Model Name	Constrained	Runtime	Ref
BNH-d2-o2	Yes	0.01	[1]
Viennet3-d2-o3	No	0.01	[2]
Viennet2-d2-o3	No	0.01	[2]
Viennet4-d2-o3	No	0.01	[2]
TwoBarTruss-d3-o2	Yes	0.01	[3]
Golinski-d7-o2	No	0.01	[4]
Water-d3-o5	Yes	0.02	[5]
ZDT6-d10-o2	No	0.02	[6]
DTLZ1-d5-o2	No	0.02	[7]
Srinivas-d2-o2	Yes	0.03	[8]
ZDT4-d10-o2	No	0.03	[6]
DTLZ5-d10-o2	No	0.03	[7]
DTLZ2-d10-o2	No	0.03	[7]
DTLZ3-d10-o2	No	0.03	[7]
DTLZ4-d10-o2	No	0.03	[7]
DTLZ2-d10-o4	No	0.03	[7]
DTLZ1-d5-o4	No	0.04	[7]
DTLZ4-d10-o4	No	0.04	[7]
DTLZ3-d10-o4	No	0.04	[7]
DTLZ5-d10-o4	No	0.04	[7]
ZDT3-d30-o2	No	0.05	[6]
ZDT1-d30-o2	No	0.05	[6]
DTLZ6-d20-o4	No	0.05	[7]
ZDT2-d30-o2	No	0.06	[6]
DTLZ6-d20-o2	No	0.06	[7]
Osyczka2-d6-o2	Yes	0.29	[9]
Tanaka-d2-o2	Yes	0.35	[10]
xomoos-d27-o4	No	0.95	[6]
xomogr-d27-o4	No	0.96	
xomofl-d27-o4	No	0.96	
xomoo2-d27-o4	No	0.99	
xomoal-d27-o4	No	1.01	
POM3B-d9-o4	No	6.28	[11], [12]
POM3A-d9-o4	No	218.98	[11], [12]
POM3C-d9-o4	No	445.03	[11], [12]
CDA	No	8100	[13]–[17]

Fig. 1. Model runtimes in milliseconds. Runtimes obtained from running the standalone model 1000 times with arbitrarily chosen inputs. The model name is specced in the following format: *name - number of inputs - number of objectives*. For example, ZDT3-d30-o2 means ZDT3 with 30 decision inputs and 2 objective outputs.

	Model	NSGA-II	GALE	SPEA2
DTLZ	DTLZ1-d5-o2	62%	70%	61%
Toolkit	DTLZ1-d5-o4	78%	83%	78%
	DTLZ2-d10-o2	65%	73%	64%
	DTLZ2-d10-o4	86%	86%	80%
	DTLZ3-d10-o2	65%	73%	64%
	DTLZ3-d10-o4	81%	85%	79%
	DTLZ4-d10-o2	70%	90%	67%
	DTLZ4-d10-o4	93%	92%	90%
	DTLZ5-d10-o2	66%	76%	64%
	DTLZ5-d10-o4	80%	84%	79%
	DTLZ6-d20-o2	67%	71%	67%
	DTLZ6-d20-o4	83%	85%	81%
xomo	xomofl-d27-o4	96%	89%	96%
models	xomogr-d27-o4	97%	89%	97%
	xomoos-d27-o4	97%	88%	97%
	xomoo2-d27-o4	96%	89%	97%
	xomoal-d27-o4	96%	87%	96%
POM3	POM3A-d9-o4	92%	91%	89%
models	POM3B-d9-o4	92%	90%	89%
	POM3C-d9-o4	96%	94%	96%
Constrained	BNH-d2-o2	98%	75%	97%
math	Osyczka2-d6-o2	77%	69%	78%
models	Srinivas-d2-o2	95%	80%	95%
	Tanaka-d2-o2	84%	85%	85%
	TwoBarTruss-d3-o2	95%	78%	95%
	Water-d3-o5	95%	90%	95%
Unconstrained	Golinski-d7-o2	81%	65%	81%
math	Viennet2-d2-o3	73%	73%	73%
models	Viennet3-d2-o3	88%	78%	88%
	Viennet4-d2-o3	90%	77%	89%
	ZDT1-d30-o2	85%	81%	85%
	ZDT2-d30-o2	73%	72%	73%
	ZDT3-d30-o2	84%	80%	85%
	ZDT4-d10-o2	68%	74%	69%
	ZDT6-d10-o2	71%	72%	69%

Fig. 6. Median scores comparing final frontier values to initial populations. Calculated using Equation 1. Lower scores are better. Gray cells are significantly different (statistically) and better than the other values in that row.

Model	n	M
DTLZ1	5	2,4
DTLZ2	10	2,4
DTLZ3	10	2,4
DTLZ4	10	2,4
DTLZ5	10	2,4
DTLZ6	20	2,4

Fig. 5. Parameter settings for DTLZ models. n = Number of decisions and M = number of objectives. Selections for n are made as recommended in [7]. We run for M = 2 and 4 objectives.

References

[1] T. T. Binh and U. Korn, "Mobes: A multiobjective evolution strategy for constrained optimization problems," in *IN PROCEEDINGS OF THE THIRD INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS (MENDEL97*, 1997, pp. 176–182.

[2] R. Viennet, C. Fonteix, and I. Marc, "Multicriteria optimization using genetic algorithms for determining a pareto set." *International Journal of Systems Science*, pp. 255–260, 1996.

[3] D. Chafekar, J. Xuan, and K. Rasheed, "Constrained multi-objective optimization using steady state genetic algorithms," in *In Proceedings of Genetic and Evolutionary Computation Conference.* Springer-Verlag, 2003, pp. 813–824.

[4] A. Kurpati, S. Azarm, and J. Wu, "Constraint handling improvements for multiobjective genetic algorithms," *Structural and Multidisciplinary Optimization*, vol. 23, no. 3, pp. 204–213, 2002. [Online]. Available: http://dx.doi.org/10.1007/s00158-002-0178-2

[5] T. Ray, K. Tai, and K. Seow, "An evolutionary algorithm for multiobjective optimization." *Engineering Optimization*, pp. 399–424, 2001.

[6] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, Jun. 2000. [Online]. Available: http://dx.doi.org/10.1162/106365600568202

[7] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multi-Objective Optimization," Computer Engineering and Networks Laboratory (TIK), ETH Zurich, TIK Report 112, Jul. 2001.

[8] N. Srinivas and K. Deb, "Multiobjective optimization using non-dominated sorting in genetic algorithms," *Evolutionary Computa-*

Constrained Multi-Objective Functions				
Name	n	Objectives	Constraints	Variable Bounds
BNH	2	$f_1(\vec{x}) = 4 * x_1^2 + 4x_2^2$ $f_2(\vec{x}) = (x_1 - 5)^2 + (x_2 - 5)^2$	$g_1(\vec{x}) = ((x_1 - 5)^2 + 2 * x_2^2) <= 25$ $g_2(\vec{x}) = ((x_1 - 8)^8 + (x_2 + 3)^2) >= 7.7$	$0 <= x_1 <= 5$ $0 <= x_2 <= 3$
Osyczka 2	6	$A(\vec{x}) = 25(x_1 - 2)^2 + (x_2 - 2)^2$ $B(\vec{x}) = (x_3 - 1)^2 * (x_4 - 4)^2 +$ $(x_5 - 2)^2$ $f_1(\vec{x}) = 0 - A - B$ $f_2(\vec{x}) = x_1^2 + x_2^2 + x_3^2 +$ $x_4^2 + x_5^2 + x_6^2$	$g_1(\vec{x}) = x_1 + x_2 - 2 >= 0$ $g_2(\vec{x}) = 6 - x_1 - x_2 >= 0$ $g_3(\vec{x}) = 2 - x_2 + x_1 >= 0$ $g_4(\vec{x}) = 2 - x_1 + 3x_2 >= 0$ $g_5(\vec{x}) = 4 - (x_3 - 3)^2 - x_4 >= 0$ $g_6(\vec{x}) = (x_5 - 3)^3 + x_6 - 4 >= 0$	$0 <= x_1, x_2, x_6 <= 10$ $1 <= x_3, x_4 <= 5$ $0 <= x_5 <= 6$
Srinivas	2	$f_1(\vec{x}) = (x_1 - 2)^2 + (x_2 - 1)^2 + 2$ $f_2(\vec{x}) = 9x_1 - (x_2 - 1)^2$	$g_1(\vec{x}) = x_2 + 9x_1 >= 6$ $g_2(\vec{x}) = -x_2 + 9x_1 >= 1$	$-20 <= x <= 20$
Tanaka	2	$f_1(\vec{x}) = x_1$ $f_2(\vec{x}) = x_2$	$A(\vec{x}) = 0.1 \cos(16 \arctan(\frac{x_1}{x_2}))$ $g_1(\vec{x}) = 1 - x_1^2 - x_2^2 + A <= 0$ $g_2(\vec{x}) = (x_1 - 0.5)^2 +$ $(x_2 - 0.5)^2 <= 0.5$	$-\pi <= x <= \pi$
Two-bar Truss	3	$s_1(\vec{x}) = \frac{20*\sqrt{16+x_3^2}}{(x_1*x_3)}$ $s_2(\vec{x}) = \frac{80*\sqrt{1+x_3^2}}{(x_2*x_3)}$ $f_1(\vec{x}) = x_1 * \sqrt{16 * x_3^2} + x_2 * \sqrt{1 + x_3^2}$ $f_2(\vec{x}) = max(s_1, s_2)$	$s_1(\vec{x}) = \frac{20*\sqrt{16+x_3^2}}{(x_1*x_3)}$ $s_2(\vec{x}) = \frac{80*\sqrt{1+x_3^2}}{(x_2*x_3)}$ $g_1(\vec{x}) = (max(s_1, s_2)) <= 100000$	$0 <= x_1, x_2 <= 0.01$ $1 <= x_3 <= 3$
Water	3	$f_1(\vec{x}) = 106780.37 * (x_2 + x_3) + 61704.67$ $f_2(\vec{x}) = 3000 * x_1$ $f_3(\vec{x}) = \frac{(305700*2289*x_2)}{((0.06*2289)**0.65)}$ $E(\vec{x}) = e^{-39.75*x_2+9.9*x_3+2.74}$ $f_4(\vec{x}) = 250 * 2289 * x_2 * E(\vec{x})$ $f_5(\vec{x}) = 25 * \frac{1.39}{x_1*x_2+4940*x_3-80}$	$g_1(\vec{x}) = (\frac{1-0.00139}{x_1*x_2} + 4.94 * x_3 - 0.08)$ $g_2(\vec{x}) = (\frac{1-0.000306}{x_1*x_2} + 1.082 * x_3 - 0.0986)$ $g_3(\vec{x}) = (\frac{5000-12.307}{x_1*x_2} + 4.9408 * x_3 - 4051.02)$ $g_4(\vec{x}) = (\frac{16000-2.09}{x_1*x_2} + 804633 * x_3 - 696.71)$ $g_5(\vec{x}) = (\frac{10000-2.138}{x_1*x_2} + 7883.39 * x_3 - 705.04)$ $g_6(\vec{x}) = (\frac{2000-0.417}{x_1*x_2} + 1721.26 * x_3 - 136.54)$ $g_7(\vec{x}) = (\frac{550-0.164}{x_1*x_2} + 631.13 * x_3 - 54.48)$ $g_i(\vec{x}) \geq 0$	$\frac{1}{100} <= x_1 <= \frac{45}{100}$ $\frac{1}{100} <= x_2, x_3 <= \frac{1}{10}$

Fig. 2. Constrained standard mathematical test problems. Note: Objectives to be minimized unless otherwise denoted.

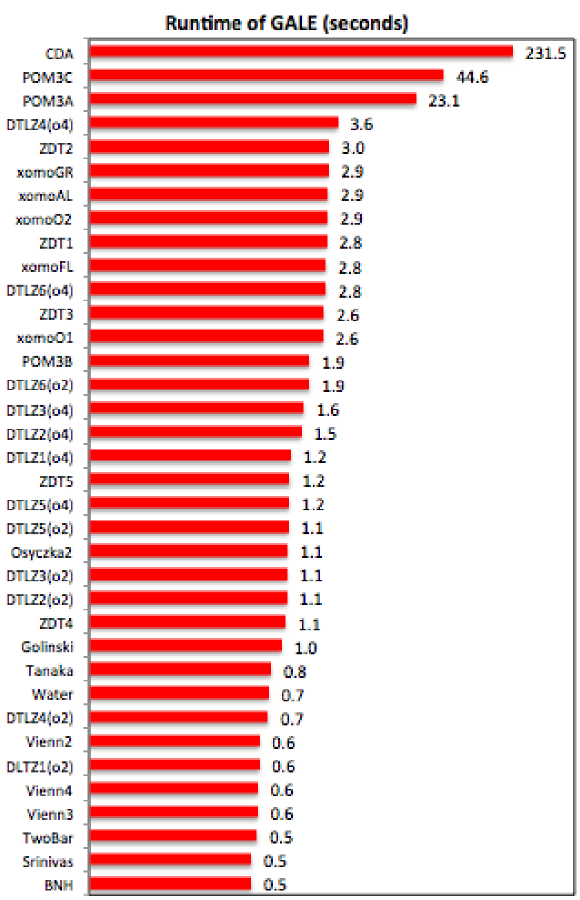

Fig. 7. GALE, mean runtime in seconds.

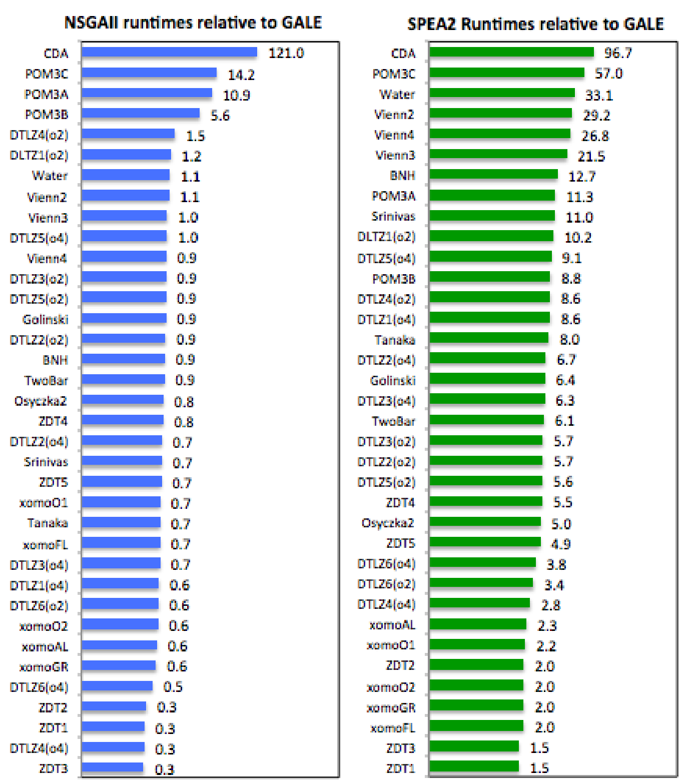

Fig. 8. NSGA-II, SPEA2, runtimes, relative to GALE (mean values over all runs) e.g., with SPEA2, ZDT1 ran 1.5 times slower than GALE.

tion, vol. 2, no. 3, pp. 221–248, 1994.
[9] A. Osyczka and S. Kundu, "A new method to solve generalized

Unconstrained Multi-Objective Functions			
Name	n	Objectives	Variable Bounds
Golinksi	7	$r = 0.7854$ $s = 14.933$ $t = 43.0934$ $u = -1.508$ $v = 7.477$ $A(\vec{x}) = rx_1x_2^2(\frac{10x_3^2}{3.0} + sx_3 - t)$ $B(\vec{x}) = ux_1(x_6^2 + x_7^2) + v(x_6^3 + x_7^3) + r(x_4 * x_6^2 + x_5 * x_7^2)$ $aux(\vec{x}) = 745.0\frac{x_4}{x_2 * x_3}$ $f_1(\vec{x}) = A + B$ $f_2(\vec{x}) = \frac{\sqrt{aux^2 + 1.69e7}}{0.1x_6^3}$	$2.6 <= x_1 <= 3.6$ $0.7 <= x_2 <= 0.8$ $17.0 <= x_3 <= 28.0$ $7.3 <= x_4, x_5 <= 8.3$ $2.9 <= x_6 <= 3.9$ $5.0 <= x_7 <= 5.5$
Viennet 2	2	$f_1(\vec{x}) = \frac{(x_1-2)(x_1-2)}{2} + \frac{(x_1+1)(x_1+1)}{13} + 3$ $f_2(\vec{x}) = \frac{(x_1+x_2-3)(x_1+x_2-3)}{36} + \frac{(-x_1+x_2+2)(-x_1+x_2+2)}{8} - 17$ $f_3(\vec{x}) = \frac{(x_1+2x_2-1)(x_1+2x_2-1)}{175} +$ $\frac{(2x_2-x_1)(2x_2-x_1)}{17} - 13$	$-4 <= x_i <= 4$
Viennet 3	2	$A(\vec{x}) = 3 * x_1 - 2 * x_2 + 4$ $B(\vec{x}) = x_1 - x_2 + 1$ $f_1(\vec{x}) = 0.5 * (x_1^2 + x_2^2) + sin(x_1^2 + x_2^2)$ $f_2(\vec{x}) = \frac{A^2}{8 + \frac{B^2}{27} + 15}$ $f_3(\vec{x}) = \frac{1}{x_1^2 + x_2^2 + 1} - 1.1 * e^{-(x_1^2) - (x_2^2)}$	$-3 <= x_i <= 3$
Viennet 4	2	$f_1(\vec{x}) = \frac{(x_1-2)(x_1-2)}{2} + \frac{(x_1+1)(x_1+1)}{13} + 3$ $f_2(\vec{x}) = \frac{(x_1+x_2-3)(x_1+x_2-3)}{175} + \frac{(2*x_2-x_1)*(2*x_2-x_1)}{17} - 13$ $f_3(\vec{x}) = \frac{(3*x_1-2*x_2+4)*(3*x_1-2*x_2+4)}{8} +$ $\frac{(x_1-x_2+1)(x_1-x_2+1)}{27} + 15$	$-4 <= x_i <= 4$
ZDT1	30	$f_1(\vec{x}) = x_1$ $f_2(\vec{x}) = g * (1 - \sqrt{\frac{x_1}{g}})$ $g(\vec{x}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n}(x_i)$	$0 <= x_i <= 1$
ZDT2	30	$f_1(\vec{x}) = x_1$ $f_2(\vec{x}) = g * (1 - (\frac{x_1}{g})^2)$ $g(\vec{x}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n}(x_i)$	$0 <= x_i <= 1$
ZDT3	30	$f_1(\vec{x}) = x_1$ $f_2(\vec{x}) = g * (1 - \sqrt{(\frac{x_1}{g})} - \frac{x_1}{g} * sin(10 * \pi * x_1))$ $g(\vec{x}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n}(x_i)$	$0 <= x_i <= 1$
ZDT4	10	$f_1(\vec{x}) = x_1$ $f_2(\vec{x}) = g * (1 - \sqrt{(\frac{x_1}{g})} - \frac{x_1}{g} * sin(10 * \pi * x_1))$ $g(\vec{x}) = 1 + 10 * (n - 1) + \sum_2^n(x_i^2 - 10 * cos(4 * \pi * x_i))$	$0 <= x_1 <= 1$ $-5 <= x_2, ..., x_{10} <= 5$
ZDT6	10	$f_1(\vec{x}) = 1 - e^{-4*x_1} * sin(6 * \pi * x_1)^6$ $f_2(\vec{x}) = g * (1 - (\frac{f_1(\vec{x})}{g})^2)$ $g(\vec{x}) = 1 + 9 * \frac{\sum_2^n x_i}{(n-1)^{0.25}}$	$0 <= x_i <= 1$

Fig. 3. Unconstrained standard mathematical test problems. Note: all objectives are to be minimized unless otherwise denoted.

multicriteria optimization problems using the simple genetic algorithm," *Structural optimization*, vol. 10, no. 2, pp. 94–99, 1995. [Online]. Available: http://dx.doi.org/10.1007/BF01743536

[10] M. Tanaka, H. Watanabe, Y. Furukawa, and T. Tanino, "Ga-based decision support system for multicriteria optimization," in *Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on*, vol. 2, Oct 1995, pp. 1556–1561 vol.2.

[11] D. Port, A. Olkov, and T. Menzies, "Using simulation to investigate requirements prioritization strategies," in *Automated Software Engineering*, 2008, pp. 268–277.

[12] B. Lemon, A. Riesbeck, T. Menzies, J. Price, J. D'Alessandro, R. Carlsson, T. Prifiti, F. Peters, H. Lu, and D. Port, "Applications of simulation and ai search: Assessing the relative merits of agile vs traditional software development," in *IEEE ASE'09*, 2009.

[13] S. Y. Kim, "Model-based metrics of human-automation function allocation in complex work environments," Ph.D. dissertation, Georgia Institute of Technology, 2011.

[14] A. R. Pritchett, H. C. Christmann, and M. S. Bigelow, "A simulation engine to predict multi-agent work in complex, dynamic, heterogeneous systems," in *IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support*, Miami Beach, FL, 2011.

[15] K. M. Feigh, M. C. Dorneich, and C. C. Hayes, "Toward a characterization of adaptive systems: A framework for researchers and system designers," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 54, no. 6, pp. 1008–1024, 2012.

[16] S. Y. Kim, A. R. Pritchett, and K. M. Feigh, "Measuring human-automation function allocation," *Journal of Cognitive Engineering and Decision Making*, 2013.

[17] A. R. Pritchett, S. Y. Kim, and K. M. Feigh, "Modeling human-automation function allocation," *Journal of Cognitive Engineering and Decision Making*, 2013.

DTLZ Family of Models			
Name	n	Objectives	Variable Bounds
DTLZ1	n	$f_1(\vec{x}) = 0.5 * x_1 * x_2 * ... * x_{M-1}(1 + g(x_M))$ $f_2(\vec{x}) = 0.5 * x_1 * x_2 * ... * (1 - x_{M-1})(1 + g(x_M))$ ... $f_{M-1}(\vec{x}) = 0.5 * x_1 * (1 - x_2)(1 + g(x_M))$ $f_M(\vec{x}) = 0.5 * (1 - x_1)(1 + g(x_M))$ $g(\vec{x}) = 100 * (\text{x\_M} + \sum(x_i - 0.5)^2 - cos(20\pi(x_i - 0.5)))$	$0 <= x_i <= 1$
DTLZ2	n	$f_1(\vec{x}) = (1 + g(X_M))cos(x_1 * \frac{\pi}{2})...cos(x_{M-1} * \frac{\pi}{2})$ $f_2(\vec{x}) = (1 + g(X_M))cos(x_1 * \frac{\pi}{2})...sin(x_{M-1} * \frac{\pi}{2})$ ... $f_M(\vec{x}) = (1 + g(X_M))sin(x_1 * \frac{\pi}{2})$ $g(\vec{x}) = \sum(x_i - 0.5)^2$	$0 <= x_i <= 1$
DTLZ3	n	$f_1(\vec{x}) = (1 + g(X_M))cos(x_1 * \frac{\pi}{2})...cos(x_{M-1} * \frac{\pi}{2})$ $f_2(\vec{x}) = (1 + g(X_M))cos(x_1 * \frac{\pi}{2})...sin(x_{M-1} * \frac{\pi}{2})$ ... $f_M(\vec{x}) = (1 + g(X_M))sin(x_1 * \frac{\pi}{2})$ $g(\vec{x}) = 100 * (\text{x\_M} + \sum(x_i - 0.5)^2 - cos(20\pi(x_i - 0.5)))$	$0 <= x_i <= 1$
DTLZ4	n	$f_1(\vec{x}) = (1 + g(X_M))cos(x_1^\alpha * \frac{\pi}{2})...cos(x_{M-1}^\alpha * \frac{\pi}{2})$ $f_2(\vec{x}) = (1 + g(X_M))cos(x_1^\alpha * \frac{\pi}{2})...sin(x_{M-1}^\alpha * \frac{\pi}{2})$ ... $f_M(\vec{x}) = (1 + g(X_M))sin(x_1^\alpha * \frac{\pi}{2})$ $g(\vec{x}) = \sum(x_i - 0.5)^2$	$0 <= x_i <= 1$
DTLZ5	n	$f_1(\vec{x}) = (1 + g(X_M))cos(\theta_1 * \frac{\pi}{2})...cos(\theta_{M-1} * \frac{\pi}{2})$ $f_2(\vec{x}) = (1 + g(X_M))cos(\theta_1 * \frac{\pi}{2})...sin(\theta_{M-1} * \frac{\pi}{2})$ ... $f_M(\vec{x}) = (1 + g(X_M))sin(\theta_1 * \frac{\pi}{2})$ $\theta_i = \frac{\pi}{4(i+g(r))}(1 + 2g(r)x_i)\ for\ i = 2, 3, ..., (M-1)$ $g(\vec{x}) = \sum(x_i - 0.5)^2$	$0 <= x_i <= 1$
DTLZ6	n	$f_1(\vec{x}) = (1 + g(X_M))cos(\theta_1 * \frac{\pi}{2})...cos(\theta_{M-1} * \frac{\pi}{2})$ $f_2(\vec{x}) = (1 + g(X_M))cos(\theta_1 * \frac{\pi}{2})...sin(\theta_{M-1} * \frac{\pi}{2})$ ... $f_M(\vec{x}) = (1 + g(X_M))sin(\theta_1 * \frac{\pi}{2})$ $\theta_i = \frac{\pi}{4(i+g(r))}(1 + 2g(r)x_i)\ for\ i = 2, 3, ..., (M-1)$ $g(\vec{x}) = \sum x_i^{0.1}$	$0 <= x_i <= 1$

Fig. 4. DTLZ test problems. Note: all objectives are to be minimized unless otherwise denoted.