Generating and Assessing Exact *G*-Optimal Designs

Douglas C. Montgomery
Regents' Professor of Industrial Engineering and Statistics
ASU Foundation Professor of Engineering
School of Computing, Informatics and Decision Systems Engineering
Arizona State University, Tempe, AZ 85287 USA

Joint work with:

Myrta Rodriguez
Tecnológico de Monterrey CSN, Hermosillo, Sonora, México.
Bradley Jones
SAS Institute, SAS Campus Drive, Cary, NC 27513, USA
Connie M. Borror
Arizona State University West, Phoenix, AZ 85069-7100, USA

Rodriguez, M., Jones, B., Borror, C.M., and Montgomery, D.C. (2010), "Generating and Assessing Exact *G*-Optimal Designs", *Journal of Quality Technology*, Vol. 42, No. 1, pp. 3-29.

Design Optimality

- One of the truly great paradigm shifts in DOX
- Can create custom designs for almost any situation
- Modern software makes this easy for at least some optimality criteria
- What optimality criteria should we use?

Three Criteria:

The *D*-criteria: **M** = $\frac{\mathbf{X'X}}{N}$

$$D_{eff} = \left[\frac{|\mathbf{X'X}|}{Max[|\mathbf{X'X}|]} \right]^{1/p}$$

Maximize the determinant of **M**

The *G*-criteria: Maximize the *SPV* over *R*

$$G_{eff} = \frac{p}{\max_{x \in R}(SPV)}$$

$$SPV = \frac{NVar[\hat{y}(\mathbf{x})]}{\sigma^2} = N\mathbf{x}'^{(m)} \left(\mathbf{X'X}\right)^{-1} \mathbf{x}^{(m)}$$

The *I*-criteria: Minimize the average prediction variance over *R*

$$I = \frac{1}{A} \int_R \frac{NVar[\hat{y}(\mathbf{x})]}{\sigma^2} d\mathbf{x}$$

$$I_{eff} = \frac{Min(APV(D))}{APV(D)}$$

Which Criterion should I Use?

- First-order models, first-order models with interaction
  - Objective is usually factor screening
  - Parameter estimation is key
  - Use the *D*-criterion
- Second-order models, mixture models
  - Response estimation is usually of primary importance
  - Use the *G* or *I* criterion

Design Construction

- The coordinate exchange approach – candidate-free, very efficient
- Construction of *D* and *I*-optimal (or highly efficient) designs pretty easy
- Construction of *G*-optimal designs is more challenging
  - There's an ugly optimization problem to be solved at each step

G-Optimal Designs

- It's a combinatorial optimization problem
- Approaches:
  - Branch and bound
  - Simulated annealing
  - Genetic algorithms
  - see Borkowski (2003), Goldfarb, Borror, Montgomery, and Anderson-Cook (2005), Haines (1987), Hamada, Martz, Reese, and Wilson (2001), Heredia-Langner, Carlyle, Montgomery, Borror, and Runger (2003), Heredia-Langner, Montgomery, Carlyle, and Borror (2004), Ortiz, Simpson, and Pignatiello (2004), Park, Richardson, Montgomery, Ozol-Godfrey, Borror, and Anderson-Cook (2005), Welch (1984), and Zhou (2001).

- Borkowski, J. J. (2003). “Using a Genetic Algorithm to Generate Exact Small Response Surface Designs”. *Journal of Probability and Statistical Science* 1, pp. 65–88.
- Goldfarb, H. B.; Borror, C. M.; Montgomery, D. C.; and Anderson-Cook, C. M. (2005). “Using Genetic Algorithms to Generate Mixture-Process Experimental Designs Involving Control and Noise Variables”. *Journal of Quality Technology* 37, pp. 60–74.
- Haines, L. M. (1987). “The Application of the Annealing Algorithm to the Construction of Exact Optimal Designs for Linear-Regression Models”. *Technometrics* 29, pp. 439–447.
- Hamada, M.; Martz, H. E.; Reese, C. S.; and Wilson, A. G. (2001). “Finding Near-Optimal Bayesian Experimental Designs via Genetic Algorithms”. *The American Statistician* 55, pp.175–181.
- Heredia-Langner, A.; Carlyle, W. M.; Montgomery, D. C.; Borror, C. M.; and Runger, G. C. (2003). “Genetic Algorithms for the Construction of D-optimal Designs”. *Journal of Quality Technology* 35, pp. 8–46.
- Heredia-Langner, A.; Montgomery, D. C.; Carlyle, W.M.; and Borror, C. M. (2004). “Model-Robust Optimal Designs: A Genetic Algorithm Approach”. *Journal of Quality Technology* 36, pp. 263–279.
- Welch, W. J. (1984). “Computer-Aided Design of Experiments for Response Estimation”. *Technometrics* 26, pp. 217–224.

Cases Considered

- Second-order response surface models
- Cubodial region of interest
- Cases considered:
  - 2 factors; N = 6, 7, 8, 9, 10, 11, 12
  - 3 factors; N = 10, 11, 12, 13, 14, 15, 16
  - 4 factors; N = 15, 20, 24
  - 5 factors; N = 21, 26, 30

- We report the best designs produced by the optimality criteria *D*, *I*, and *G*.
- *D*-optimal and *I*-optimal designs were created using JMP Version 7.
- The *G*-optimal designs are created using the *G*-criterion and Brent's minimization algorithm* programmed in JSL.
- The starting designs or *parents* include *D*-optimal, *I*-optimal, completely random designs and sometimes a specific design.
- The execution time for the program is minimal (in seconds) and could easily be implemented in commercially available software.
- The JMP program used to create designs can be obtained from Myrta.

*Brent, R. P. (1973). *Algorithms for Minimization without Derivatives*. Englewood Cliffs, NJ: Prentice Hall.

Eye Chart!!

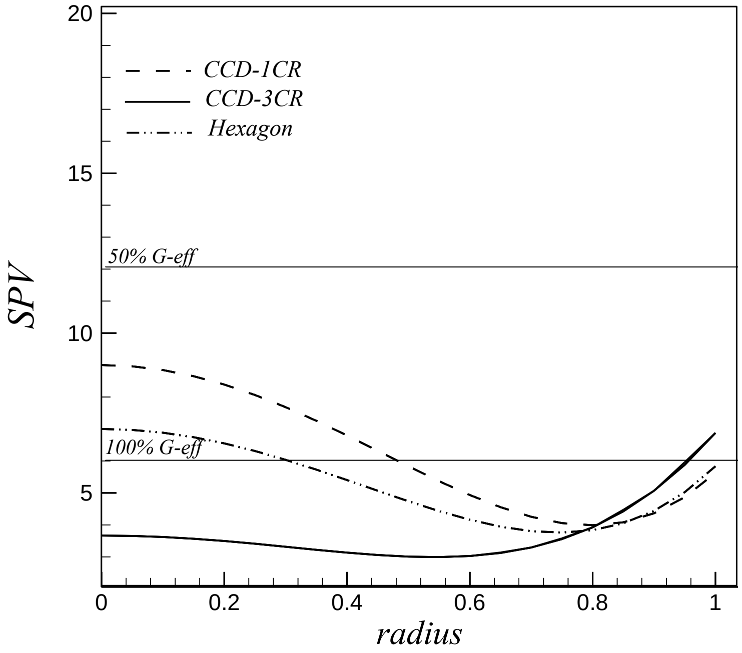
n	Optimal Design	*G*-eff.	*D*-eff.	*I*-eff.	*n*	Optimal Design	*G*-eff.	*D*-eff.	*I*-eff.
2 Factors					**3 Factors**				
6	G	100.00	94.60	92.40	10	G	100.00	91.40	79.06
	D	71.82	100.00	93.67		D	77.40	100.00	96.84
	I	57.96	91.17	100.00		I	69.38	97.56	100.00
7	G	100.00	88.06	75.10	11	G	100.00	93.27	74.97
	D	78.29	100.00	86.93		D	78.30	100.00	70.97
	I	56.41	85.11	100.00		I	63.67	91.51	100.00
8	G	100.00	97.55	81.05	12	G	100.00	93.04	69.39
	D	75.61	100.00	88.45		D	69.33	100.00	70.38
	I	64.94	89.70	100.00		I	59.91	87.66	100.00
9	G	100.00	93.86	81.28	13	G	100.00	98.50	68.90
	D	95.87	100.00	94.76		D	95.78	100.00	72.77
	I	70.44	90.71	100.00		I	65.17	86.80	100.00
10	G	100.00	94.31	75.45	14	G	100.0	99.95	98.21
	D	88.40	100.00	84.29		D	93.28	100.00	76.78
	I	87.54	97.37	100.00		I	53.08	86.10	100.00
11	G	100.00	93.93	70.88	15	G	100.00	99.57	90.17
	D	82.25	100.00	78.14		D	93.16	100.00	72.63
	I	79.77	92.81	100.00		I	99.65	97.14	100.00
12	G	100.00	89.21	73.90	16	G	100.00	96.94	95.39
	D	78.30	100.00	75.78		D	92.14	100.00	71.03
	I	74.39	87.56	100.00		I	98.97	93.65	100.00
4 Factors					**5 Factors**				
15	G	100.00	98.01	75.89	21	G	100.00	98.89	65.20
	D	98.28	100.00	78.23		D	92.86	100.00	63.04
	I	76.44	87.01	100.00		I	54.35	78.26	100.00
20	G	100.00	93.75	73.40	26	G	93.08	99.34	80.86
	D	95.18	100.00	65.63		D	86.41	100.0	61.06
	I	82.11	85.48	100.00		I	64.50	82.69	100.00
24	G	99.97	96.59	87.00	30	G	100.00	99.93	60.04
	D	90.71	100.00	61.82		D	98.65	100.00	60.15
	I	55.38	87.79	100.00		I	71.06	87.08	100.00

Results

- We found designs that were identical to published *G*-optimal designs
  - $k = 2, 3$ factors
- There are cases where our designs are better than published designs
  - $K = 4, N = 15; k = 5, N = 21$
- There are cases for which no published designs were available for comparison
  - $K = 4, N = 20; k = 5, N = 30$

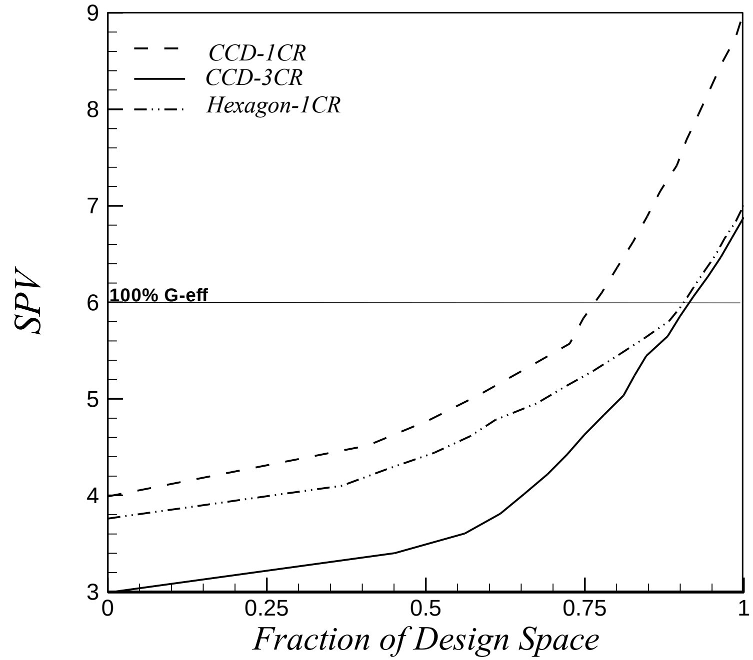
Looking at Efficiencies can be Misleading

The Hexagon and the CCD-3CR have about the same *G*-efficiency (86%)



Variance Dispersion Graph

The CCD-3CR is uniformly superior in the sense that for a given *SPV*, it has the largest fraction of design space at or below this level



Fraction of Design Space Plot

Variance Ratio Fraction of Design Space Plot (VRFDS)

At a specific design point **x** the prediction variance of design *i* is compared to the prediction variance of a reference design:

$$VR_i = \frac{PV_i(\mathbf{x})}{PV_{\text{Ref}}(\mathbf{x})}$$

If $VR_i < 1$ the reference design predicts worse than design *i*; if $VR_i > 1$ the reference design predicts better than design *i*

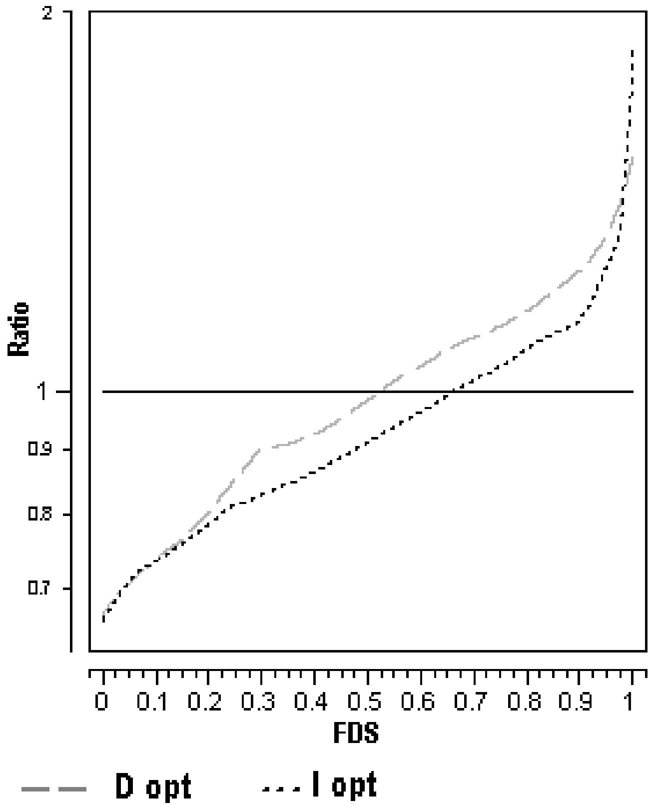
$$VR_D = \frac{PV_D(\mathbf{x})}{PV_G(\mathbf{x})} \qquad VR_I = \frac{PV_I(\mathbf{x})}{PV_G(\mathbf{x})}$$

We used the VRFDS plots to compare the *D* and *I*-optimal designs to the *G*-optimal designs:

$$VR_D = \frac{PV_D(\mathbf{x})}{PV_G(\mathbf{x})} \qquad VR_I = \frac{PV_I(\mathbf{x})}{PV_G(\mathbf{x})}$$

Each VRFDS plot has a horizontal reference line at unity

$K = 2$, $N = 6$

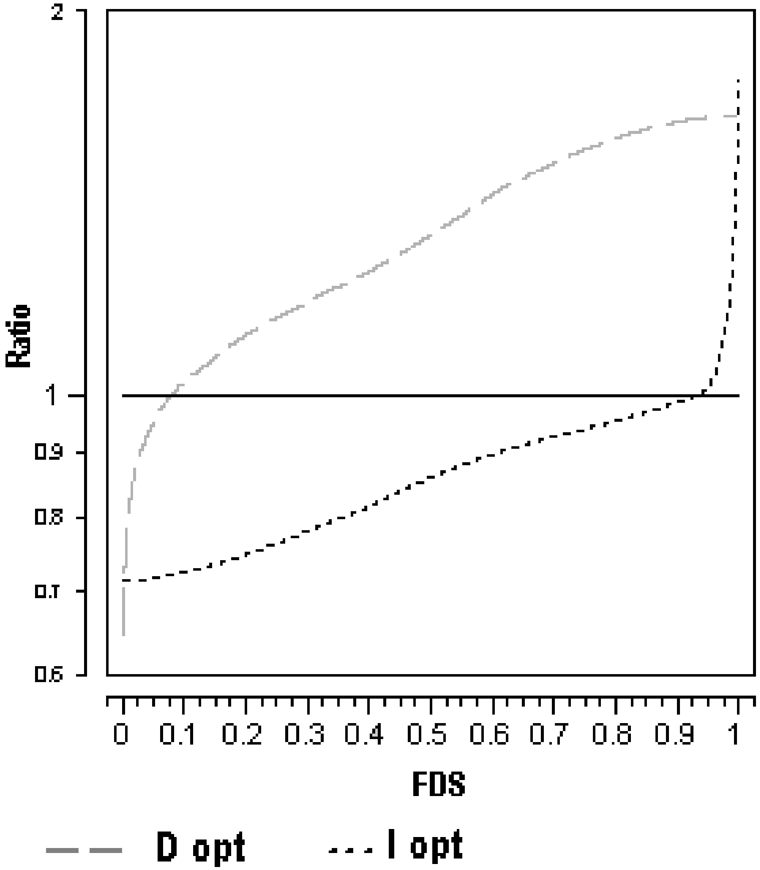


The *I*-optimal design outperforms the *D*-optimal design

The *D*-optimal design outperforms the *G*-optimal design over about 50% of the design space

The *I*-optimal design outperforms the *G*-optimal design over about 2/3 of the design space

$K = 3, N = 15$



Both the *D* and *I*-optimal designs have similar *G*-efficiencies

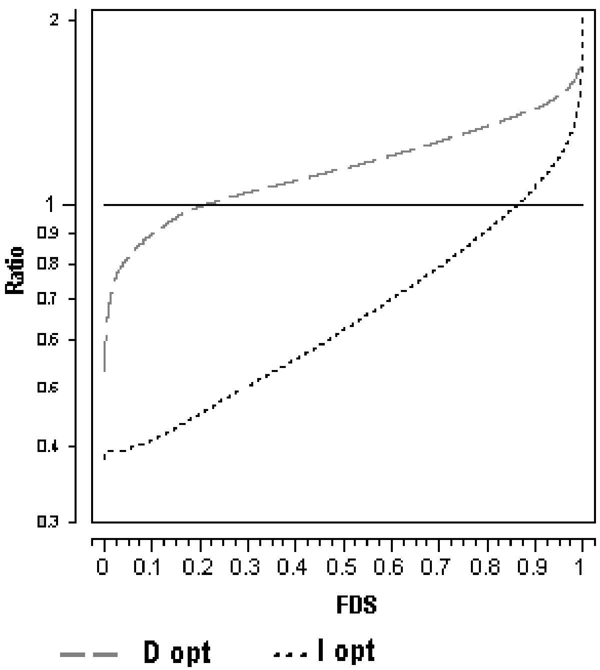
The *I*-optimal design outperforms the *D*-optimal design

The *D*-optimal design is worse than the *G*-optimal design over about 90% of the design space

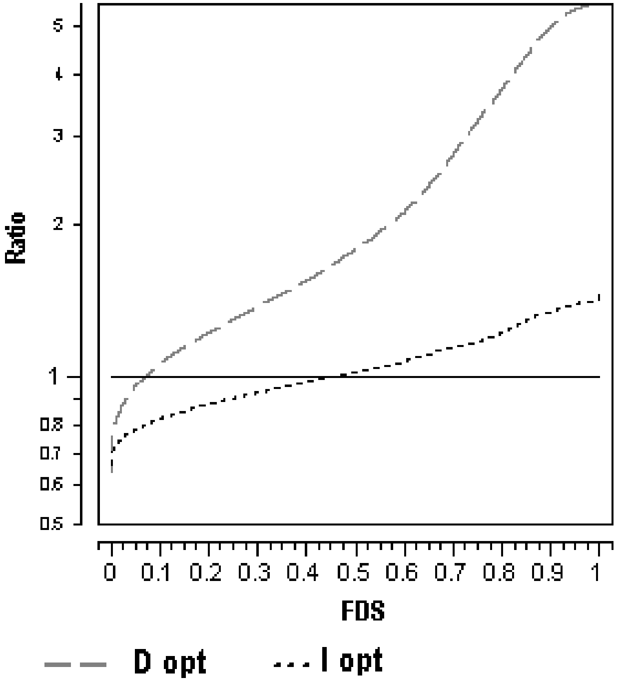
The *I*-optimal design outperforms the *G*-optimal design over about 93% of the design space

As design size increases, generally the *D*-optimal designs result in poorer prediction variance than the *G*-optimal designs

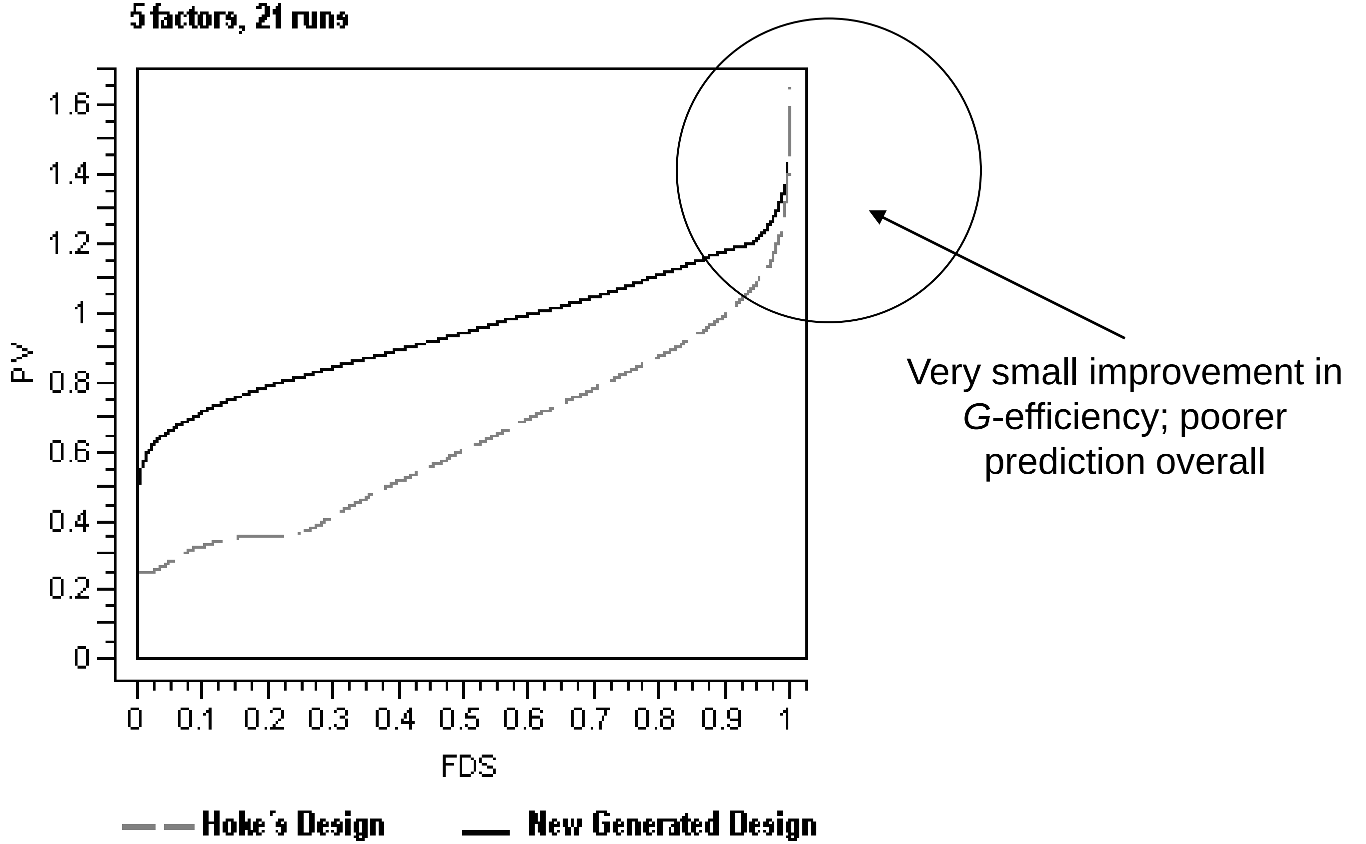
K = 4, *N* = 20



K = 5, *N* = 26



The difference between the new designs and standard designs can be subtle:



Conclusions

- The maximum PV for the *I*-optimal design is always worse that the maximum PV for the *G*-optimal design
- The *I*-optimal designs have smaller PV over most of the design region
- For 60% of the cases considered, the PV for the *I*-optimal design is less than the PV for the *G*-optimal design over 85% of the design space

Conclusions

- The new algorithm produces G-optimal designs in commercially-feasible times (seconds)
- Many *G*-optimal designs are not competitive in PV over a large portion of the design space
- This may be a big price to pay for protection near the boundaries of the region
- So, is it worth it?
- Probably not

Possibly the performance of the *I*-optimal design could be improved near the region boundary by using a weighted objective function:

$$I = \frac{1}{A}\int_R \frac{N w(\mathbf{x}) Var[\hat{y}(\mathbf{x})]}{\sigma^2}\, d\mathbf{x}$$