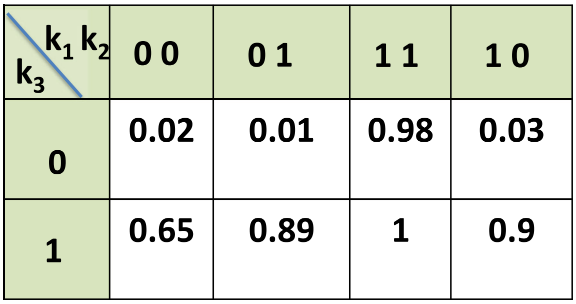
Model type

In this section, we provide a brief review of the model used to represent each spatial tumor biopsy or normal cell line. A Probabilistic Target Inhibition Map (PTIM) model provides an estimate of sensitivity for all possible target inhibitions. Consider the example PTIM model with 3 targets k_1, k_2, k_3 shown in Fig. 1 where the values for each cell represent the sensitivity corresponding to that specific inhibition. For instance, inhibition of k_3 alone will produce a sensitivity of 0.75. Since the commonly used targeted drugs inhibit oncogenes, we consider the targets to be all oncogenes and inhibition of more oncogenes can only cause the sensitivity to remain same or increase. For instance, since the inhibition of k_3 alone produces a sensitivity of 0.75, all supersets of that inhibition ($[k_3, k_1]$, $[k_3, k_2]$, $[k_3, k_1, k_2]$) will have sensitivity $\geq$ 0.75. Similarly, any subset of known inhibition will have sensitivity less than or equal to the observed value. Based on these two biological constraints and limited drug perturbation experiments, we can arrive at an inferred PTIM model that can provide an estimate of sensitivity for all possible target inhibitions. The details of the model are available at [4–6] along with biological validation at [17]. Note that a PTIM can also be approximately represented as a tumor proliferation circuit as shown in Fig. 2 where the tumor proliferation can be restricted by inhibiting at least one series block. For instance, inhibition of the $[k_1, k_2]$ block will provide a sensitivity of 0.95 whereas inhibition of k_3 will provide a sensitivity of 0.75. Inhibiting more than the minimum will produce higher sensitivities that are given by the original map shown in Fig. 1.

Structure of tumor and normal cell models

Based on the previously discussed model structure, each of the k tumor models will be represented as a probabilistic target inhibition map that can also be approximated by a circuit representation of series of parallel blocks as shown in Fig. 3.

In Fig. 3, the number of blocks for models MT_i for $i = 1, \cdots, k$ and MN_j for $j = 1, \cdots, p$ are denoted by n_{Ti} and n_{Nj} respectively. Every model is composed of five blocks connected in series. Each block, b, contains a set of targets



k_3 \ k_1 k_2	0 0	0 1	1 1	1 0
0	0.02	0.01	0.98	0.03
1	0.65	0.89	1	0.9

Fig. 1 PTIM example. An example PTIM model with 3 targets k_1, k_2 and k_3

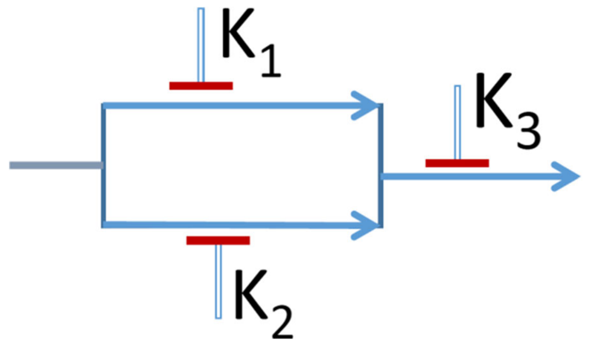


Fig. 2 PTIM circuit. A circuit representation of a PTIM model in Fig. 1

T_{bi} (up to a maximum of 5 targets), that are connected in parallel. Thus each model can have up to 25 targets.

Optimization objectives

For optimization, we consider both the worst case and best expected scenario. Let $O(MT_i, \phi)$ denote the sensitivity of Tumor model i for $i = 1, \cdots, k$ with inhibition ϕ. Let $O(MN_i, \phi)$ denote the sensitivity of normal model i for $i = 1, \cdots, p$ with inhibition ϕ.

Worst case optimization (WCO): We desire high sensitivity over the tumor cell lines and low sensitivity over the normal cells which can be formulated in the worst case scenario as maximizing the minimum sensitivity over the tumorous cells while maintaining the maximum sensitivity over the normal cells below a certain threshold θ_1.

i.e. $\max_{\phi}(\min_i [O(MT_i, \phi)])$ while $\max_i [O(MN_i, \phi)] \leq \theta_1$

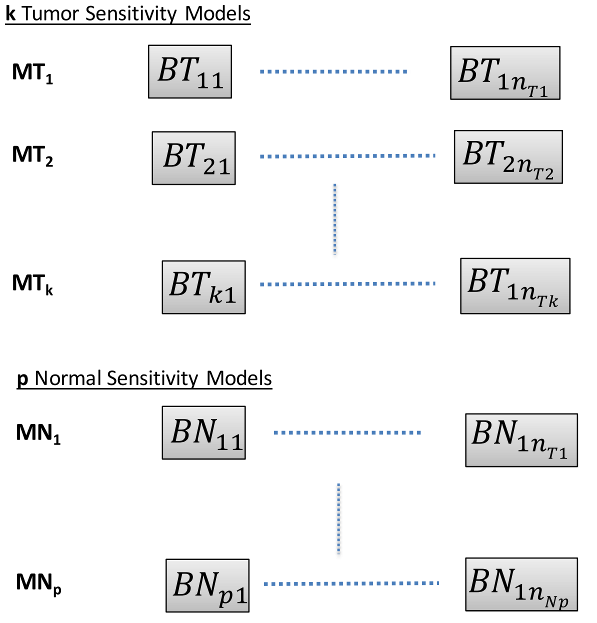


Fig. 3 PTIM block diagram. Representation of *k* tumor models and *p* normal models as series of parallel target blocks

Best expected optimization (BEO): In this scenario, our goal will be to maximize the average sensitivity over the tumorous cells while maintaining the average sensitivity over the normal cells below a threshold θ_2 i.e. $\max_{\phi}(\overline{O(MT_i, \phi)})$ while $\overline{O(MN_i, \phi)} \leq \theta_2$.

Algorithms

Lexicographic search algorithm

In order to find the optimal target set for our problem, we can exhaustively search through all possible target combinations for a given toxicity threshold. Normally, for T targets this would require searching through 2^T combinations, which is not computationally feasible for large T. However, we can utilize the monotonic relationships of PTIM models to our advantage to reduce the number of search steps. Given a set of targets S_1 whose toxicity (i.e. sensitivity over normal cell lines) is greater than a given threshold θ_1, then all possible supersets of S_1 will also have a toxicity $\geq \theta_1$ and thus, there is no need to search through the supersets of S_1. Note that this is only valid when all the targets are oncogenes.

To take advantage of this property, we perform a branching Lexicographical Search of the solution space. We can view the solution space as a directed graph where each node of the graph is our target set represented as a binary string with T bits. Each edge of the graph corresponds to turning on one bit to the right of the least significant bit, creating a superset of that node. If the toxicity at a node exceeds the threshold, then there is no need to continue along the associated edges and we should

Algorithm 1 Lexicographic search

```
Require: Set of Targets $S_0$, Toxicity Threshold $\theta_1$
Ensure: Ideal Target Set $S_b$
  Initializing population:
  $D$ = Starting set of Drugs or Targets
  $S_b = S_0$
  for Each target $t$ right of the LSB in $S_0$ do
    $S_t$ = Turn on target $t$ in $S_0$
    if tox($S_t$) < $\theta_1$ then
      $S_t$ = LexicographicSearch($S_t, \theta_1$)
      if sens($S_t$) > sens($S_b$) then
        $S_b = S_t$
      end if
    end if
  end for
  return $S_b$
```

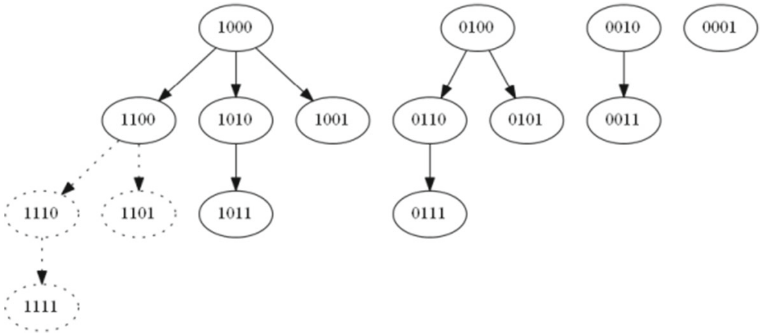


Fig. 4 Lexicographic search example. Lexicographical Search for 4 Targets. Utilizing the superset rule, sets surrounded by *dotted lines* are excluded from the search process when toxicity of [1100] $\geq \theta_1$

instead trace back to the previous node. A recursive algorithm to perform this search is shown in Algorithm 1. A demo using four targets is shown in Fig. 4. Note that in Fig. 4, we are assuming that the sensitivity over normal cell lines exceed the given threshold for target set [1100] and thus its supersets consisting of [1110], [1101] and [1111] marked by dotted lines are excluded from the search process.

Lexicographic search analysis

In this section, we consider the stochastic analysis of the proposed Lexicographic Search to generate the expected number of searches. Let $CDF(l,\theta) = f(l,\theta)$ denote the probability that a normal cell model will have sensitivity $\leq \theta$ for a random inhibition of l targets. We then define $g_P(l,\theta)$ as the probability that we will not exceed sensitivity θ while targeting l random inhibitions in a set of P cells. For the worst case scenario, $g_P^{wco}(l,\theta)$ is the probability that the maximum sensitivity over the normal cells is below threshold θ. Considering, independence of the normal cell sensitivities, we have:

$$\begin{aligned} g_P^{wco}(l,\theta) &= Pr(max_i(O(MT_i)) \leq \theta) \\ &= Pr(O(MT_1) \leq \theta, O(MT_2) \leq \theta, \cdots, \\ &\quad O(MT_P) \leq \theta) \\ &= Pr(O(MT_1) \leq \theta) \bullet Pr(O(MT_2) \leq \theta) \cdots \\ &\quad \bullet Pr(O(MT_P) \leq \theta) \\ &= f(l,\theta)^P \end{aligned} \tag{1}$$

For the best expected scenario, let us consider the probability density function of observing a sensitivity of θ after l inhibitions $PDF(l,\theta) = \frac{\partial f(l,\theta)}{\partial \theta}$. Let X denote the random variable with $PDF(l,\theta)$ and Z denote the sum of P such random variables. The probability density function of Z denoted by $q_P(l,\theta)$ can be calculated by repeatedly convolving $PDF(l,\theta)$ with itself for $P-1$ times and is given by

$$q_P(l,\theta) = PDF(l,\theta) * PDF(l,\theta) \cdots * PDF(l,\theta)$$

Let Y denote the random variable denoting the average sensitivity over P cell lines with l random inhibitions. The probability density function of Y is given by

$$pdf_Y(\theta) = h_p(l,\theta) = P * q_P(l, P\theta) \tag{2}$$

We can then estimate the cumulative density function of Y, $g_P^{beo}(l,\theta)$ by integrating across θ:

$$g_P^{beo}(l,\theta) = \int_0^{\theta} h_P(l,u)du$$

Expected savings

Define A_i to denote the event that the sensitivity over P normal models with i random inhibitions $\geq \theta$, i.e. $Pr(A_i) = 1 - g_P(i,\theta)$. Let L_i denote the event of stoping at level i of the Lexicographic Search where i represents the number of bits we are searching through. The probability of event L_i is given by:

$$\begin{aligned} P(L_i) &= P\left(A_i \cap A_{i-1}^C \cap A_{i-2}^C \cdots \cap A_1^C\right) \\ &= P\left(A_i | \bigcap_{j=1}^{i-1} A_j^C\right) P\left(\bigcap_{j=1}^{i-1} A_j^C\right) \end{aligned} \tag{3}$$

We note that:

$$P\left(A_i | \bigcap_{j=1}^{i-1} A_j^C\right) = 1 - P\left(A_i^C | \bigcap_{j=1}^{i-1} A_j^C\right) \tag{4}$$

By applying Bayes' theorem, we can simplify further:

$$\begin{aligned} P\left(A_i^C | A_{i-1}^C \cap A_{i-2}^C \cap \ldots A_1^C\right) &= \frac{P\left(A_i^C\right) P\left(\bigcap_{j=1}^{i-1} A_j^C | A_i^C\right)}{P\left(\bigcap_{j=1}^{i-1} A_j^C\right)} \\ &= \frac{P\left(A_i^C\right)}{P\left(\bigcap_{j=1}^{i-1} A_j^C\right)} \end{aligned} \tag{5}$$

By combining Eqs. 3, 4 and 5, we have:

$$P(L_i) = g(i-1,\theta) - g(i,\theta) \tag{6}$$

To find the expected savings, we note that by stopping at L_i, we search through $\sum_{j=0}^{i} \binom{T}{j}$ combinations. Thus, the expected savings $E(S)$ is given by :

$$E(S) = \sum_{i=1}^{T} P\left(A_i \cap A_{i-1}^C \cap \ldots A_1^C\right) \left[2^T - \sum_{j=0}^{i} \binom{T}{j}\right] \tag{7}$$

$$= \sum_{i=1}^{T} \left[g(i-1,\theta) - g(i,\theta)\right] \left[2^T - \sum_{j=0}^{i} \binom{T}{j}\right] \tag{8}$$

Genetic algorithm based search

Pareto optimality

We consider a multi-objective optimization scenario where we maximize sensitivity over tumor cells and minimize sensitivity over normal cells. For worst case optimization scenario, if therapies ϕ_1 and ϕ_2 satisfy the following relation: $min_{1 \le i \le k}[O(MT_i, \phi_1)] \ge min_{1 \le i \le k}[O(MT_i, \phi_2)]$ and $max_{1 \le i \le p}[O(MN_i, \phi_1)] \le max_{1 \le i \le p}[O(MN_i, \phi_2)]$ with either $min_{1 \le i \le k}[O(MT_i, \phi_1)] > min_{1 \le i \le k}[O(MT_i, \phi_2)]$ or $max_{1 \le i \le p}[O(MN_i, \phi_1)] < max_{1 \le i \le p}[O(MN_i, \phi_2)]$, then therapy ϕ_1 is considered to dominate ϕ_2 from the multi-objective Pareto sense. The therapies that are not dominated by any other therapy will form the Pareto efficient front.

Algorithm

Genetic Algorithms (GA) are inspired by evolutionary theory where strong species have a higher opportunity to pass their genes to offspring via reproduction and weaker chromosomes are eliminated by natural selection [18, 19]. Each generation or population consists of diverse individuals or chromosomes and in our Genetic Algorithm based Combination Therapy design (GACT), each therapy ϕ is regarded as a chromosome comprised of different target inhibitions. These target inhibitions are binary variables with values of 0 (non-inhibited) or 1 (completely inhibited). In order to select the best solutions (therapies) for the next generation, the fitness of each solution is computed. The therapies with the best fitness (our Pareto front) will be selected as the parents of the next generation. During each reproduction process, crossover and mutation operators are applied for the purpose of generating new solutions from existing ones. Mutation is performed by randomly flipping inhibition values of our targets. Crossover is performed by randomly picking values between two different target sets. For example, if we take the two target sets $[a_1, a_2, a_3]$ and $[b_1, b_2, b_3]$ a crossover between them can be performed by considering:

$$[c_1, c_2, c_3] = [a_1, b_2, a_3]$$

Based on our starting set of targets (M), we form the initial population P_0 of N random target inhibition profiles. After calculating the fitness functions for the existing population, we calculate different Pareto front layers according to their dominance relationships. The top Pareto optimal points are selected to pairwise conduct crossover and mutations to form offsprings. Here we have set the number of offspring to be at least twice the number of points in our Pareto fronts with a minimum of $Offmin = 1000$ and a maximum of $nOffmax = 15,000$ offsprings. After merging these offsprings with their parent population P_{t-1}, we extract top N therapies to generate population P_t. We iterate our algorithm until we have achieved *totalG* generations or the number of offspring is greater than *nOffmax*. Note that evolutionary algorithms like GA will not guarantee convergence of the Pareto front but the performance of our therapies will improve if the Pareto front moves towards our desired direction with subsequent GA iterations. The detailed procedure for multi-objective GACT is shown as Algorithm 2. Figure 5 illustrates how the algorithm moves our pareto front towards better solutions across successive iterations. After running the GACT, we consider the final Pareto front and pick the target set that provides the maximum sensitivity over tumor cell lines when the toxicity is below threshold θ_1.

Algorithm 2 Multi-objective GACT

```
Require: Set of Drugs/Targets $N$, $Q$, $totalG$
Ensure: Population $P_t$
  Initializing population:
  $D$ = Starting set of Drugs or Targets
  for $i = 1$ to $N$ do
    $\phi_i$ = Random($D$, $Q$)
    Add $\phi_i$ to $P_0$
  end for
  $f_0$ = EvaluateFitness($P_0$)
  for $t = 1$ to $totalG$ do
    ComputePareto front:$front_t$ = ParetoFront($f_{t-1}$)
    Perform Crossover and Mutation on the Pareto Front
    crossover($front_t$, $rate_c$)
    mutation($front_t$, $rate_m$)
    Recalculate Fitness
    $f_t$ = EvaluateFitness($P_{t-1} \cup front_t$)
    if $f_t > nOffmax$ then
      Break
    end if
  end for
  return $P_t$
```

Random restart hill climbing

We consider an additional suboptimal algorithm based on Hill Climbing to search the target space. Hill Climbing is an iterative method for finding the local maximum for any

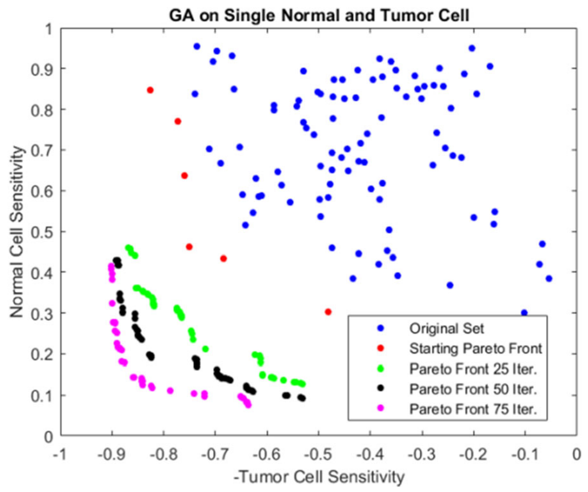


Fig. 5 GACT example. Pareto fronts converge for subsequent genetic algorithm iterations

Algorithm 3 Random restart hill climbing

```
Require: Toxicity Threshold $\theta_1$, $maxIter$
Ensure: Ideal Target Set $S_b$
  $S$ = RandomTargetSet()
  for $n = 1$ to $maxIter$ do
    $S_N$ = FindNeighbors($S$)
    if tox($S$) $> \theta_1$ then
      $S_1$ = minToxicity($S_N$)
    else
      $S_1$ = maxSensitivity($S_N$, $\theta_1$)
    end if
    if $S_1 = S$ then
      if sens($S_1$) $>$ sens($S_b$) then
        $S_b = S_1$
      end if
      $S$ = RandomTargetSet()
    else
      $S = S_1$
    end if
  end for
  return $S_b$
```

arbitrary function. Given a starting point, the algorithm considers all the nearest neighbors and then selects the neighbor that provides the best solution for the given optimization criteria. These steps are subsequently repeated until there are no neighbors that provide a better solution or a maximum number of iterations have been reached. While simple and effective in finding local optimum, Hill Climbing will rarely find the global optimum for non-convex functions. In order to overcome this handicap, we will randomly restart our search to a new random position whenever the algorithm converges to a local optimum.

In our case, the starting criteria will be a random set of targets chosen using latin hypercube numbers and each neighbor will be created by inhibiting or un-inhibiting a single target. As shown in Algorithm 3, our optimization criteria will change depending on the toxicity of our current set. If the toxicity is greater than the given threshold then we choose the neighbor with the least toxicity, otherwise we pick the neighbor with the highest sensitivity whose toxicity is below the threshold. When no improvements can be found among the neighbors we randomly choose a new set of targets. This is repeated until we have completed *maxIter* iterations.

Results and discussion

To evaluate the performance of our algorithms, we considered both synthetic models and models based on experimental datasets.

Synthetic model generation

The synthetic models are simulated using a proliferation network structure based on probabilistic target inhibition maps [5, 7]. Each cellular pathway, i, representing either a tumor or normal cell model is modelled by connecting a set of blocks in series. The number of blocks for models MT_i for $i = 1, \cdots, k$ and MN_j for $j = 1, \cdots, p$ are denoted by n_{Ti} and n_{Nj} respectively. Within each block, b, contains a set of targets T_{bi} (up to a maximum of 5 targets), that are connected in parallel. Since the targets are in parallel, the effective inhibition for each block given a set of target inhibitions, ϕ, is the minimum inhibition of the given targets within the block. Thus, the effective inhibition of block b in model MT_i with target inhibition ϕ is given by $\lambda(MT_i, b, \phi) = \min(\forall \phi \in T_{bi})$.

Each block is also given a score, S_{bi}, randomly using a uniform distribution with a minimum of 0.5 and maximum of 1. Finally, the overall sensitivity of the pathway can be computed using the following equation where we assume independence between the series blocks:

$$Sensitivity(MT_i, \phi) = 1 - \prod_{b=1}^{N_{Ti}} (1 - S_{bi}\lambda(MT_i, b, \phi))$$

Similary for normal cells:

$$Sensitivity(MN_i, \phi) = 1 - \prod_{b=1}^{N_{Ni}} (1 - S_{bi}\lambda(MN_i, b, \phi))$$

A representation of k tumor and p normal models as series of parallel target blocks is shown in Fig 3.

The synthetic model set consists of a total of 1000 synthetic pathways, 500 normal and 500 cancerous. A total of 25 targets are examined and all targets are equiprobable

Table 1 Cell lines used in GDSC dataset

Breast	Lymphoma
BT474	A3KAW
COLO824	A4FUK
DU4475	BC1
EVSAT	CTB1
HCC1187	DOHH2
HCC2157	HT
HCC2218	KARPAS422
MFM223	MC116
MRKnu1	RL
OCUBM	TUR

Table 2 Worst case optimization tumor cell sensitivity using GACT for sythetic models

		rate$_c$			
		0.6	0.7	0.8	0.9
rate$_m$	0.02	0.5826	0.5891	0.5933	0.5643
	0.05	0.5903	0.5842	0.5877	0.5753
	0.1	0.5770	0.5829	0.5757	0.5728
	0.2	0.5938	0.5923	0.5848	0.5812
	0.5	0.5669	0.5673	0.5762	0.5796

Table 3 WCO GACT for synthetic models with varying nGen

nGen	*Sensitivity*
100	0.5938
200	0.6009
300	0.6027
400	0.6094
500	0.6074

Table 4 Hill climbing BEO performance with varying number of iterations

maxIter	*Sensitivity*
10000	0.8029
15000	0.8136
20000	0.8214
25000	0.8248
30000	0.8275

Table 5 LS worst case optimization synthetic data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9934	0.9828	0.9633	0.9681	0.9096
	2	0.9867	0.9603	0.9099	0.9255	0.7852
	3	0.9794	0.9454	0.8836	0.8110	0.7331
	4	0.9681	0.9255	0.8487	0.7728	0.6858
	5	0.9607	0.9005	0.8113	0.7194	0.6157

Table 6 GACT worst case optimization synthetic data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9934	0.9828	0.9633	0.9384	0.9096
	2	0.9862	0.9599	0.9085	0.8594	0.7843
	3	0.9772	0.9439	0.8811	0.8007	0.7268
	4	0.9647	0.9200	0.8430	0.7679	0.6717
	5	0.9561	0.8938	0.8056	0.7007	0.6097

Table 7 Hill climbing worst case optimization synthetic data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9926	0.9817	0.9614	0.9369	0.9086
	2	0.9805	0.9550	0.9031	0.8449	0.7635
	3	0.9625	0.9270	0.8650	0.7929	0.7016
	4	0.9402	0.8923	0.8089	0.7341	0.6241
	5	0.9137	0.8514	0.7630	0.6573	0.5536

Table 8 LS worst case optimization GDSC data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9792	0.9536	0.9218	0.8505	0.8282
	2	0.9601	0.9226	0.8718	0.7853	0.7556
	3	0.9326	0.8767	0.8207	0.7017	0.6270
	4	0.9163	0.8280	0.7405	0.6126	0.4880
	5	0.9045	0.7978	0.6809	0.5535	0.4539

Table 9 GACT worst case optimization GDSC data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9789	0.9536	0.8738	0.7532	0.6056
	2	0.9598	0.9124	0.8121	0.6900	0.4951
	3	0.9320	0.8688	0.8062	0.6401	0.4843
	4	0.9152	0.8210	0.7130	0.5525	0.3458
	5	0.9028	0.7905	0.6551	0.4735	0.3348

Table 10 Hill climbing worst case optimization GDSC data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9788	0.9390	0.8859	0.8489	0.8230
	2	0.9588	0.8762	0.8163	0.7399	0.7074
	3	0.9357	0.8283	0.7283	0.6215	0.5724
	4	0.9148	0.7773	0.6533	0.5286	0.4683
	5	0.8959	0.7422	0.5936	0.4584	0.3908

Table 11 LS best expected optimization synthetic data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9934	0.9828	0.9633	0.9384	0.9096
	2	0.9916	0.9756	0.9468	0.9170	0.8744
	3	0.9901	0.9734	0.9461	0.9110	0.8698
	4	0.9883	0.9702	0.9355	0.9049	0.8662
	5	0.9872	0.9668	0.9305	0.8984	0.8596

Table 12 GACT best expected optimization synthetic data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9934	0.9828	0.9632	0.9384	0.9095
	2	0.9912	0.9755	0.9451	0.9156	0.8741
	3	0.9889	0.9716	0.9443	0.9102	0.8664
	4	0.9861	0.9679	0.9325	0.9024	0.8640
	5	0.9830	0.9629	0.9270	0.8945	0.8549

Table 13 Hill climbing best expected optimization synthetic data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9923	0.9817	0.9621	0.9377	0.9069
	2	0.9853	0.9708	0.9409	0.9085	0.8524
	3	0.9779	0.9597	0.9305	0.8900	0.8496
	4	0.9684	0.9485	0.9138	0.8783	0.8268
	5	0.9603	0.9361	0.9016	0.8628	0.8123

Table 14 LS best expected outcome GDSC data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9792	0.9585	0.9315	0.8681	0.8572
	2	0.9783	0.9615	0.9360	0.8909	0.8813
	3	0.9737	0.9568	0.9366	0.8910	0.8653
	4	0.9735	0.9511	0.9263	0.8851	0.8508
	5	0.9736	0.9498	0.9219	0.8826	0.8544

Table 15 GACT best expected optimization GDSC data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9789	0.9584	0.8916	0.8474	0.8129
	2	0.9781	0.9611	0.9357	0.8609	0.8672
	3	0.9734	0.9478	0.8996	0.8709	0.8246
	4	0.9732	0.9502	0.8975	0.8771	0.8328
	5	0.9733	0.9402	0.9124	0.8733	0.8407

Table 16 Hill climbing best expected optimization GDSC data

		nNormal				
		1	2	3	4	5
nTumor	1	0.9777	0.9447	0.9030	0.8600	0.8464
	2	0.9738	0.9360	0.8995	0.8589	0.8286
	3	0.9734	0.9369	0.8956	0.8543	0.8267
	4	0.9731	0.9370	0.8956	0.8501	0.8152
	5	0.9731	0.9380	0.8934	0.8502	0.8177

Table 17 Computational complexity synthetic data

	WCO			BEO		
	Searches	Avg. Time (s)	Searches per s	Searches	Avg. Time (s)	Searches per s
LS	1033500	132.2	7814	1033500	126.3	7934
GACT	501000	0.6052	827825	501000	0.5806	862900
HC	380130	20.11	18902	378940	16.69	22705

in both the cancer and normal pathways. We group the pathways into 100 groups where each group has 5 normal pathways and 5 cancerous pathways. From every group, we consider *nNormal* normal pathways and *nTumor* cancerous pathways.

GDSC data

In order to test our algorithms on biological functional data, we have utilized the GDSC database [16] to generate a set of PTIM models for 20 different cell lines. These cell lines were segregated into groups of 10, the first group is composed of breast-cancer cell lines and the second group is B-cell lymphoma cancer cell lines. A list of the cell lines is shown in Table 1. For each of the cell lines, we considered the IC50 values for 32 drugs and combined with the corresponding drug panels generated a PTIM model. The drug panels contained the K_d values for 404 targets and 62 of these targets were found to correspond to the PTIM model of at least one of the cell lines, 42 targets in the breast cell lines, 49 in the lymphoma and 27 targets where found in both the breast and lymphoma cell lines.

For the generation of sets of tumor and normal models, we generate 100 groups and for each group we randomly assign one type of cell lines (breast or lymphoma) to be the *Normal* cells and the other type to be the *Tumor* cells. We then randomly pick *nNormal* cell lines from the corresponding group and assign them as *Normal* cell models. Likewise, we pick *nTumor* random cell lines from the other group and assign them as our *Tumor* cell models.

Performance comparisons

For both the synthetic and GDSC cases, we select the solution that provides the maximum cancer cell sensitivity while keeping the normal cell sensitivity below a threshold of $\theta = 0.1$ using both the best expected scenario and the worst case optimization method. The maximum cancer sensitivity is then averaged across all 100 groups.

Table 18 Computational complexity GDSC data

	WCO			BEO		
	Searches	Avg. Time (s)	Searches per s	Searches	Avg. Time (s)	Searches per s
LS	879	0.095	9253	2032	0.1317	15429
GACT	146430	4.32	33896	30850	1.01	30433
HC	300650	13.82	21749	299580	11.41	26256

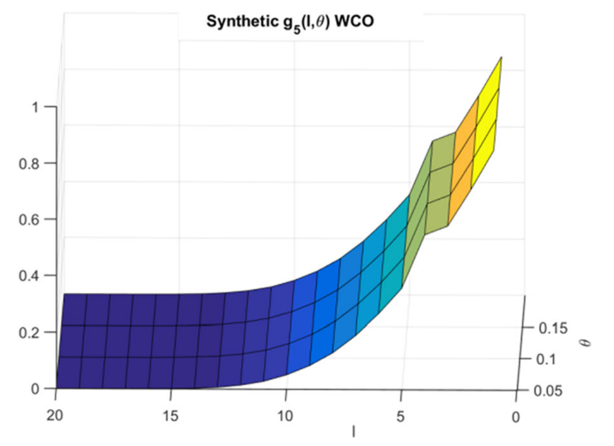


Fig. 6 CDFs for synthetic cells and WCO

GACT parameter selection

In order to find an optimal $rate_m$ and $rate_c$ for the GACT, we set $nTumor, nNormal = 5$ and $totalG = 100$. We then repeated the WCO GACT on the synthetic dataset, varying either $rate_c$ or $rate_m$ on each run. The results are shown in Table 2. Based on these results, we have selected a $rate_m$ of 0.2 and $rate_c$ of 0.6. We then start varying *nGen* and the corresponding results are recorded in Table 3. From these results we can see that past 400 iterations, there is no significant improvement in the GACT.

Hill climbing parameter selection

We perform a similar operation on the Hill Climbing algorithm where we keep $nTumor = nNormal = 5$ and vary *nIter* and measuring the best expected outcome in contrast to worst case optimization in previous case. The

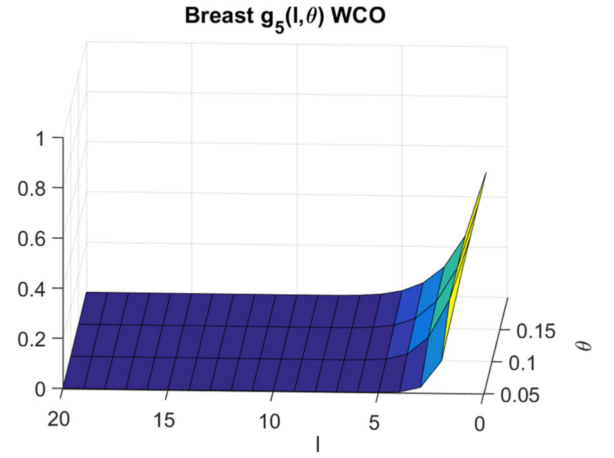


Fig. 7 CDFs for breast cells and WCO

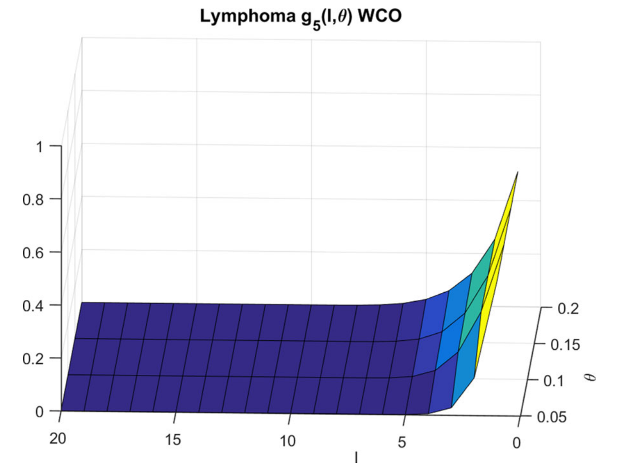


Fig. 8 CDFs for lymphoma cells and WCO

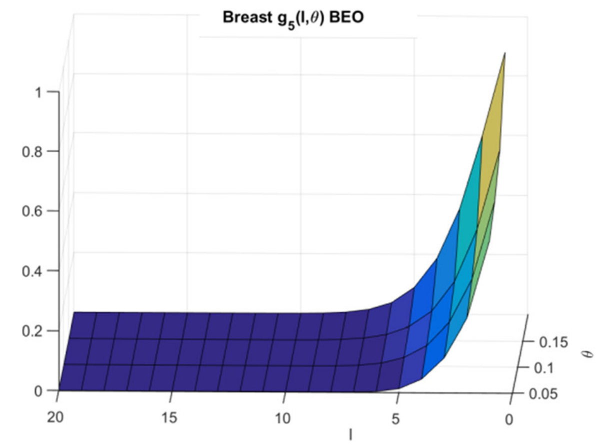


Fig. 10 CDFs for breast cells and BEO

results are shown in Table 4. Since, there are no significant changes in performance after 15,000 iterations, we use *maxIter* = 15000 for running our Hill Climbing algorithm.

WCO results

The results for worst case optimization for the three approaches on synthetic and biological models are shown in Tables 5, 6, 7, 8, 9 and 10. For the synthetic dataset, we achieved minimum cancer sensitivities > 0.60 using the LS (Table 5) and GACT (Table 6) algorithms while the Hill Climbing algorithm could only achieve a sensitivity of 0.55 (Table 7). Since the LS algorithm is the optimal approach, it produces the best performance followed closely by the GACT algorithm for synthetic dataset. However, for the GDSC dataset, the GACT was only able to achieve a sensitivity of 0.33 which is significantly worse than the LS which was able to achieve a sensitivity of 0.45. For the GDSC dataset, it appears that GACT is close to the optimal solution for smaller number of normal cells and increasing the number of normal cells reduces its performance (Table 9). Hill Climbing has reasonable performance for GDSC data based models (Table 10).

BEO results

The results for best expected optimization scenario for the three algorithms on synthetic and biological models are shown in Tables 11, 12, 13, 14, 15 and 16. For the synthetic dataset, we achieved expected sensitivities over cancer models higher than 0.85 using the LS (Table 11) and GACT (Table 12) algorithms and > 0.81 for Hill Climbing Approach (Table 13). For the GDSC based models, performance close to LS (Table 14) is observed with GACT (Table 15) followed by Hill Climbing (Table 16).

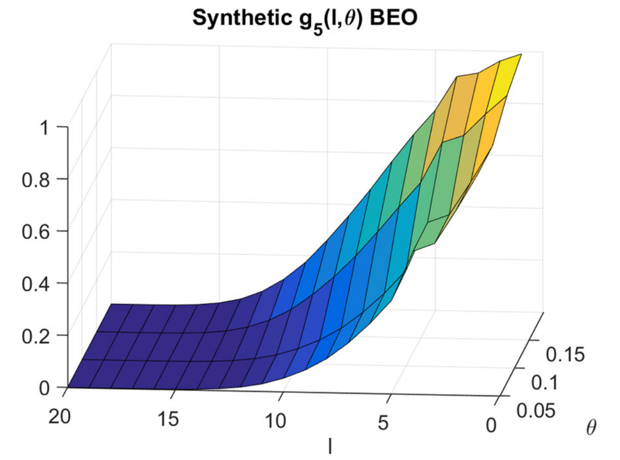


Fig. 9 CDFs for synthetic cells and BEO

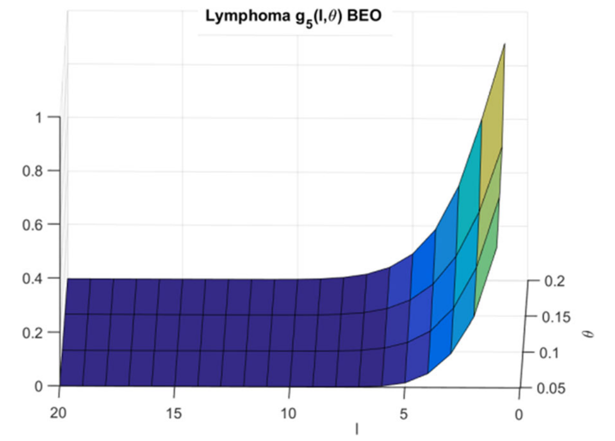


Fig. 11 CDFs for lymphoma cells and BEO

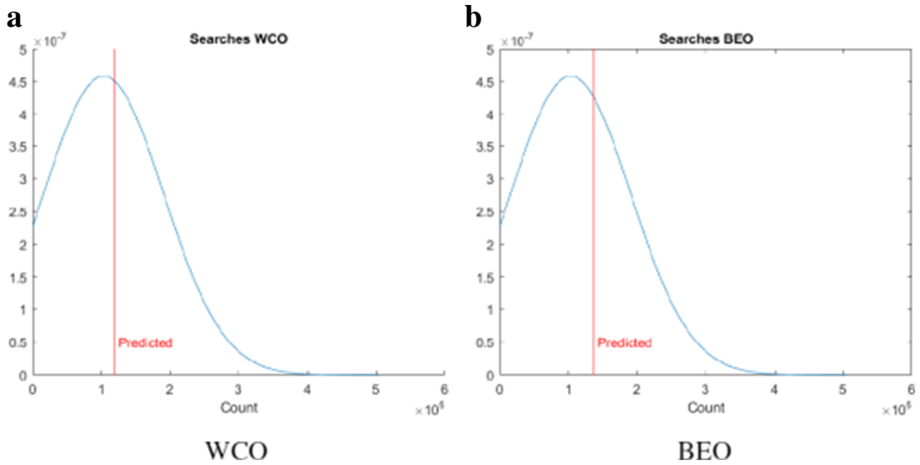


Fig. 12 Estimated vs. actual searches for synthetic dataset

Computational complexity

Tables 17 and 18 shows the average number of searches and runtime for each algorithm for synthetic and GDSC datasets respectively. All results are for *nNormal* = *nTumor* = 5. Each program is written using MATLAB and ran on a Inspiron 15 laptop with a Core i5-6300HQ processor with 8 GB of RAM. For the synthetic models, GACT was the fastest, taking less than a second for both WCO and BEO. The second fastest was the Hill Climbing (HC) algorithm that takes around 20 s per group while the LS algorithm takes the longest at over 125 s per group. It should be noted that the GACT is able to perform a proportionally larger number of searches in a shorter amount of time than both the LS and HC algorithm. This is because the GACT uses a small number of iterations but performs a large number of searches per iteration and is thus able to be vectorized more than the other algorithms.

In contrast for the GDSC dataset, the LS algorithm performs the fastest due to the small number of searches required. The GACT is now the second fastest and HC is the slowest. It should be noted that the GACT performs significantly slower in the GDSC dataset than the synthetic case. This is because we are unaware of the underlying circuit models of the PTIMs in the GDSC dataset as the sensitivities are computed using a lookup table and cannot be vectorized as efficiently as the synthetic data. In the next section, we explain the smaller number of searchers required for the LS approach for GDSC data as compared to Synthetic data based on the the specific structures of $g(l, \theta)$.

Estimated number of searches

In this section, we empirically generate the distributions of searches required for LS technique for random sets of normal and cancerous cells for both synthetic and GDSC

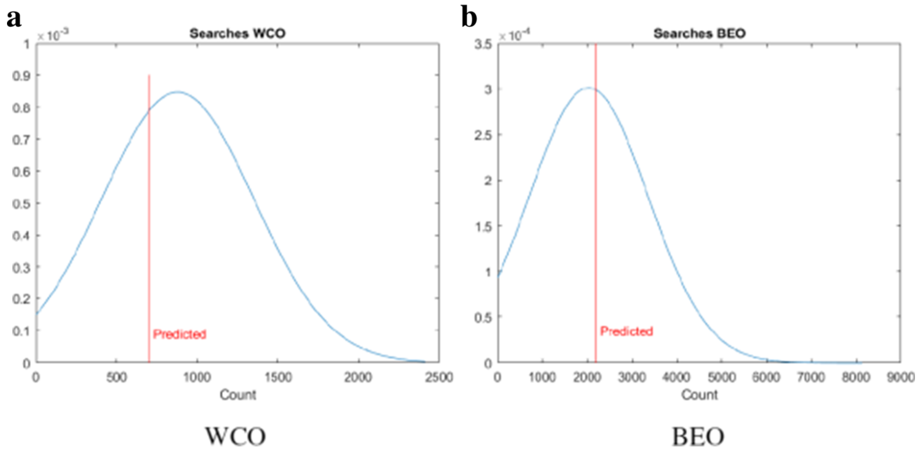


Fig. 13 Estimated vs. actual searches for GDSC dataset

data and compare them to theoretical estimates. In order to generate $f(l,\theta)$, we randomly select l targets and record the measured sensitivity for each cell line following inhibition of the selected l targets. This process is repeated 300,000 times for $l = 1$ to *nTargets* to generate the entire CDF. The $g_P^{wco}(l,\theta)$ distributions for Synthetic, Breast cancer and Lymphoma cell lines are shown in Figs. 6, 7 and 8 respectively.

The corresponding $g_P^{beo}(l,\theta)$ distributions for Synthetic, Breast cancer and Lymphoma cell lines are shown in Figs. 9, 10 and 11 respectively.

We then use the theoretical estimate outlined in "Lexicographic search analysis" section to predict the number of searches required by the LS algorithm. In all the scenarios, we are assuming $P = 5$. The predicted value is plotted against the actual value fitted to a normal distribution and the plots are shown in Fig. 12 for synthetic data and Fig. 13 for the GDSC data. We note that the theoretical estimates are very close to the mean of the distributions confirming our savings estimate.

To explain the limited number of searches required for GDSC models as compared to synthetic models, note that $g_P^{wco}(l,\theta)$ and $g_P^{beo}(l,\theta)$ have gradual decreases with l for synthetic data as shown in Figs. 6 and 9 respectively whereas $g_P^{wco}(l,\theta)$ and $g_P^{beo}(l,\theta)$ for GDSC models have sharp decreases with l as shown in Figs. 7, 8, 10 and 11. Due to the sharp change in the probabilities, the expected number of searches required for LS for GDSC models (in the 600 to 2200 range) is significantly lower than the expected number of searches required for the synthetic models (in the 10^6 range).

Conclusions

In this paper, we have formulated the combination therapy design problem of maximizing efficacy while minimizing toxicity as an algorithmic search problem to find the optimal target set that maximally inhibit spatially heterogeneous cancer cell models while maintaining the effect along multiple normal cell models below a certain threshold. Our cell proliferation models are based on probabilistic target inhibition map (PTIM) framework [4–7] that consists of a series of blocks where each block contains a set of targets connected in parallel. To find the ideal target inhibition profile, we proposed a lexicographic search method to effectively search through all possible solutions. This method takes advantage of the properties of the PTIM to significantly reduce the number of potential solutions that we have to search through. We compare the performance and computational complexity of this method with other commonly used algorithms such as Genetic algorithms and Hill Climbing. We demonstrated the effectiveness of our algorithms using both synthetic models and models generated from Drug Sensitivity of Cancer Database. A theoretical analysis of the expected number of steps required by the Lexicographic Search process is included that was shown to provide a close approximation to actual observed search steps.

Abbreviations

BEO: Best expected optimization; GA: Genetic algorithm; GACT: Genetic algorithm based combination therapy; GDSC: Genomics of drug sensitivity in cancer; HC: Hill climbing LS: Lexicographic search; PTIM: Probabilistic target inhibition map; WCO: Worst case optimization

Acknowledgments

Not applicable.

Funding

Publication charges were covered by NSF grant PFI:AIR -TT 1500234. The funders had no role in study design, analysis, decision to publish, or preparation of the manuscript.

Availability of data and materials

Synthetic Data can be downloaded from the following link, http://tinyurl.com/hfq3krr, functional data analysed during the current study is available in the Genomics of Drug Sensitivity in Cancer repository, http://www.cancerrxgene.org/.

Authors' contributions

Conceived and Designed the algorithms: KM RP. Provided Biological Models: NB CK. Performed the simulation experiments: KM. Analyzed the Results: KM RP. Wrote the article: KM RP. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Consent for publication

Not applicable.

Ethics approval and consent to participate

Not applicable.

About this supplement

This article has been published as part of BMC Bioinformatics Volume 18 Supplement 4, 2017: Selected original research articles from the Third International Workshop on Computational Network Biology: Modeling, Analysis, and Control (CNB-MAC 2016): bioinformatics. The full contents of the supplement are available online at https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-18-supplement-4.

Author details

[1]Department of Electrical and Computer Engineering, Texas Tech University, 1012 Boston Ave, 79409 Lubbock, TX, USA. [2]Children's Cancer Therapy Development Institute, 97005 Portland, OR, USA.

Published: 22 March 2017

References

1. Costello JC, et al. A community effort to assess and improve drug sensitivity prediction algorithms. Nat Biotechnol. 2014. doi:10.1038/nbt.2877.
2. Wan Q, Pal R. An ensemble based top performing approach for NCI-DREAM drug sensitivity prediction challenge. PLOS ONE. 2014;9(6): e101183.
3. Barretina J, et al. The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. Nature. 2012;483(7391):603–607. Available from: http://dx.doi.org/10.1038/nature11003.
4. Pal R, Berlow N. A Kinase inhibition map approach for tumor sensitivity prediction and combination therapy design for targeted drugs. In: Proceedings of the Pacific Symposium on Biocomputing, Kohala Coast, Hawaii, USA, 3–7 January 2012. PMID: 22174290. Kohala Coast: World Scientific; 2012. p. 351–62.
5. Berlow N, Davis LE, Cantor EL, Seguin B, Keller C, Pal R. A new approach for prediction of tumor sensitivity to targeted drugs based on functional data. BMC Bioinforma. 2013;14:239.

6. Berlow N, Haider S, Wan Q, Geltzeiler M, Davis LE, Keller C, et al. An integrated approach to anti-cancer drugs sensitivity prediction. IEEE/ACM Trans Comput Biol Bioinforma. 2014. doi:10.1155/2014/873436.
7. Haider S, Pal R. Inference of tumor inhibition pathways from drug perturbation data. In: IEEE Global Conference on Signal and Image Processing (GlobalSIP). Austin: IEEE; 2013. p. 95–98. doi:10.1109/GlobalSIP.2013.6736823.
8. Borisy AA, Elliott PJ, Hurst NW, Lee MS, Lehár J, Price ER, et al. Systematic discovery of multicomponent therapeutics. Proc Nat Acad Sci USA. 2003;100(13):7977–82.
9. Wadman M. The Right Combination. Nature. 2006;439:390–401.
10. Zimmermann GR, Lehár J, Keith CT. Multi-target therapeutics: when the whole is greater than the sum of the parts. Drug Discov Today. 2007;12(1–2):34–42.
11. Zinner R, Barrett B, Popova E, Damien P, et al. Algorithmic guided screening of drug combinations of arbitrary size for activity against cancer cells. Mol Cancer Ther. 2009;8:521–32.
12. Calzolari D, Bruschi S, Coquin L, Schofield J, Feala JD, Reed JC, et al. Search algorithms as a framework for the optimization of drug combinations. PLoS Comput Biol. 2008;4(12):e1000249+.
13. Wong PK, Yu F, Shahangian A, Cheng G, Sun R, Ho CM. Closed-loop control of cellular functions using combinatory drugs guided by a stochastic search algorithm. Proc Nat Acad Sci. 2008;105(13):5105–10.
14. Yoon BJ. Enhanced stochastic optimization algorithm for finding effective multi-target therapeutics. BMC Bioinforma. 2011;12(Suppl 1):S18+.
15. Caglar MU, Pal R. A diverse stochastic search algorithm for combination therapeutics. BioMed Res Int. 2014;873436:9.
16. Yang Wea. Genomics of Drug Sensitivity in Cancer (GDSC): a resource for therapeutic biomarker discovery in cancer cells. Nucleic Acids Res. 2013;41(D1):D955—D961.
17. Grasso CS, Tang Y, Truffaux N, Berlow NE, Liu L, Debily M, et al. Functionally-defined Therapeutic Targets in Diffuse Intrinsic Pontine Glioma. Nat Med. 2015;21(nm.3855):555–559.
18. Bäck T. Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford: Oxford University Press; 1996.
19. Konak A, Coit DW, Smith AE. Multi-objective optimization using genetic algorithms: A tutorial. Reliability Eng Syst Safety. 2006;91(9):992–1007.