Nature-Inspired metaheuristic algorithms for finding optimal designs for high dimensional problems

Weng Kee WONG
Department of Biostatistics, UCLA

Meeting the Statistical Challenges in High Dimensional Data and Complex Networks
IMS NUS, Singapore
5 Feb - 16 February 2018

Outline

Outline

- Motivation from Optimal Design Problems
- Why Nature-inspired Meta-heuristic Algorithms?
- Optimal Designs via PSO and a Quick Demonstration
- Closing Thoughts

Outline

- Motivation from Optimal Design Problems
- Why Nature-inspired Meta-heuristic Algorithms?
- Optimal Designs via PSO and a Quick Demonstration
  - Standardized maximin optimal designs for enzyme kinetic models
- Closing Thoughts

Outline

- Motivation from Optimal Design Problems
- Why Nature-inspired Meta-heuristic Algorithms?
- Optimal Designs via PSO and a Quick Demonstration
  - Standardized maximin optimal designs for enzyme kinetic models
  - Extended adaptive 2-stage Simon's designs
- Closing Thoughts

Outline

- Motivation from Optimal Design Problems
- Why Nature-inspired Meta-heuristic Algorithms?
- Optimal Designs via PSO and a Quick Demonstration
  - Standardized maximin optimal designs for enzyme kinetic models
  - Extended adaptive 2-stage Simon's designs
  - Optimal designs for GLMs with mixed factors
- Closing Thoughts

Outline

- Motivation from Optimal Design Problems
- Why Nature-inspired Meta-heuristic Algorithms?
- Optimal Designs via PSO and a Quick Demonstration
    - Standardized maximin optimal designs for enzyme kinetic models
    - Extended adaptive 2-stage Simon's designs
    - Optimal designs for GLMs with mixed factors
    - Multiple-objective optimal designs for the 4-parameter logistic model
- Closing Thoughts

Outline

1.1 Background

An Illustrative Dose Response Example

$$y = \theta_0 \; + \; \theta_1 \; x \; + \; error = f^T(x)\theta \; + \; error, \quad \theta^T = (\theta_1, \theta_2)$$

$$\text{E(error)} \; = 0 \qquad \text{var(error)} = \sigma^2 \qquad N \;\; \text{i.i.d. observations}$$

On $X = [-1, 1]$, what is the optimal design for estimating

- (1) θ_0?

1.1 Background

An Illustrative Dose Response Example

$$y = \theta_0 \ + \ \theta_1 \ x \ + \ error = f^T(x)\theta \ + \ error, \quad \theta^T = (\theta_1, \theta_2)$$

$$\mathrm{E(error)} \ = 0 \qquad \mathrm{var(error)} = \sigma^2 \qquad N \ \text{ i.i.d. observations}$$

On $X = [-1, 1]$, what is the optimal design for estimating

- (1) θ_0?
- (2) θ_1?

1.1 Background

An Illustrative Dose Response Example

$$y = \theta_0 \ + \ \theta_1 \ x \ + \ error = f^T(x)\theta \ + \ error, \quad \theta^T = (\theta_1, \theta_2)$$

$$\mathrm{E(error)} \ = 0 \qquad \mathrm{var(error)} = \sigma^2 \qquad N \ \text{ i.i.d. observations}$$

On $X = [-1, 1]$, what is the optimal design for estimating

- (1) θ_0?
- (2) θ_1?
- (3) both θ_0 and θ_1?

1.1 Background

An Illustrative Dose Response Example

$$y = \theta_0 + \theta_1 x + error = f^T(x)\theta + error, \quad \theta^T = (\theta_1, \theta_2)$$

$$\mathrm{E(error)} = 0 \qquad \mathrm{var(error)} = \sigma^2 \qquad N \text{ i.i.d. observations}$$

On $X = [-1, 1]$, what is the optimal design for estimating

- (1) θ_0?
- (2) θ_1?
- (3) both θ_0 and θ_1?
- Answer for (2) comes from

$$var(\hat{\theta}_1) = \frac{\sigma^2}{\sum_{i=1}^N (x_i - \bar{x})^2} \geq \frac{\sigma^2}{\sum_{i=1}^N x_i^2} \geq \frac{\sigma^2}{N}.$$

1.1 Background

An Illustrative Dose Response Example

$$y = \theta_0 \; + \; \theta_1 \; x \; + \; error = f^T(x)\theta \; + \; error, \quad \theta^T = (\theta_1, \theta_2)$$

$$\mathrm{E(error)} \; = 0 \qquad \mathrm{var(error)} = \sigma^2 \qquad N \;\; \text{i.i.d. observations}$$

On $X = [-1, 1]$, what is the optimal design for estimating

- (1) θ_0?
- (2) θ_1?
- (3) both θ_0 and θ_1?
- Answer for (2) comes from

$$var(\hat{\theta}_1) = \frac{\sigma^2}{\sum_{i=1}^N (x_i - \bar{x})^2} \geq \frac{\sigma^2}{\sum_{i=1}^N x_i^2} \geq \frac{\sigma^2}{N}.$$

- For (2) and (3), take equal number of observations at ± 1.

1.1 Background

An Illustrative Dose Response Example

$$y = \theta_0 + \theta_1 x + error = f^T(x)\theta + error, \quad \theta^T = (\theta_1, \theta_2)$$

$\mathrm{E}(\text{error}) = 0 \qquad \text{var}(\text{error}) = \sigma^2 \qquad N$ i.i.d. observations

On $X = [-1, 1]$, what is the optimal design for estimating

- (1) θ_0?
- (2) θ_1?
- (3) both θ_0 and θ_1?
- Answer for (2) comes from

$$var(\hat{\theta}_1) = \frac{\sigma^2}{\sum_{i=1}^N (x_i - \bar{x})^2} \geq \frac{\sigma^2}{\sum_{i=1}^N x_i^2} \geq \frac{\sigma^2}{N}.$$

- For (2) and (3), take equal number of observations at ± 1.
- For (1) answer is any design with $\bar{x} = 0$.

1.2 A Typical Setup for a Design Problem

- a given compact design space X
- a parametric model with unknown parameters
- errors are normally and independently distributed
- observations have with constant variance
- a pre-determined sample size N

QUESTION

Given X, $f(x)$, N and an optimality criterion ϕ, how best to select the N points from the design space X to observe the responses y?

1.3 Approximate designs (Kiefer, 1958-1982)

Optimal Approximate Design Problem: How many points are needed to optimize the criterion? Find k

Where are the optimal design (or support) points?

Find $x_1, x_2, \cdots, x_k \in X$

What is the optimal proportion of the total observations to take at each of these points?

Find $w_1, w_2, \cdots, w_k$ such that $0 < w_i < 1,\ i = 1, 2, \cdots, k$ and $w_1 + w_{2+} \cdots + w_k = 1$.

The implemented design takes $n_i = [Nw_i]$ observations at $x_i, i = 1, 2, \cdots, k$ and rounded so that $n_1 + n_2 + \cdots + n_k = N$.

1.3 Approximate designs (Kiefer, 1958-1982)

Optimal Approximate Design Problem: How many points are needed to optimize the criterion? Find k

Where are the optimal design (or support) points?

Find $x_1, x_2, \cdots, x_k \in X$

What is the optimal proportion of the total observations to take at each of these points?

Find $w_1, w_2, \cdots, w_k$ such that $0 < w_i < 1, \ i = 1, 2, \cdots, k$ and $w_1 + w_{2+} \cdots + w_k = 1.$

The implemented design takes $n_i = [Nw_i]$ observations at $x_i, i = 1, 2, \cdots, k$ and rounded so that $n_1 + n_2 + \cdots + n_k = N$.

- Optimal Exact Design Problem finds positive integers n_i directly subject to $n_1 + n_2 + \cdots + n_k = N$.

1.4 Advantages of Approximate Designs

When the optimality criteria is formulated as a convex function of the design:

1.4 Advantages of Approximate Designs

When the optimality criteria is formulated as a convex function of the design:

- There are algorithms available for generating many types of optimal designs.

1.4 Advantages of Approximate Designs

When the optimality criteria is formulated as a convex function of the design:

- There are algorithms available for generating many types of optimal designs.
- These algorithms can be shown to converge to the optimum.

1.4 Advantages of Approximate Designs

When the optimality criteria is formulated as a convex function of the design:

- There are algorithms available for generating many types of optimal designs.
- These algorithms can be shown to converge to the optimum.
- Theory provides an equivalence theorem for confirming optimality of the generated design or assessing its proximity to the optimum using an efficiency lower bound.

1.4 Advantages of Approximate Designs

When the optimality criteria is formulated as a convex function of the design:

- There are algorithms available for generating many types of optimal designs.
- These algorithms can be shown to converge to the optimum.
- Theory provides an equivalence theorem for confirming optimality of the generated design or assessing its proximity to the optimum using an efficiency lower bound.
- Does not require an endless list of tables describing optimal design for each model, each N and each type of criterion

1.4 Advantages of Approximate Designs

When the optimality criteria is formulated as a convex function of the design:

- There are algorithms available for generating many types of optimal designs.
- These algorithms can be shown to converge to the optimum.
- Theory provides an equivalence theorem for confirming optimality of the generated design or assessing its proximity to the optimum using an efficiency lower bound.
- Does not require an endless list of tables describing optimal design for each model, each N and each type of criterion
- When the design space X has dimension 1 or 2, a simple way to verify whether a design is optimal among all designs on X is to*********draw pictures!

1.5 Optimal Approximate Designs on $X = [-1, 1]$

D-optimal designs for estimating model parameters and making inference on the mean response at a given dose level.

design criterion		linear model		quadratic model		
D-optimality	x_i	-1	1	-1	0	1
	w_i	$1/2$	$1/2$	$1/3$	$1/3$	$1/3$
Extrapolation at dose level $z = 2$	x_i	-1	1	-1	0	1
	w_i	$1/4$	$3/4$	$1/7$	$3/7$	$3/7$

1.5 Optimal Approximate Designs on $X = [-1, 1]$

D-optimal designs for estimating model parameters and making inference on the mean response at a given dose level.

design criterion		linear model		quadratic model		
D-optimality	x_i	-1	1	-1	0	1
	w_i	$1/2$	$1/2$	$1/3$	$1/3$	$1/3$
Extrapolation at dose level $z = 2$	x_i	-1	1	-1	0	1
	w_i	$1/4$	$3/4$	$1/7$	$3/7$	$3/7$

- Next, designs for nonlinear models are complicated because they depend on the parameters we want to estimate!

1.6 Locally D-optimal Designs for the Logistic Model on $X = [-1, 1]$ (Ford's PhD thesis, 1972)

$$log \, \frac{\pi(x)}{1-\pi(x)} = \theta_1 + \theta_2 x, \quad \theta^T = (\theta_1, \theta_2), \quad \theta_1 > 0 \;\; \& \;\; \theta_2 > 0.$$

1.6 Locally D-optimal Designs for the Logistic Model on $X = [-1, 1]$ (Ford's PhD thesis, 1972)

$$log \frac{\pi(x)}{1-\pi(x)} = \theta_1 + \theta_2 x, \quad \theta^T = (\theta_1, \theta_2), \quad \theta_1 > 0 \ \ \& \ \ \theta_2 > 0.$$

- Let a solve $exp(z) = (z+1)/(z-1)$ and let u^* solve

$$exp(\theta_1 + \theta_2 u) = \frac{2 + (u+1)\theta_2}{-2 + (u+1)\theta_2}.$$

1.6 Locally D-optimal Designs for the Logistic Model on $X = [-1, 1]$ (Ford's PhD thesis, 1972)

$$log\ \frac{\pi(x)}{1-\pi(x)} = \theta_1 + \theta_2 x, \quad \theta^T = (\theta_1, \theta_2), \quad \theta_1 > 0 \ \ \& \ \ \theta_2 > 0.$$

- Let a solve $exp(z) = (z+1)/(z-1)$ and let u^* solve
$$exp(\theta_1 + \theta_2 u) = \frac{2 + (u+1)\theta_2}{-2 + (u+1)\theta_2}.$$
-

condition	locally D-optimal design
$\{\theta : \theta_2 - \theta_1 \geq a\}$	$\{\frac{a-\theta_1}{\theta_2}, \frac{-a-\theta_1}{\theta_2}; \frac{1}{2}, \frac{1}{2}\}$
$\{\theta : \theta_2 - \theta_1 < a, exp(\theta_1 + \theta_2) \leq \frac{\theta_2 + 1}{\theta_2 - 1}\}$	$\{-1, u^*; \frac{1}{2}, \frac{1}{2}\}$
$\{\theta : exp(\theta_1 + \theta_2 > \frac{\theta_2 + 1}{\theta_2 - 1}\}$	$\{-1, 1; \frac{1}{2}, \frac{1}{2}\}$

1.6 Locally D-optimal Designs for the Logistic Model on $X = [-1, 1]$ (Ford's PhD thesis, 1972)

$$log \frac{\pi(x)}{1-\pi(x)} = \theta_1 + \theta_2 x, \quad \theta^T = (\theta_1, \theta_2), \quad \theta_1 > 0 \ \ \& \ \ \theta_2 > 0.$$

- Let a solve $exp(z) = (z+1)/(z-1)$ and let u^* solve
$$exp(\theta_1 + \theta_2 u) = \frac{2 + (u+1)\theta_2}{-2 + (u+1)\theta_2}.$$
condition	locally D-optimal design
$\{\theta : \theta_2 - \theta_1 \geq a\}$	$\{\frac{a-\theta_1}{\theta_2}, \frac{-a-\theta_1}{\theta_2}; \frac{1}{2}, \frac{1}{2}\}$
$\{\theta : \theta_2 - \theta_1 < a, exp(\theta_1 + \theta_2) \leq \frac{\theta_2+1}{\theta_2-1}\}$	$\{-1, u^*; \frac{1}{2}, \frac{1}{2}\}$
$\{\theta : exp(\theta_1 + \theta_2 > \frac{\theta_2+1}{\theta_2-1}\}$	$\{-1, 1; \frac{1}{2}, \frac{1}{2}\}$
- Corrected results in Sebastiani and Settimi (JSPI, 1997)

1.7 By Brute Force and Guess Work

Consider the logistic model on a given design space X given by

$$log \frac{\pi(x)}{1 - \pi(x)} = \theta_0 + \theta_1 x,$$

where $\theta^T = (\theta_0, \theta_1) \in \Theta$ and Θ is known.

Design Criterion: Find $\xi^* = \arg\min_\xi \max_{\theta \in \Theta} log|M(\xi, \theta)|^{-1}$.

1.7 By Brute Force and Guess Work

Consider the logistic model on a given design space X given by

$$log \frac{\pi(x)}{1 - \pi(x)} = \theta_0 + \theta_1 x,$$

where $\theta^T = (\theta_0, \theta_1) \in \Theta$ and Θ is known.

Design Criterion: Find $\xi^* = \arg\min_\xi \max_{\theta \in \Theta} log|M(\xi, \theta)|^{-1}$.

- There is no known algorithm that is guaranteed to find a minimax optimal design.

1.7 By Brute Force and Guess Work

Consider the logistic model on a given design space X given by

$$log \frac{\pi(x)}{1-\pi(x)} = \theta_0 + \theta_1 x,$$

where $\theta^T = (\theta_0, \theta_1) \in \Theta$ and Θ is known.

Design Criterion: Find $\xi^* = \arg\min_\xi \max_{\theta\in\Theta} log|M(\xi,\theta)|^{-1}$.

- There is no known algorithm that is guaranteed to find a minimax optimal design.
- King & Wong (Biometrics, 2002) found minimax D-optimal designs when $\Theta = [0, 3.5] \times [1, 3.5]$ and X is unrestricted:

x_i	-0.35	0.62	1.39	2.11	2.88	3.85
w_i	0.18	0.21	0.11	0.11	0.21	0.18

1.8 Sensitivity Plot of the Generated Design

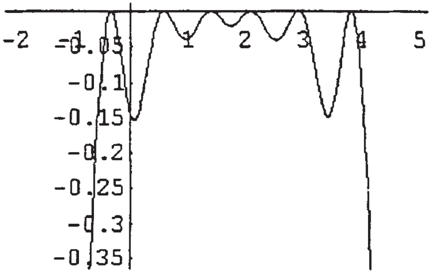


Figure 1. Plot of $\psi(x, \xi^*, \mu^*)$ for example 3.2 with $\Theta = [0, 3.5] \times [1, 3.5]$.

1.8 Sensitivity Plot of the Generated Design

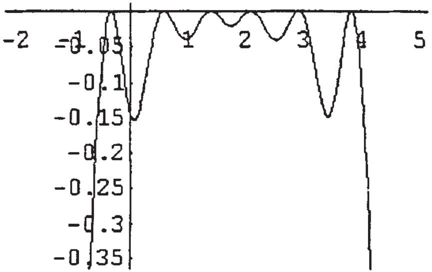


Figure 1. Plot of $\psi(x, \xi^*, \mu^*)$ for example 3.2 with $\Theta = [0, 3.5] \times [1, 3.5]$.

- References: Wong (Biometrika, 1992), Wong & Cook (JRSSB, 1993), Berger, King & Wong (Psychometrika, 1994)

1.9 Time required to discretize a 10-dimensional search space with different number of equally spaced points using a Mac laptop 2.6 GHz Intel Core i5

number of equally spaced points per covariate space	total number of grid points	CPU time required to generate the grid (secs)
2	$2^{10} = 1024$	0.0067
3	$3^{10} = 59049$	0.2302
4	$4^{10} = 1,048,576$	3.1136
5	$5^{10} = 9,765,625$	27.5529
6	$6^{10} = 60,466,176$	172.2832
7	$7^{10} = 282,475,249$	848.2922

1.9 Time required to discretize a 10-dimensional search space with different number of equally spaced points using a Mac laptop 2.6 GHz Intel Core i5

number of equally spaced points per covariate space	total number of grid points	CPU time required to generate the grid (secs)
2	$2^{10} = 1024$	0.0067
3	$3^{10} = 59049$	0.2302
4	$4^{10} = 1,048,576$	3.1136
5	$5^{10} = 9,765,625$	27.5529
6	$6^{10} = 60,466,176$	172.2832
7	$7^{10} = 282,475,249$	848.2922

- There are immediate implications....

1.10 Implications

- Current design algorithms such as Cocktail-based algorithms: Yu (Stat. & Comp., 2011) and Yang, et al., (JASA, 2014) may not work for high dimensional problems.

1.10 Implications

- Current design algorithms such as Cocktail-based algorithms: Yu (Stat. & Comp., 2011) and Yang, et al., (JASA, 2014) may not work for high dimensional problems.

- What about Bayesian optimal designs, other than independent uniform prior distribution on $[-1, 1]$? Even finding optimal designs for additive models on $[-10, 1]^k$ instead of $[-1, 1]^k$ becomes quickly problematic.

1.10 Implications

- Current design algorithms such as Cocktail-based algorithms: Yu (Stat. & Comp., 2011) and Yang, et al., (JASA, 2014) may not work for high dimensional problems.

- What about Bayesian optimal designs, other than independent uniform prior distribution on $[-1, 1]$? Even finding optimal designs for additive models on $[-10, 1]^k$ instead of $[-1, 1]^k$ becomes quickly problematic.

- Mathematical programming tools that require the search space be discretized, such as semi-definite programming (Papp, JASA, 2011, Duarte & Wong, Stat. & Comp., 2014, Duarte, Wong & Atkinson, J. of Multi. Ana., 2015 and Duarte, Wong & Dette, Stat. & Comp., 2017) may become inapplicable.

1.11 Mathematistry

In Praise of Simplicity not Mathematistry! Ten Simple Powerful Ideas for the Statistical Scientist

Roderick J. LITTLE

Ronald Fisher was by all accounts a first-rate mathematician, but he saw himself as a scientist, not a mathematician, and he railed against what George Box called (in his Fisher lecture) "mathematistry." Mathematics is the indispensable foundation of statistics, but for me the real excitement and value of our subject lies in its application to other disciplines. We should not view statistics as another branch of mathematics and favor mathematical complexity over clarifying, formulating, and solving real-world problems. Valuing simplicity, I describe 10 simple and powerful ideas that have influenced my thinking about statistics, in my areas of research interest: missing data, causal inference, survey sampling, and statistical modeling in general. The overarching theme is that statistics is a missing data problem and the goal is to predict unknowns with appropriate measures of uncertainty.

KEY WORDS: Calibrated Bayes; Causal inference; Measurement error; Missing data; Penalized spline of propensity.

1. INTRODUCTION: THE UNEASY RELATIONSHIP BETWEEN STATISTICS AND MATHEMATICS

American Statistical Association President, Sastry Pantula, recently proposed renaming the Division of Mathematical Sciences at the U.S. National Science Foundation as the Division of Mathematical and Statistical Sciences. Opponents, who viewed statistics as a branch of mathematics, questioned why statistics should be singled out for special treatment.

Data can be assembled in support of the argument that statistics *is* different—for example, the substantial number of academic departments of statistics and biostatistics, the rise of the statistics advanced placement examination, and the substantial number of undergraduate statistics majors. But the most important factor for me is that statistics is not just a branch of mathematics. It is an inductive method, defined by its applications to the sciences and other areas of human endeavor, where we try to glean information from data.

The relationship between mathematics and statistics is somewhat uneasy. Since the mathematics of statistics is often viewed as basically rather pedestrian, statistics is rather low on the totem pole of mathematical subdisciplines. Statistics needs its mathematical parent, since it is the indispensable underpinning of the subject. On the other hand, unruly statistics has ambitions to reach beyond the mathematics fold: it comes alive in applica- and medicine, and with increasing influence recently on the hard sciences such as astronomy, geology and physics.

The scientific theme of modern statistics fits the character of its most influential developer, the great geneticist, R. A. Fisher, who seemed to revolutionize the field of statistics in his spare time! Fisher's momentous move to Rothampsted Experimental Station rather than academia underlined his dedication to science. Though an excellent mathematician, Fisher viewed himself primarily as a scientist, and disparaged rivals like Neyman and Pearson as mere "mathematicians."

George Box's engaging Fisher lecture focused on the links between statistics and science (Box 1976). He wrote:

> My theme then will be first to show the part that [Fisher] being a good scientist played in his astonishing ingenuity, originality, inventiveness, and productivity as a statistician, and second to consider what message that has for us now.

Box attributed Fisher's hostility to mathematicians to distaste for what he called "mathematistry," which he defined as

> [...] the development of theory for theory's sake, which, since it seldom touches down with practice, has a tendency to redefine the problem rather than solve it. Typically, there has once been a statistical problem with scientific relevance but this has long since been lost sight of. (Box 1976)

Outline

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical
- does not need to discretize the search space

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical
- does not need to discretize the search space
- can solve high dimensional optimization problems

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical
- does not need to discretize the search space
- can solve high dimensional optimization problems
- optimal designs with various number of support points

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical
- does not need to discretize the search space
- can solve high dimensional optimization problems
- optimal designs with various number of support points
- can find singular designs all the same

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical
- does not need to discretize the search space
- can solve high dimensional optimization problems
- optimal designs with various number of support points
- can find singular designs all the same
- solely reliance on a mathematical approach can be limiting

2.1 Why Nature-Inspired Metaheuristic Algorithms?

- flexible, applicable to broad class of problems
- assumptions free
- easy to program and implement
- powerful and fast, and fun
- amazing, magical
- does not need to discretize the search space
- can solve high dimensional optimization problems
- optimal designs with various number of support points
- can find singular designs all the same
- solely reliance on a mathematical approach can be limiting
- can help find analytic solution or formula of optimal design for complicated problems

2.3 Usage of Nature-Inspired Metaheuristic Algorithms

- Recent trends indicate rapid growth of nature-inspired optimization in academia and industry. (Whitacre, 2011, Computing, Vol. 93, 121-133.)
- Survival of the flexible: explaining the recent dominance of nature-inspired optimization within a rapidly evolving world. (Whitacre, 2011, Computing, Vol. 93, 135-146.)

2.3 Usage of Nature-Inspired Metaheuristic Algorithms

- Recent trends indicate rapid growth of nature-inspired optimization in academia and industry. (Whitacre, 2011, Computing, Vol. 93, 121-133.)

- Survival of the flexible: explaining the recent dominance of nature-inspired optimization within a rapidly evolving world. (Whitacre, 2011, Computing, Vol. 93, 135-146.)

- Can lead in the new frontier of research: solve optimization problems with millions or billions of variables (Foreword by editors in a special issue in Information Sciences, 2015, Vol. 316, 437-439.)

2.4 Metaheuristic Algorithms

From Wikipedia, the free encyclopedia: Metaheuristic

In computer science, metaheuristic designates a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. Metaheuristics make few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. However, metaheuristics do not guarantee an optimal solution is ever found. Many metaheuristics implement some form of stochastic optimization.

2.4 Metaheuristic Algorithms

From Wikipedia, the free encyclopedia: Metaheuristic

In computer science, metaheuristic designates a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. Metaheuristics make few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. However, metaheuristics do not guarantee an optimal solution is ever found. Many metaheuristics implement some form of stochastic optimization.

- Our interest here is nature-inspired metaheuristic algorithms

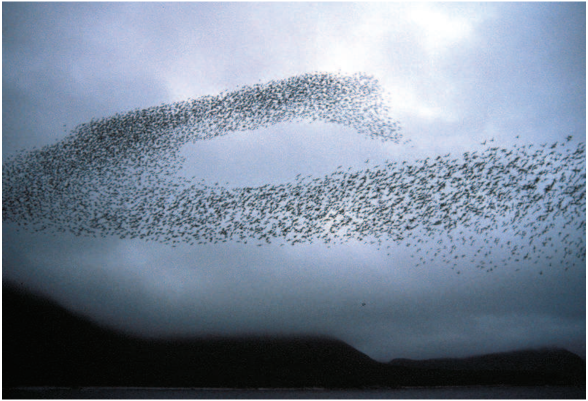
2.4 Metaheuristic Algorithms

From Wikipedia, the free encyclopedia: Metaheuristic

In computer science, metaheuristic designates a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. Metaheuristics make few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. However, metaheuristics do not guarantee an optimal solution is ever found. Many metaheuristics implement some form of stochastic optimization.

- Our interest here is nature-inspired metaheuristic algorithms
- Particle Swarm Optimization (PSO) method is based on animal instincts (Eberhard & Kennedy, IEEE, 1995)

2.5 PSO (Kennedy & Eberhard, 1995)



2.6 PSO: School of Fish

2.7 Basic Equations and Tuning Parameters in PSO

Two defining equations:

$$\mathbf{v}_{i+1} = \omega_i \mathbf{v}_i + c_1 \beta_1 (\mathbf{p}_i - \mathbf{x}_i) + c_2 \beta_2 (\mathbf{p}_g - \mathbf{x}_i),$$
$$\mathbf{x}_{i+1} = \mathbf{x}_i + \mathbf{v}_i.$$

x_i and v_i: position and velocity for the i^{th} particle
β_1 and β_2: random vectors
ω_i: inertia weight that modulates the influence of the last velocity
c_1: cognitive learning parameter
c_2: social learning parameter
p_i: Best position for the i^{th} particle (local optimal)
p_g : Best position for all particles (global optimal)

For many applications, $c_1 = c_2 = 2$ seem to work well and usually $20 - 50$ particles will suffice (Kennedy, IEEE, 1997).

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)
- Glowworm swarm optimization (2009)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)
- Glowworm swarm optimization (2009)
- Cuckoo search (Yang & Deb, 2009, Journal of Mathematical Modeling and Numerical Optimization)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)
- Glowworm swarm optimization (2009)
- Cuckoo search (Yang & Deb, 2009, Journal of Mathematical Modeling and Numerical Optimization)
- Firefly algorithm (2009, 2010)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)
- Glowworm swarm optimization (2009)
- Cuckoo search (Yang & Deb, 2009, Journal of Mathematical Modeling and Numerical Optimization)
- Firefly algorithm (2009, 2010)
- Bat algorithm (2010)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)
- Glowworm swarm optimization (2009)
- Cuckoo search (Yang & Deb, 2009, Journal of Mathematical Modeling and Numerical Optimization)
- Firefly algorithm (2009, 2010)
- Bat algorithm (2010)
- Grey Wolf algorithm (2014,2016)

2.8 Other Nature-Inspired Meta-Heuristic Algorithms

- Ant colony (1991)
- Differential Evolutionary (1997)
- Bees algorithm (2006)
- Artificial bee colony algorithm (2007)
- Saplings growing-up algorithm (2007)
- Monkey search (2007)
- Viral Systems (2008)
- Intelligent water drops algorithm (2009)
- Glowworm swarm optimization (2009)
- Cuckoo search (Yang & Deb, 2009, Journal of Mathematical Modeling and Numerical Optimization)
- Firefly algorithm (2009, 2010)
- Bat algorithm (2010)
- Grey Wolf algorithm (2014,2016)
- Bioinspired flower pollination algorithm (2015)

2.9 Cuckoo search (Yang & Deb, 2010)

Cuckoo search is a metaheuristic algorithm inspired by cuckoos' parasitic breeding behavior.

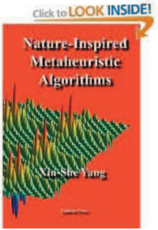
Figure 1: Balmer, 2009

2.10 Examples of Variants of Particle Swarm Optimization

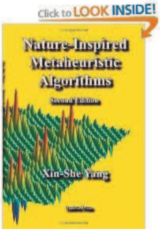
- Hierarchical PSO (Applications of Evol. Comput., 2004)
- Quantum PSO (Evolutionary Computation, 2004)
- Unified PSO (Advances in Natural Computation, 2005)
- Tournament PSO (IEEE Symposium Proceedings, 2007)
- Ladder PSO (Applied Soft Computing, 2009)
- Simplified PSO (Natural Computation, 2010)
- Strength Pareto PSO (Evolutionary Computation, 2010)
- Set-Based PSO (IEEE Transactions on Evol. Comp., 2010)
- Catfish PSO (Artificial Intelligence Research, 2012)
- Compact PSO (Information Sciences, 2013)
- Human Behavior-based PSO (Scientific World Journal, 2014)
- Selectively Informed PSO (Scientific Reports, 2014)
- Competitive Swarm Optimizer (Cybernetika, 2014)
- Fast PSO (Soft Computing, 2015)
- Galactic Swarm Optimization (Applied Soft Computing, 2016)

2.11 Resources for Metaheuristic Optimization and Nature-Inspired Metaheuristic Codes

Scholarpedia, the peer-reviewed open-access encyclopedia:
http://www.scholarpedia.org/article/Metaheuristic_Optimization
Xin-She Yang's 2008 book and updated in 2010:



2008



2010 (2nd edition)

2.12 Applications of PSO to Find Optimal Designs

- Qiu, J. H., Chen, R. B., Wang, W.C. & Wong, W. K. (2014). Using Animal Instincts to Design Efficient Biomedical Studies. Swarm and Evolutionary Computation Journal.
- Chen, R. B., Wang, W.C., Chang, & Wong, W. K. (2014). Minimax Optimal Designs via Particle Swarm Optimization Methods. Statistics & Computing.
- Wong, W. K., Wang, W.C., Chang, C. & Chen, R. B. (2015). Optimal Designs for Mixture Models using Particle Swarm Optimization Methods. PlosOne.
- Phoa, K. H. F., Chen, R. B., Wang, W. C. & Wong, W. K. (2015). Optimizing Two-level Supersaturated Designs by Particle Swarm. Technometrics.
- Kim, S. & Wong, W. K. (2017). Extended Two-stage Adaptive Designs for Phase II Clinical Trials. Statistical Methods in Medical Research.
- Chen, P. Y., Chen, R. B., Tung, H. C. and Wong, W. K. (2017). Standardized Maximim *D*-optimal Designs for Enzyme Kinetic Inhibition Models. Chemometrics and Intelligent Laboratory System.
- Lukemire, J., Mandal, A. and Wong, W. K. (2018). *d*-QPSO: A Quantum-Behaved Particle Swarm Technique for Finding *D*-Optimal Designs for Models with Mixed Factors and a Binary Response. Technometrics.

Outline

3.1 Application I: Standardized Maximin Designs

- Locally optimal designs can be sensitive to nominal values.

- The maximin approach assumes a known plausible region Θ for the model parameters θ. These maximin or minimax optimal designs maximize the minimal efficiency among all $\theta \in \Theta$, see King & Wong (Biometrics, 2000), and Biedermann and Dette (JASA, 2003).

- The standardized maximin D-optimal design ξ^*_{SM} maximizes

$$\Psi(\xi) = \min_{\theta \in \Theta} \left\{ \frac{|M(\xi, \theta)|}{\sup_\gamma |M(\gamma, \theta)|} \right\}^{1/p},$$

where $M(\gamma, \theta)$ is the $p \times p$ Fisher Information matrix for the nonlinear model with parameter θ from design γ.

3.2 Application I: Standardized Maximin Designs (cont'd)

- Chen, Chen & Wong (Chemometrics and Intelligent Laboratory System, 2018) applied PSO and found locally standardized maximin D-optimal designs for 4 common inhibit models used in enzyme kinetic studies.

- Contrary to common assumptions, not all locally *D*-optimal designs for these 3 or 4-parameter models with 2 factors are minimally supported.

- Using information of the PSO-generated designs, we were able to derive formulae of such optimal designs for the various inhibit models, including some with 3 nonlinear parameters.

3.3 Application II: Adaptive Designs

In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$

3.3 Application II: Adaptive Designs

In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$
- Determine 4 positive integers subject to type 1 and type 2 error constraints:
    - number of patients in Stage 1

3.3 Application II: Adaptive Designs

In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$
- Determine 4 positive integers subject to type 1 and type 2 error constraints:
    - number of patients in Stage 1
    - number of responders in Stage 1

3.3 Application II: Adaptive Designs

In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$
- Determine 4 positive integers subject to type 1 and type 2 error constraints:
  - number of patients in Stage 1
  - number of responders in Stage 1
  - number of (additional) patients in Stage 2

3.3 Application II: Adaptive Designs

In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$
- Determine 4 positive integers subject to type 1 and type 2 error constraints:
    - number of patients in Stage 1
    - number of responders in Stage 1
    - number of (additional) patients in Stage 2
    - number of (additional) responders in Stage 2

3.3 Application II: Adaptive Designs

In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$
- Determine 4 positive integers subject to type 1 and type 2 error constraints:
  - number of patients in Stage 1
  - number of responders in Stage 1
  - number of (additional) patients in Stage 2
  - number of (additional) responders in Stage 2
- Apply a greedy search to solve the discrete optimization problem relating Binomial probabilities and error rates

3.3 Application II: Adaptive Designs

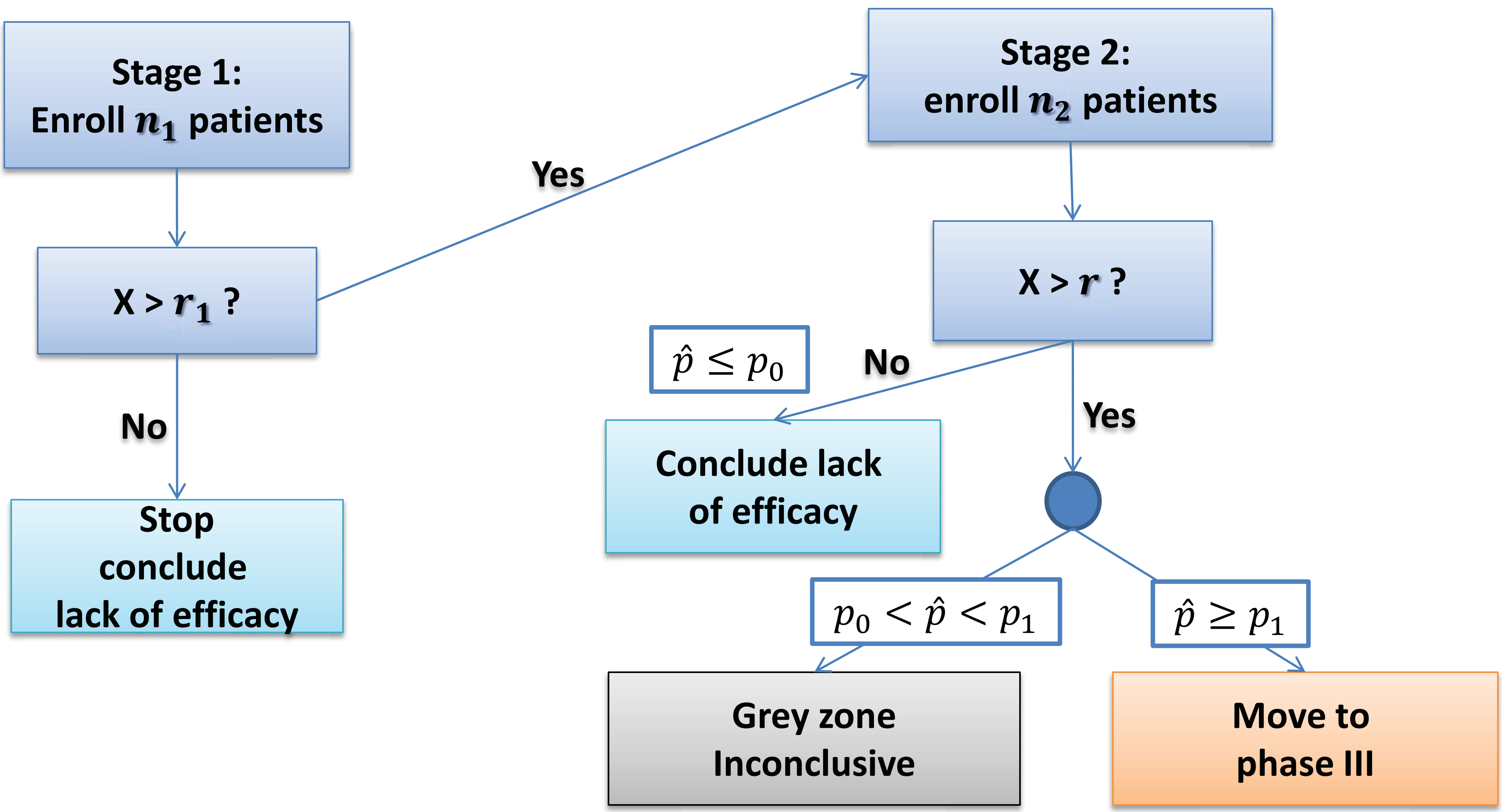
In Simon 2-Stage design for Phase II trials, user first selects two efficacy rates of interest p_0 and p_1 with $p_0 < p_1$.

- Set up hypothesis: $H_O : p \leq p_0$ versus $H_1 : p > p_1$
- Determine 4 positive integers subject to type 1 and type 2 error constraints:
  - number of patients in Stage 1
  - number of responders in Stage 1
  - number of (additional) patients in Stage 2
  - number of (additional) responders in Stage 2
- Apply a greedy search to solve the discrete optimization problem relating Binomial probabilities and error rates
- Lin & Shih (Biometrics, 2004) generalized the problem to 2 alternative hypotheses, and we extended it to 3 sets of alternative hypotheses.

3.4 Application II: A Discrete Optimization Problem

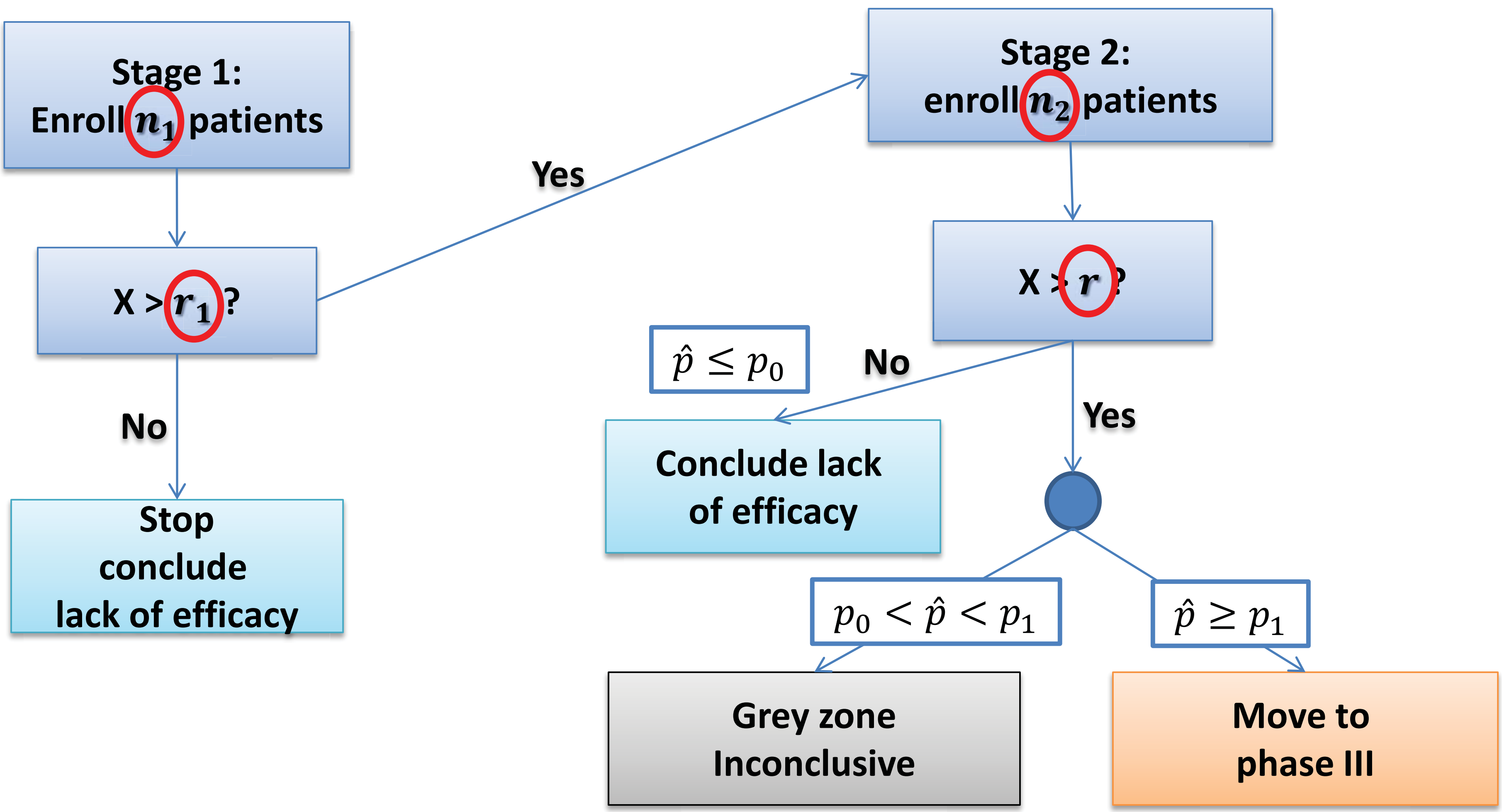
Simon's Two-Stage Designs

- X: the number of responders



Simon's Two-Stage Designs

- X: the number of responders



3.5 Test limits of PSO

Simon's 2-stage design has 4 parameters and the criterion was to minimize the expected sample size, or minimize the maximum sample size for the whole trial.

Goal: Extend Simon's 2 stage designs for 3 alternatives target efficacy rates

3.5 Test limits of PSO

Simon's 2-stage design has 4 parameters and the criterion was to minimize the expected sample size, or minimize the maximum sample size for the whole trial.

Goal: Extend Simon's 2 stage designs for 3 alternatives target efficacy rates

- Kim & Wong (SMMR, 2018) applied a modified version of PSO and searched over a constrained 10-dimensional space of positive integers and found optimal designs that a greedy algorithm cannot.

3.6 A 10 Integer-valued Parameters Problem to Optimize

Problem is to optimize $\theta^T = (s_1, r_1, q_1, n_1, s, l, r, m, q, n)$ given error for testing each of the three possible alternative hypotheses rates and the criterion is one of minimizing the maximum (or expected) sample sizes.

The parameters l, m, n are the total number of patients required for the entire trial corresponding to the alternative hypotheses, H_{11}: $p > p_1$, H_{12}: $p > p_2$, and H_{13}: $p > p_3$, respectively.

If true response probability is p, similar argument in Simon's original paper shows the probability of failing to reject H_0 is

$$G(\theta|p) = B(s_1, n_1, p) + \sum_{x=s_1+1}^{\min(r_1,s)} b(x, n_1, p)B(s-x, l_2, p) + \sum_{x=r_1+1}^{\min(q_1,r)} b(x, n_1, p)B(r-x, m_2, p)$$
$$+ \sum_{x=q_1+1}^{\min(q,n_1)} b(x, n_1, p)B(q-x, n_2, p),$$

3.7 Application 3:Optimal Designs for GLMs with Mixed Factors

Table 1: The left panel is the theoretical design from Yang, Zhang & Huang (Statistica Sinica, 2011) assuming one continuous factor has a unbounded range. The right panel is the D-optimal design from PSO with a large boundary $[-10, 10]$.

X_1	X_2	X_3	p_i
-2	-1	-0.456	0.125
-2	-1	-2.544	0.125
-2	1	-1.456	0.125
-2	1	-3.544	0.125
2	-1	1.544	0.125
2	-1	-0.544	0.125
2	1	0.544	0.125
2	1	-1.544	0.125

X_1	X_2	X_3	p_i
-2	-1	-2.544	0.25
-2	1	-1.457	0.25
2	-1	1.544	0.25
2	1	-1.544	0.25

3.8 The PSO-generated design when the continuous factor has a small range $[-2, 2]$ (Lukemire, Mandal & Wong, Technometrics, 2018).

3.8 The PSO-generated design when the continuous factor has a small range $[-2, 2]$ (Lukemire, Mandal & Wong, Technometrics, 2018).

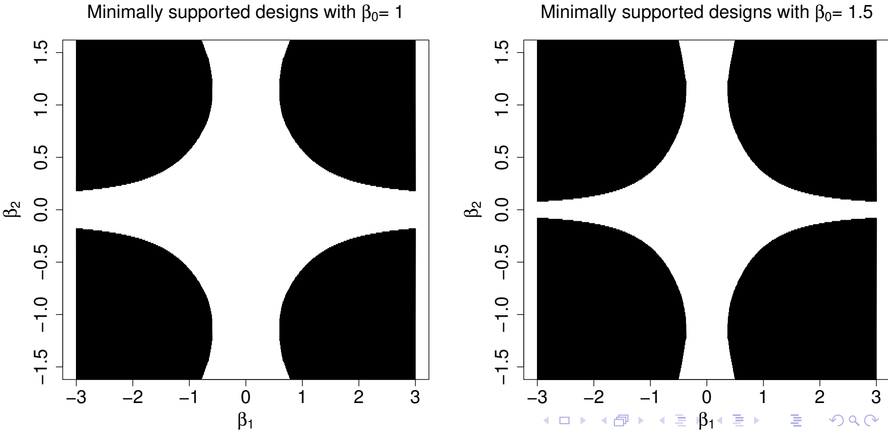
- When theory is not available, PSO methodology can help. When we restrict the final continuous factor to its natural setting PSO finds the following design.

3.8 The PSO-generated design when the continuous factor has a small range $[-2, 2]$ (Lukemire, Mandal & Wong, Technometrics, 2018).

- When theory is not available, PSO methodology can help. When we restrict the final continuous factor to its natural setting PSO finds the following design.
-

X_1	X_2	X_3	w_i
-2	-1	-2.000	0.212
-2	1	-2.000	0.043
-2	1	-1.649	0.166
2	-1	1.745	0.214
2	-1	-0.748	0.075
2	1	-1.748	0.214
2	1	0.748	0.075

3.9 Minimally supported designs for logistic model with 2 additive factors and an intercept term



3.10 Application 4: Multiple-objective Optimal Designs

- Experiments may have multiple objectives of varying importance. For example, extrapolate and estimate parameters at the same time or estimate parameters but there is model uncertainty (Dette, Melas & Wong, JASA, 2001)

3.10 Application 4: Multiple-objective Optimal Designs

- Experiments may have multiple objectives of varying importance. For example, extrapolate and estimate parameters at the same time or estimate parameters but there is model uncertainty (Dette, Melas & Wong, JASA, 2001)

- Want to find a design that delivers user-specified efficiencies under the various objectives with more important objectives having higher efficiencies requirements

3.10 Application 4: Multiple-objective Optimal Designs

- Experiments may have multiple objectives of varying importance. For example, extrapolate and estimate parameters at the same time or estimate parameters but there is model uncertainty (Dette, Melas & Wong, JASA, 2001)

- Want to find a design that delivers user-specified efficiencies under the various objectives with more important objectives having higher efficiencies requirements

- Cook & Wong (JASA, 1994) proposed a graphical method of constructing a dual-objective optimal design for linear regression problems; Clyde & Chaloner (JASA, 1996) extended the method to several objectives for nonlinear models

3.11 Dual Objective Optimal Designs

- Constrained Optimal Designs
  i.e. design that satisfies a set of user-specified efficiency requirements; eg. minimize $\phi_2(\xi)$ subject to $\phi_1(\xi) \leq c$.

3.11 Dual Objective Optimal Designs

- Constrained Optimal Designs
  i.e. design that satisfies a set of user-specified efficiency requirements; eg. minimize $\phi_2(\xi)$ subject to $\phi_1(\xi) \leq c$.

- Compound Optimal Designs
  i.e. design that minimizes a fixed convex combination of convex functionals: $\phi(\xi|\lambda) = \lambda\phi_1(\xi) + (1-\lambda)\phi_2(\xi)$.

3.11 Dual Objective Optimal Designs

- Constrained Optimal Designs
  i.e. design that satisfies a set of user-specified efficiency requirements; eg. minimize $\phi_2(\xi)$ subject to $\phi_1(\xi) \leq c$.

- Compound Optimal Designs
  i.e. design that minimizes a fixed convex combination of convex functionals: $\phi(\xi|\lambda) = \lambda\phi_1(\xi) + (1-\lambda)\phi_2(\xi)$.

- Compound Optimal Designs are equivalent to Constrained Optimal Designs: Plot efficiencies of each compound optimal versus λ, $\lambda \in [0, 1]$.

3.11 Dual Objective Optimal Designs

- Constrained Optimal Designs
  i.e. design that satisfies a set of user-specified efficiency requirements; eg. minimize $\phi_2(\xi)$ subject to $\phi_1(\xi) \leq c$.

- Compound Optimal Designs
  i.e. design that minimizes a fixed convex combination of convex functionals: $\phi(\xi|\lambda) = \lambda\phi_1(\xi) + (1-\lambda)\phi_2(\xi)$.

- Compound Optimal Designs are equivalent to Constrained Optimal Designs: Plot efficiencies of each compound optimal versus λ, $\lambda \in [0, 1]$.

- Prioritize the importance of the objectives and apply theory for single-objective study (Cook & Wong, JASA, 1994).

3.12 Efficiency Plots

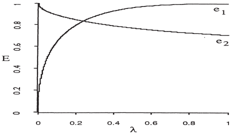


Figure 1. Efficiencies E Versus λ for Example 1.

Further, using (9), we obtain the values given in the first two lines of table 1 in Studden (1982).

To connect explicitly the constrained design problems and the compound design problems, the relationship between a constraint expressed as $E_1(\xi) \geq e_1$ (or $E_2(\xi) \geq e_2$) and the corresponding value of λ must be established. Using results from Fedorov (1980, thm. 1), it can be shown that ξ_λ maximizes $\phi(\xi \mid \lambda)$ if and only if

$$2(1-\lambda)d_1(x, \xi_\lambda) + \lambda\{b^T M_2^{-1}(\xi_\lambda) f_2(x)\}^2 - 4(1-\lambda) - \lambda b^T M_2^{-1}(\xi_\lambda) b \leq 0 \quad (11)$$

for all x in $[-1, 1]$. Further (11) becomes an equality at the support points for ξ_λ. In that case, substituting (9) into (11) yields

$$\lambda = \frac{e_1^2}{4 + 4(1-e_1)^{1/2} - 4e_1 + e_1^2}. \quad (12)$$

Figure 1, constructed using (10) and (12), shows the relationship between optimal designs with efficiency constraints and compound optimal designs. For example, a design that maximizes ϕ_2 subject to the constraint $E_1(\xi) \geq .6$ can be found by maximizing $\phi(\xi \mid \lambda)$ with $\lambda \approx .1$. Figure 1 contains useful information on the interpretation of λ as well. In particular, it might be felt that setting $\lambda = .5$ would yield a design in which equal interest is placed on the two criteria. But from Figure 1, the compound design problem with $\lambda = .5$ is equivalent to the constrained problem in which we maximize ϕ_2 subject to the constraint that $E_1(\xi) \geq .96$. The resulting constrained design has $E_2(\xi_{\lambda=.5}) \approx .78$. In terms of the efficiencies, placing equal interest on the two criteria would seem to require $\lambda = .25$, because at that point $E_1(\xi_\lambda) \approx E_2(\xi_\lambda) \approx .84$. Finally, reconstructing the plot in Figure 1 so that the horizontal axis is $1 - \lambda$ rather than λ provides the corresponding plot for maximizing ϕ_1 subject to a constraint on ϕ_2.

Example 2. For the simple linear regression model $f_1^T(x) = (1, x)$ on $\chi = [-1, 1]$, consider balancing A optimality with precise estimation of the response at the point $z = .75$:

$$\phi_1(\xi) = -d_1(z, \xi)/d_1(z, \xi^1) = -[E_1(\xi)]^{-1}$$

and

$$\phi_2(\xi) = -\text{tr } M_1^{-1}(\xi)/\text{tr } M_1^{-1}(\xi^2) = -[E_2(\xi)]^{-1}.$$

The design ξ^1 is optimal for ϕ_1 and has the minimum variance possible for a fitted value at the point z. This minimum variance is the same as that obtained under the design that places mass 1 at z. The design ξ^2 is the A-optimal design. Both ξ^1 and ξ^2 are supported at ± 1, with the masses at 1 being $\frac{7}{8}$ and $\frac{1}{2}$. Thus $d_1(z, \xi^1) = 1$ and $\text{tr}[M_1^{-1}(\xi^2)] = 2$.

Define $w(x) = (9x^2 - 25x + 16)^{1/2}$ and $g(x) = (3x + 4 - w(x))/8$ for $0 \leq x \leq 1$. Then

$$\xi_\lambda(1) = g(E_1(\xi_\lambda)), \quad (13)$$

which is the analog of (9) for this example. Next, again from Fedorov (1980, thm. 1) ξ_λ satisfies

$$\lambda(f_1^T(x) M_1^{-1}(\xi_\lambda) f_1(z))^2 + (1-\lambda) f_1(x) M_1^{-2}(\xi_\lambda) f_1(x) = \lambda d_1(z, \xi_\lambda) + (1-\lambda)\text{tr } M_1^{-1}(\xi_\lambda)$$

at the support points $x = \pm 1$. Substituting (13) into this equation yields

$$\lambda = \frac{8(w(e_1) - 3e_1)}{34g(e_1) - 17 - 48g^2(e_1)},$$

which is the analog of (12) for $e_1 \geq .64$. From this we constructed Figure 2. The relationships in Figure 2 are, of course, qualitatively similar to those in Figure 1. But note that the value of λ at which the efficiencies are equal is much larger than that for Figure 1. Generally, the interpretation of λ depends heavily on the functionals involved. Useful inter-

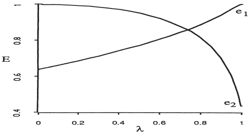


Figure 2. Efficiencies E Versus λ for Example 2.

3.13 A Quick Demonstration using the Hill's model

- Additional references include: (Stigler, JASA, 1971, Studden, JASA, 1980, Cook & Wong, JASA, 1992, Clyde & Chaloner, JASA, 1994, Huang & Wong, Biometrics, 1998, Zhu & Wong, J. Biopharm. Stat., 2000, Stat. in Med., 2001, Imhof & Wong , Biometrics, 2000).

3.13 A Quick Demonstration using the Hill's model

- Additional references include: (Stigler, JASA, 1971, Studden, JASA, 1980, Cook & Wong, JASA, 1992, Clyde & Chaloner, JASA, 1994, Huang & Wong, Biometrics, 1998, Zhu & Wong, J. Biopharm. Stat., 2000, Stat. in Med., 2001, Imhof & Wong , Biometrics, 2000).

- We now present a demo to find single and multiple-objective locally optimal approximate designs using different nature-inspired metaheuristic algorithms. The 3 objectives of interest are to estimate the ED50, minimum effective dose (MED) or parameters in a 4-parameter logistic model.

3.13 A Quick Demonstration using the Hill's model

- Additional references include: (Stigler, JASA, 1971, Studden, JASA, 1980, Cook & Wong, JASA, 1992, Clyde & Chaloner, JASA, 1994, Huang & Wong, Biometrics, 1998, Zhu & Wong, J. Biopharm. Stat., 2000, Stat. in Med., 2001, Imhof & Wong , Biometrics, 2000).

- We now present a demo to find single and multiple-objective locally optimal approximate designs using different nature-inspired metaheuristic algorithms. The 3 objectives of interest are to estimate the ED50, minimum effective dose (MED) or parameters in a 4-parameter logistic model.

- Bayesian optimal approximate designs can likewise be constructed and verified using an equivalence theorem.

3.11 Mean function of the Hill model

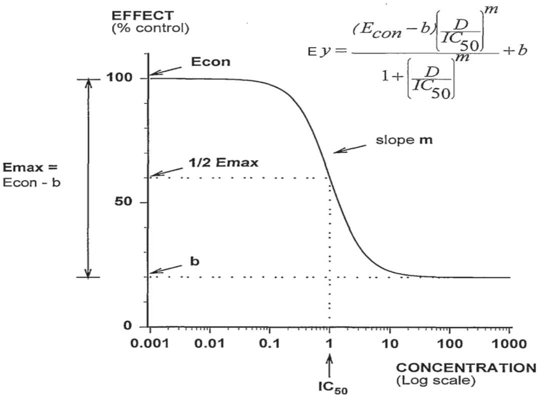


Figure 1. Graph of the 4-parameter Hill model. The following parameter values have been assumed: $E_{\mathrm{con}} = 100$, $b = 20$, $IC_{50} = 1$, and $m = -1.5$.

3.12 Three-Objective Optimal Designs

Assume nominal values, dose interval and the minimum effect sought δ are given for the Hill's model. For a user-selected vector $\lambda = (\lambda_1, \lambda_2, \lambda_3)$ with $\lambda_i \geq 0, i = 1, 2, 3$ and $\lambda_1 + \lambda_2 + \lambda_3 = 1$, the sought multiple-objective optimal design is the approximate design that maximizes

$$\lambda_1 log(Eff_D(\xi)) + \lambda_2 log(Eff_{ED_{50}}(\xi)) + \lambda_3 log(Eff_{MED}(\xi))$$

$$= \lambda_1 0.25 log(|M(\xi, \boldsymbol{\Theta})|) - \lambda_2 log(Var(\widehat{ED}_{50})) - \lambda_3 log(Var(\widehat{MED})).$$

Here $ED50$ and MED are the median effective dose and the user-specified minimum effective dose.

3.12 Three-Objective Optimal Designs

Assume nominal values, dose interval and the minimum effect sought δ are given for the Hill's model. For a user-selected vector $\lambda = (\lambda_1, \lambda_2, \lambda_3)$ with $\lambda_i \geq 0, i = 1, 2, 3$ and $\lambda_1 + \lambda_2 + \lambda_3 = 1$, the sought multiple-objective optimal design is the approximate design that maximizes

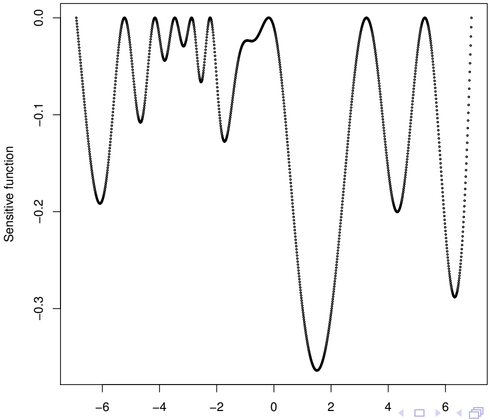
$$\lambda_1 log(Eff_D(\xi)) + \lambda_2 log(Eff_{ED_{50}}(\xi)) + \lambda_3 log(Eff_{MED}(\xi))$$

$$= \lambda_1 0.25 log(|M(\xi, \mathbf{\Theta})|) - \lambda_2 log(Var(\widehat{ED}_{50})) - \lambda_3 log(Var(\widehat{MED})).$$

Here $ED50$ and MED are the median effective dose and the user-specified minimum effective dose.

Reference: Hyun, W. & Wong, W. K. (2015). Multiple-objective optimal designs for studying the dose response function and interesting dose levels. International Journal of Biostatistics.

3.13 Sensitivity Plot of a Robust Bayesian Multiple Objective Optimal Design with uniform prior distributions



3.14 Current Work

3.14 Current Work

- Given a fixed time interval, a fixed number of observations, a statistical model and an optimality criterion, design questions for a longitudinal study are

3.14 Current Work

- Given a fixed time interval, a fixed number of observations, a statistical model and an optimality criterion, design questions for a longitudinal study are
    - how many time points is optimal?

3.14 Current Work

- Given a fixed time interval, a fixed number of observations, a statistical model and an optimality criterion, design questions for a longitudinal study are
  - how many time points is optimal?
  - what are the sampling time points to observe the correlated responses?

3.14 Current Work

- Given a fixed time interval, a fixed number of observations, a statistical model and an optimality criterion, design questions for a longitudinal study are
  - how many time points is optimal?
  - what are the sampling time points to observe the correlated responses?
  - do I need replicates and if so how to distribute the replicates?

3.14 Current Work

- Given a fixed time interval, a fixed number of observations, a statistical model and an optimality criterion, design questions for a longitudinal study are
  - how many time points is optimal?
  - what are the sampling time points to observe the correlated responses?
  - do I need replicates and if so how to distribute the replicates?

- Convergence Issues of PSO

3.14 Current Work

- Given a fixed time interval, a fixed number of observations, a statistical model and an optimality criterion, design questions for a longitudinal study are
  - how many time points is optimal?
  - what are the sampling time points to observe the correlated responses?
  - do I need replicates and if so how to distribute the replicates?
- Convergence Issues of PSO
- Finding optimal designs for nonlinear models with many factors and interaction terms (high dimension models)

3.15 A locally *D*-optimal design found by Twice Competitive Swarm Optimizer for a five-factor Poisson model with all pairwise interaction terms (Zhang and Wong, IEEE, 2018, under review)

x_1	x_2	x_3	x_4	x_5	w_i
1.000	1.000	0.685	1.000	-0.730	0.033
1.000	1.000	0.430	-1.000	-1.000	0.062
1.000	-1.000	-1.000	1.000	1.000	0.010
1.000	0.011	1.000	1.000	-1.000	0.062
1.000	1.000	1.000	-0.460	-1.000	0.063
1.000	1.000	0.677	1.000	-1.000	0.057
0.404	1.000	0.670	1.000	-1.000	0.058
1.000	-0.470	1.000	1.000	-0.581	0.061
0.406	1.000	1.000	-0.454	-1.000	0.063
1.000	-0.479	0.508	1.000	-1.000	0.062
-1.000	-1.000	-1.000	1.000	1.000	0.048
1.000	1.000	1.000	1.000	-0.724	0.056
1.000	0.405	1.000	0.127	-1.000	0.062
0.390	-0.012	1.000	1.000	-1.000	0.062
1.000	1.000	1.000	-0.601	-0.691	0.063
1.000	1.000	1.000	1.000	-1.000	0.061
0.337	1.000	1.000	1.000	-0.686	0.057
0.411	1.000	1.000	1.000	-1.000	0.059

With small weights, a large sample is required to implement the design.

3.16 Optimal Discrimination Designs for 2 or 3 multi-factor polynomial models without a null model assumption (Yue, Vanderburgh & Wong, under review)

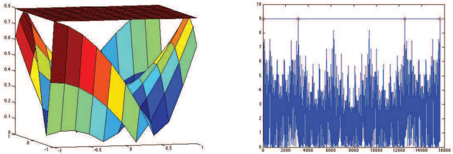


Figure 1: Plots of the sensitivity functions of two designs found by our algorithm for Example 3 (left) and Example 5 (right) to confirm their optimality.

Outline

4.1 Closing Thoughts

- Remember the "No Free Lunch Theorem"

4.1 Closing Thoughts

- Remember the "No Free Lunch Theorem"
- How to find efficient algorithms for a class of problems of interest?

4.1 Closing Thoughts

- Remember the "No Free Lunch Theorem"
- How to find efficient algorithms for a class of problems of interest?
- PSO is a general optimization tool, not limited to minimizing convex functionals or for finding efficient designs

4.1 Closing Thoughts

- Remember the "No Free Lunch Theorem"
- How to find efficient algorithms for a class of problems of interest?
- PSO is a general optimization tool, not limited to minimizing convex functionals or for finding efficient designs
- Find minimum bias designs, minimum mean-square error (MSE) designs (Stokes, Mandal & Wong, 2017, under prep.)

4.1 Closing Thoughts

- Remember the "No Free Lunch Theorem"
- How to find efficient algorithms for a class of problems of interest?
- PSO is a general optimization tool, not limited to minimizing convex functionals or for finding efficient designs
- Find minimum bias designs, minimum mean-square error (MSE) designs (Stokes, Mandal & Wong, 2017, under prep.)
- Identify parameter redundancy in mixture distributions (Park & Wong, 2017, under prep.)

4.1 Closing Thoughts

- Remember the "No Free Lunch Theorem"
- How to find efficient algorithms for a class of problems of interest?
- PSO is a general optimization tool, not limited to minimizing convex functionals or for finding efficient designs
- Find minimum bias designs, minimum mean-square error (MSE) designs (Stokes, Mandal & Wong, 2017, under prep.)
- Identify parameter redundancy in mixture distributions (Park & Wong, 2017, under prep.)
- Can hybridize with mathematical programming tools and traditional methods (such as simplex methods,Interior Point, etc) and speed up the search for the optimum (Garcia-Rodenas, Fidalgo-Lopez & Wong, 2018, under review)

4.2 PSO for Solving a System of Nonlinear Equations

Let $x^T = (x_1, x_2, .., x_n)$. We want to solve

$$f_1(x_1, x_2, .., x_n) = 0,$$
$$f_2(x_1, x_2, .., x_n) = 0,$$
$$\cdot$$
$$\cdot$$

and

$$f_r(x_1, x_2, .., x_n) = 0.$$

4.2 PSO for Solving a System of Nonlinear Equations

Let $x^T = (x_1, x_2, .., x_n)$. We want to solve

$$f_1(x_1, x_2, .., x_n) = 0,$$
$$f_2(x_1, x_2, .., x_n) = 0,$$
$$.$$
$$.$$

and

$$f_r(x_1, x_2, .., x_n) = 0.$$

- If x^* is the global minimum of $F(x) = \sum_{i=1}^{r} f_i{}^2(x)$, then x^* solves the above system of equations. (Jabeipour, et al., 2011, Computers and Mathematics with Applications)

4.2 PSO for Solving a System of Nonlinear Equations

Let $x^T = (x_1, x_2, .., x_n)$. We want to solve

$$f_1(x_1, x_2, .., x_n) = 0,$$
$$f_2(x_1, x_2, .., x_n) = 0,$$
$$.$$
$$.$$

and

$$f_r(x_1, x_2, .., x_n) = 0.$$

- If x^* is the global minimum of $F(x) = \sum_{i=1}^{r} f_i{}^2(x)$, then x^* solves the above system of equations. (Jabeipour, et al., 2011, Computers and Mathematics with Applications)
- Alternatively, assume $r = n$ and define $F(x) = \sum_{i=1}^{r} |f_i(x)|$ and find its global minimum. (Wang, et al., 2009, IEEE Xplore)

4.6 Conclusions

- There are many more meta-heuristic algorithms, for example, Imperialist Competitive Algorithm (Masoudi, Holling & Wong, CSDA, 2016)

4.6 Conclusions

- There are many more meta-heuristic algorithms, for example, Imperialist Competitive Algorithm (Masoudi, Holling & Wong, CSDA, 2016)

- Incorporate optimization classes from Engineering and Computer Science into the statistics training curriculum

4.6 Conclusions

- There are many more meta-heuristic algorithms, for example, Imperialist Competitive Algorithm (Masoudi, Holling & Wong, CSDA, 2016)
- Incorporate optimization classes from Engineering and Computer Science into the statistics training curriculum
- More?

4.6 Conclusions

- There are many more meta-heuristic algorithms, for example, Imperialist Competitive Algorithm (Masoudi, Holling & Wong, CSDA, 2016)

- Incorporate optimization classes from Engineering and Computer Science into the statistics training curriculum

- More?

  - Workshop on Particle Swarm Optimization and Evolutionary Computation (20 - 21 Feb 2018) at IMS, NUS (in this room).

    http://ims.nus.edu.sg/events/2018/wpso/index.php

Acknowledgements

The research of Wong was partially supported by the National Institute of General Medical Sciences of the National Institutes of Health under Award Number R01GM107639

Co-authors:

Anthony Atkinson (The London School of Economics and Political Science, UK)
Ray-Bing Chen, Ping Yang Chen (Dept. of Statistics, National Cheng Kung University, Taiwan)
Holger Dette (Ruhr University, Germany)
Belmiro Duarte (Dept. of Chemical Engineering, University of Coimbra, Portugal)
Ricardo Garcia-Ródenas(Dept. of Mathematics, UCLM, Spain)
Won Hyun (Johnson & Johnson, Irvine, California)
Jozef Kiselak (Institute of Mathematics, Safarik University, Slovakia)
Jesús López-Fidalgo (Institute for Culture and Society, University of Navarra, Spain)
Joshua Lukemire (Dept. of Biostatistics, Emory University)
Abhyudal Mandal (Dept. of Statistics, U of Georgia)
Ehsan Masoudi, Heinz Holling (Dept. of Psychology, Münster University, Germany)
Guanghao Qi (Dept. of Biostatistics, Johns Hopkins University, USA)
Yu Shi, Zizhao Zhao (Dept. of Biostatistics, UCLA)
Milan Stehlik (Dept. of Applied Statistics, Johannes Kepler Univeristy, Linz, Austria)
Lieven Vanderburghe (Dept. of Electrical Engineering, UCLA)
Weichung Wang (Dept. of Mathematics, National Taiwan University)
Yuguang Yue (Dept. of Mathematics, Austin, University of Texas)