d-QPSO : Finding optimal designs for models with many continuous and discrete factors and a binary response

Joshua Lukemire, Abhyuday Mandal, and Weng Kee Wong

UNIVERSITY OF GEORGIA

Introduction

- A motivating example is the odor removal study conducted by textile engineers at the UGA.
- Scientists are interested in removing odor causing volatiles from bio-plastics.

Factor types and levels for the bio-plastics odor removal experiment

Type	Factor	Levels −	Levels +
Discrete	Algae	Catfish algae	Solix Microalgae
	Scavenger	Activated Carbon	Zeolite
	Resin	Polyethylene	Polypropylene
	Compatibilizer	Absent	Present
Continuous	Temperature	Temperature from 5°C to 35°C	

- The response Y is binary, denoting whether the odor is successfully removed from the bio-plastic.
- We model μ, the mean response of Y, as

$$logit(\mu) = \beta_0 + \beta_1 Algae + \beta_2 Scavenger + \beta_3 Resin + \beta_4 Compatibilizer + \beta_5 Temp$$

but we also allow generalizations of this model to various link functions, more mixed factors, including interaction terms.

Optimal Designs for Generalized Linear Models

- Unlike linear models, the information matrix for GLMs depend on the unknown parameters. This makes the problem of identifying optimal designs for GLMs challenging, especially when there are both continuous and discrete (i.e. mixed) factors and there number of factors is large.
- Our goal is to construct locally optimal designs (Chernoff, 1953) and Bayesian optimal designs (Chaloner & Verninelli, 1995) for mixed models with many factors. Bayesian optimal designs incorporate a prior distribution on the unknown parameters, whereas local optimal design permits only a degenerate distribution.

Quantum Particle Swarm Optimization

- d-QPSO is a variant of Quantum-Behaved Particle Swarm Optimization (QPSO) (Sun et. al, 2004). Unlike PSO, in QPSO the particles do not have velocities.
- In QPSO a collection of particles, known as a swarm, searches for the optimal solution to the problem of interest.
- The worth of a particle is measured by its fitness: the value of the objective function at its current position.
- Each member of the swarm has its own idea of where the best solution is based on the solutions that particle has seen. This position is known as the personal best or pbest.
- Additionally, each member of the swarm knows where the overall best solution any particle has seen is, a position known as the global best or gbest.
- At each iteration the particle positions are updated via random draws, with positions that are near to known good solutions (high pbest, gbest) being more likely to be drawn (Sun et. al, 2006). This applies to both discrete and continuous factor positions, allowing us to optimize both simultaneously.
- For design problems, each particle corresponds to a candidate experimental design, and a particle's fitness is the log determinant of the Fisher information matrix for that experimental design.
- By searching the settings for the discrete and continuous factors, d-QPSO can find designs with an arbitrary number of support points.
- We allow multiple swarms to search at once, which can randomly share their knowledge about the best design with each other via an elitist breeding mutator.

$$\xi_i = \begin{array}{|ccc|} \hline X_1 & X_2 & P \\ \hline x_{i,11} & x_{i,12} & p_{i,1} \\ x_{i,21} & x_{i,22} & p_{i,2} \\ x_{i,31} & x_{i,32} & p_{i,3} \\ x_{i,41} & x_{i,42} & p_{i,4} \\ \hline \end{array} \xrightarrow{vectorize} \begin{bmatrix} x_{i,11} \\ x_{i,12} \\ p_{i,1} \\ x_{i,21} \\ \vdots \\ p_{i,4} \end{bmatrix} = \begin{bmatrix} \psi_{i,1} \\ \psi_{i,2} \\ \psi_{i,3} \\ \psi_{i,4} \\ \vdots \\ \psi_{i,4(2+1)=12} \end{bmatrix} = \boldsymbol{\psi}_i.$$

Figure 1: Illustration of converting a design ξ_i to a particle $\boldsymbol{\psi}_i$.

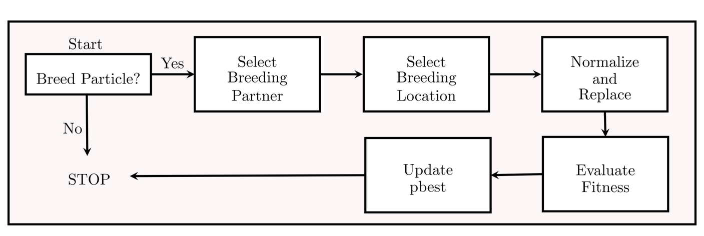

Figure 2: Steps in the Elitist Breeding mutator. At the swarm level the possible breeding particles are the particles within the same swarm, whereas at the habitat level the possible breeding partners are the other particles within the same swarm or from another randomly selected swarm.

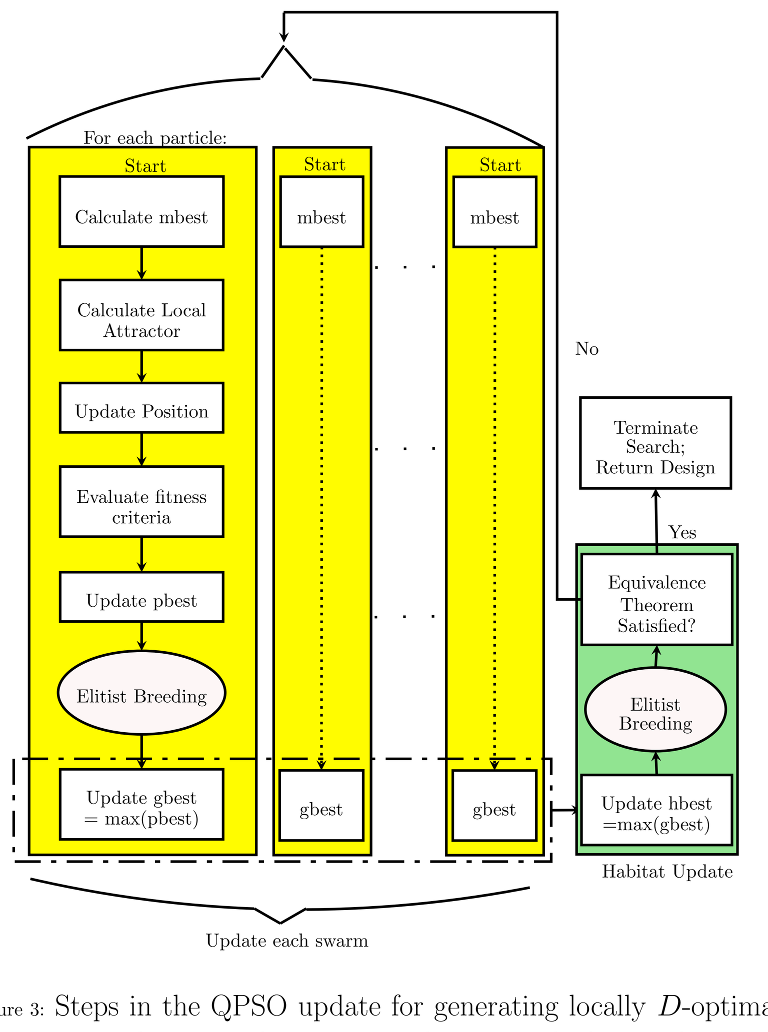

Figure 3: Steps in the QPSO update for generating locally D-optimal approximate designs. The swarm update is applied to each swarm individually, and the habitat update is performed on all swarms.

Locally D-optimal Approximate Designs

d-QPSO algorithm-generated locally D-optimal design for the odor removal experiment − Nominal value:s $\boldsymbol{\beta} = (-1, 2, 0.5, -1, -0.25, 0.13)^T$

Support point	Alg.	Sca.	Res.	Com.	Temp.	$p_i(\%)$
1	−1	−1	−1	−1	9.040	3.70
2	−1	−1	−1	−1	25.788	4.30
3	−1	−1	−1	1	29.710	10.17
4	−1	−1	1	−1	35.000	4.73
5	−1	−1	1	1	29.579	11.59
6	−1	1	−1	−1	5.000	9.80
7	−1	1	−1	1	5.206	7.86

Support point	Alg.	Sca.	Res.	Com.	Temp.	$p_i(\%)$
8	−1	1	1	−1	16.894	2.20
9	−1	1	1	−1	33.366	8.80
10	−1	1	1	1	35.000	6.10
11	1	−1	−1	1	5.000	5.11
12	1	−1	1	−1	5.000	10.75
13	1	−1	1	1	5.000	5.23
14	1	1	1	1	5.000	9.71

Bayesian D-optimal Designs

Bayesian design for the odor removal experiment using independent uniform priors for all parameters. We took $\beta_0 \sim U(-2, 0)$, $\beta_1 \sim U(0, 4)$, $\beta_2 \sim U(0, 1)$, $\beta_3 \sim U(-2, 0)$, $\beta_4 \sim (-.5, 0)$, and $\beta_5 \sim U(0, 0.26)$.

Support point	Alg.	Sca.	Res.	Com.	Temp.	$p_i(\%)$
1	−1	−1	−1	−1	5.000	8.50
2	−1	−1	−1	−1	34.911	3.87
3	−1	−1	−1	1	28.818	6.60
4	−1	−1	1	−1	31.261	10.10
5	−1	1	−1	−1	5.000	7.60
6	−1	1	−1	−1	21.550	3.15
7	−1	1	−1	1	5.000	8.18

Support point	Alg.	Sca.	Res.	Com.	Temp.	$p_i(\%)$
8	−1	1	1	1	14.674	5.00
9	−1	1	1	1	34.980	9.49
10	1	−1	−1	1	5.000	9.80
11	1	−1	1	−1	5.000	4.70
12	1	−1	1	1	5.000	9.92
13	1	1	1	−1	5.000	10.79
14	1	1	1	1	5.000	2.30

Designs when Theoretical Results Are Not Available

- Yang et al. (2011) developed a complete class approach to finding optimal designs under GLMs that can be applied when all factors are continuous. A disadvantage of this approach is that it requires one factor to be unbounded, while in practice all factors are bounded due to physical constraints.
- Consider the design problem given in Stufken and Yang (2012), where all three factors are continuous with $x_1 \in [-2, 2]$, $x_2 \in [-1, 1]$, and $x_3 \in (-\infty, \infty)$. We consider the nominal values: $\boldsymbol{\beta} = (1, -0.5, 0.5, 1)^T$ and main-effects logit model.

Design by Yang et al. (2011) - unrestricted design space

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

X_1	X_2	X_3	p_i
2	−1	1.544	0.125
2	−1	−0.544	0.125
2	1	0.544	0.125
2	1	−1.544	0.125

d-QPSO algorithm-generated locally D-optimal design with a sufficiently large design space

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

Design found by the *d*-QPSO algorithm when the third factor was constrained further to $[-2, 2]$

X_1	X_2	X_3	p_i
−2	−1	−2.000	0.212
−2	1	−1.727	0.208
2	−1	−0.743	0.076
2	−1	1.757	0.215
2	1	−1.750	0.214
2	1	0.71	0.075

Conclusions

- In this work we proposed a novel and flexible *d*-QPSO algorithm to find several types of D-optimal designs for GLMs with both discrete and continuous factors and a binary response.
- We applied the *d*-QPSO algorithm to find a locally D-optimal design for an additive model, for an experiment with as many as ten factors. The resulting design has 12 support points.
- The *d*-QPSO algorithm can generate locally D-optimal designs with the same or fewer or more support points than many other algorithms. Having fewer support points can be desirable due to cost issues. Having more support points can be an advantage to perform a lack of fit test.
- These particle swarm type algorithms are applicable to a wide range of problems; we have had success applying them to design of experiments with ordered categorical, Poisson, and gamma responses.

References

- Sun, J., Feng, B., and Xu, W. (2004). Particle swarm optimization with particles having quantum behavior. In Evolutionary Computation, 2004. CEC2004. volume 1, pages 325331. IEEE.
- Sun, J., Xu, W., and Fang, W. (2006). Quantum-behaved particle swarm optimization with a hybrid probability distribution. In Pacific Rim International Conference on Artificial Intelligence, pages 737746. Springer.