Multi-objective programming: adaptive surrogate based approach with Chebyshev scalarization

Alexis Pospelov

DATADVANCE, IITP RAS

Multi-objective optimization

Solution set

$$X \subset \mathbb{R}^N$$

Objectives

$$f : \mathbb{R}^N \to \mathbb{R}^K$$

Objective vector

$$y = f(x)$$

Set of all feasible objective vectors

$$Y = f(X)$$

Multi-objective optimization problem

$$f \to \min, \quad x \in X$$

Pareto domination

A point $y' \in \mathbb{R}^m$ is better than a point $y'' \in \mathbb{R}^m$ (dominate a point y'') means following

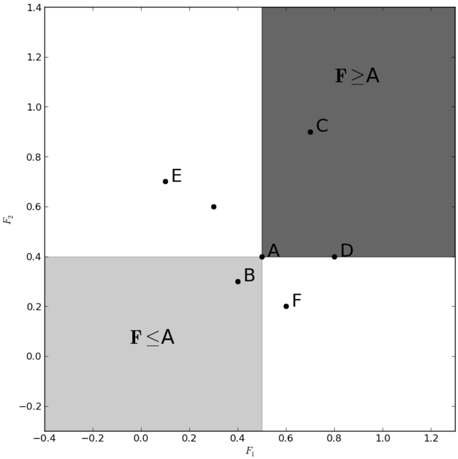
$$\left\{y_i' \leq y_i'',\ i = 1, 2, \ldots, m;\ y' \neq y''\right\}$$

In other words, points dominated by y' belongs to the non-negative cone $y' + \mathbb{R}^m_+$

A set of all non-dominated points is called non-dominated (Pareto) frontier

$$P(Y) = \{y \in Y \mid \left\{y' \in Y \mid y_i' \leq y_i,\ i = 1, 2, \ldots, m;\ y' \neq y\right\} = \emptyset\}$$

Pareto optimality



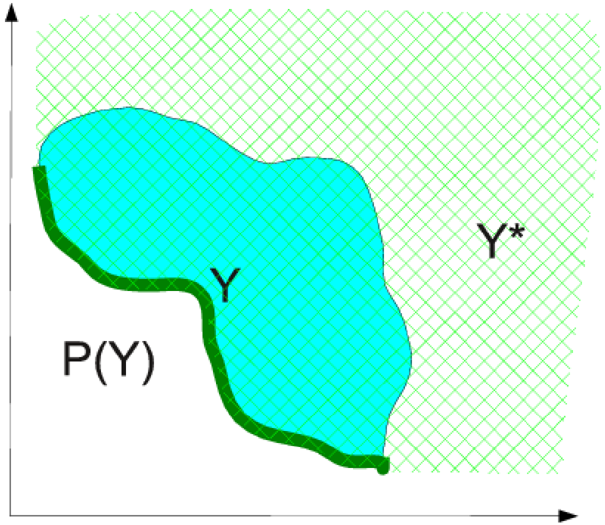
Egdeworth-Pareto hull

Set of all feasible objective vectors

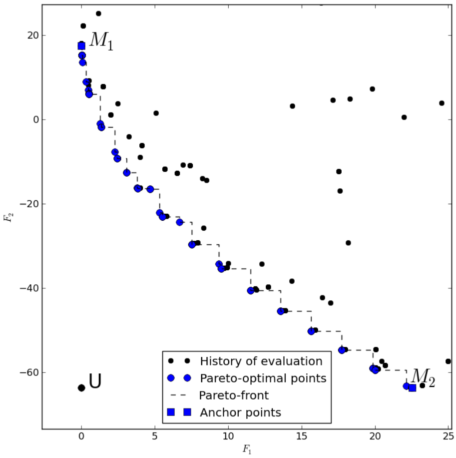
$$Y = f(X)$$

Egdeworth-Pareto hull

$$Y^* = f(X) + \mathbb{R}^m = \\ = P(Y) + \mathbb{R}^m$$



Pareto front



Common approaches

- Genetic algorithms – a wide class of methods, based on biological analogies
- Multi-objective gradients – methods use local model of objective function and constrains
- Scalarization – methods reduce multiobjective problem in series of single objective problems

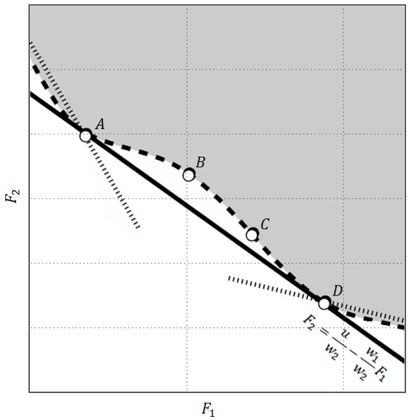
Scalarization

Most common type scalarization approaches are following

- Using scalarization function $U(\mathbf{f}(\cdot))$
- Transfer scalarization into constrains
- Mixed approach

Linear scalarization

- $U(x) = w_1 f_1(x) + w_2 f_2(x) + \ldots + w_K f_K(x)$
- Solve problem $U \to \min, \quad x \in X$



Merits and demerits

- + Multiple single objective optimization approacher can be applied
- - Not all Pareto optimal points can be found
- - Non-uniform coverage of the front
- - Globalization is necessary to be applies multiple times
- - Independent solving of problems leads to loosing information about objective functions

Search for anchor points

- Anchor points estimate range for each objective on Pareto front.
- For seeking anchor points we need to solve single objective problems $f_i(\mathbf{x})$ for each K

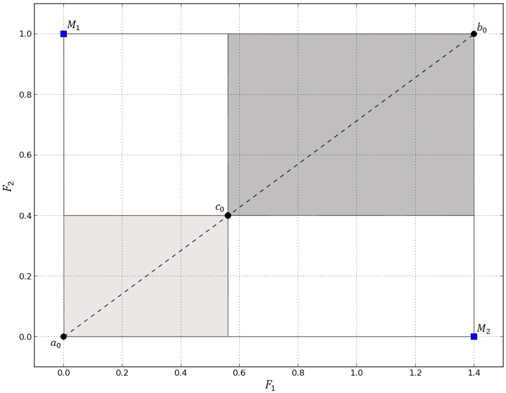
Discovering Pareto front

- We suggest to use following (Chebyshev) scalarization

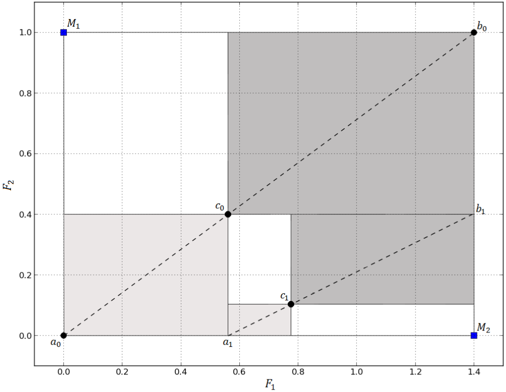
$$U = \max_k u_k$$
$$u_k = \alpha_k(f_k - \beta_k)$$

- Solve single objective optimization problem $U \to \min, \quad x \in X$
- Minimization of such function leads to search Pareto-optimal point along line with angle α, and goes through point β

Adaptive choice of parameters



Adaptive choice of parameters



Merits and demerits

- + Multiple single objective optimization approacher can be applied
- * No all Pareto optimal points can be found
- * Non-uniform coverage of the front
- - Globalization is necessary to be applies multiple times
- - Independent solving of problems leads to loosing information about objective functions

Surrogate modeling

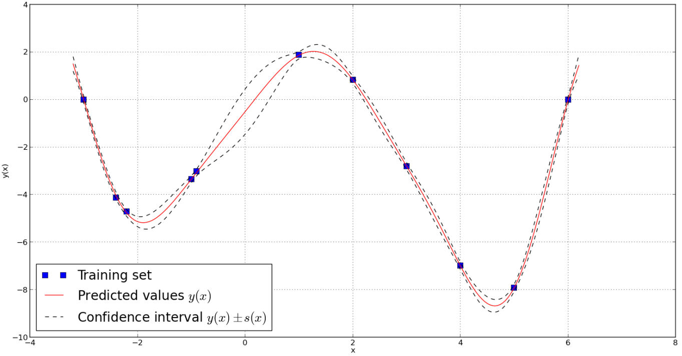
Main ideas are

- Inputs for building models are *training set* – finite set of pairs $S = \{(\mathbf{x_i}, f(\mathbf{x_i})),\ i = 1 \dots n\}$
- Approximation of function using training set allows predicating values without calling real calculation of objective function
- Some type of models are able to estimate error

Kriging

- We assumes that in each point x value of function $y(\mathbf{x})$ is realization of random variable $Y(\mathbf{x})$ with normal distribution with mean μ variance s^2
- The model allows predicating $\mu(\mathbf{x})$ and $s(\mathbf{x})$

Example of surrogating by Kriging



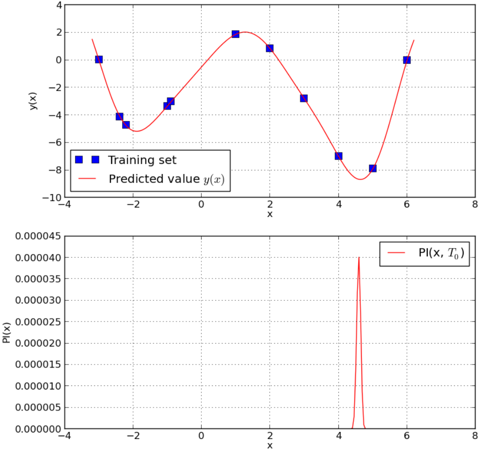
Maximization of probability of improvement

- Let f_{min} is the best know values of goal function
- Let $T < f_{\text{min}}$ is a some value
- Probability to reach value T in point $\mathbf{x}$:

$$PI(\mathbf{x}, T) = \Pr[Y(\mathbf{x}) \leq T] = \Phi\left(\frac{T - \hat{f}(\mathbf{x})}{s(\mathbf{x})}\right)$$

  where $\Phi(\cdot)$ is normal distribution function
- Iterative seeking maximal of $PI(\mathbf{x})$ and adding point to training set we can move toward global optimum of the problem.

Illustration of maximization of probability of improvement



Merits and demerits of maximization of probability of improvement

- + Allows searching global optimum
- + Possible to control number of evaluating points
- - Requires to select target
- - No evident stopping criteria

Multiple target selection

- Choose a set of goal values

$$T(\alpha) = f_{\min} - \alpha(f_{\max} - f_{\min}), \ \alpha \geq 0$$

- Solve probability maximization problem for different $T(\alpha)$
- Clusterize produced point
- Choose a point from each cluster and evaluation function in the point and add it to training set

Overview scheme of algorithm

1. Build training set.
2. Search for anchor points.
3. Discovering Pareto front using adaptive scalarization.
4. Use surrogate model based optimization for each scalar optimization.
5. Keep models (and training sets) obtained on previous optimization steps.
6. Use as stopping criteria number of evaluation heuristically dividing computational budget between steps and at the same time deciding about number of steps.

Starting training set

- Let the optimization problem has box bounds $(L_i \leq x_i \leq U_i)$
- Let n number of points in training set are determinated by dimension by heuristical rule $n = cN$
- Separate linear constrains in problem
- Choose n random point in polytop given by linear constraints of the problem

Mean and standard deviation, $K = 2$

$$\hat{U} = E(U) = \hat{F}_1 \Phi\left(\frac{\hat{F}_1 - \hat{F}_2}{\theta}\right) + \hat{F}_2 \Phi\left(\frac{\hat{F}_2 - \hat{F}_1}{\theta}\right) + \theta\phi\left(\frac{\hat{F}_1 - \hat{F}_2}{\theta}\right)$$

$$E(U^2) = (s_1^2 + \hat{F}_1^2)\Phi\left(\frac{\hat{F}_1 - \hat{F}_2}{\theta}\right) + (s_2^2 + \hat{F}_2^2)\Phi\left(\frac{\hat{F}_2 - \hat{F}_1}{\theta}\right) + $$

$$+ (\hat{F}_1^2 + \hat{F}_2^2)\theta\phi\left(\frac{\hat{F}_1 - \hat{F}_2}{\theta}\right)$$

where $\theta = \sqrt{s_1^2 + s_2^2}$

Mean and standard deviation, $K > 2$

Distribution function for maximum of independent random variables:

$$P(t) = \Pr\{\max_k u_k \le t\} = \Pr\{u_1 \le t \cap \ldots \cap u_K \le t\}$$
$$= \Pr\{u_1 \le t\} \cdots \Pr\{u_K \le t\} = \prod_{k=1}^{K} P_k(t)$$

Calculation moment of obtained destitution

$$E(U^n) = \int_{t=0}^{\infty} n t^{n-1} \left[1 - \Pr\{U > t\} + (-1)^n \Pr\{U < -t\}\right] dt$$

Some numerical results

- Here we compare three algorithms:

  SAS — algorithms proposed in this work
  GTOpt — gradient based method that used local geometry of Pareto front
  NSGA2 — genetic algorithms

Problem ZDT1

$$\min_{x \in Q} |F_1, F_2|$$

$$F_1 = x_1$$

$$F_2 = 1 - \sqrt{\frac{F_1}{g}}$$

$$Q = [0, 1]^N$$

$$\text{where } g = 1 + 9 \sum_{i=2}^{N} \frac{x_i}{N - 1}$$

Problem ZDT1

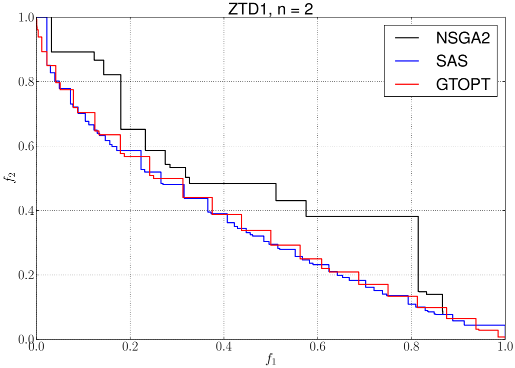
Algorithms	number of evaluations	Q	Time
SAS	**87**	0.0294	61.7c
NSGA2	120	0.179	0.8c
GTOpt	127	0.0307	0.7c

Table: ZDT1, $n = 2$

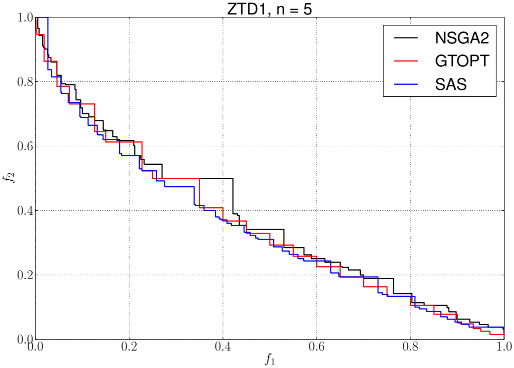
Algorithms	number of evaluations	Q	Time
SAS	**124**	0.0307	171.3c
NSGA2	800	0.081	2.1c
GTOpt	216	0.048	1.1c

Table: ZDT1, $n = 5$

Problem ZDT1



Problem ZDT1



Problem ZDT2

$$\min_{x \in Q} \left| F_1, F_2 \right|$$

$$F_1 = x_1$$

$$F_2 = 1 - \left(\frac{F_1}{g} \right)^2$$

$$Q = [0, 1]^N$$

$$g = 1 + 9 \sum_{i=2}^{N} \frac{x_i}{N - 1}$$

Problem ZDT2

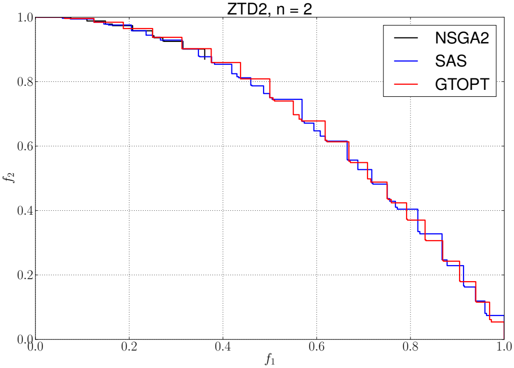
Algorithms	number of evaluations	Q	Time
SAS	**88**	0.031	63.5c
NSGA2	600	0.101	1.8c
GTOpt	114	0.029	0.7c

Table: ZDT2, $n = 2$

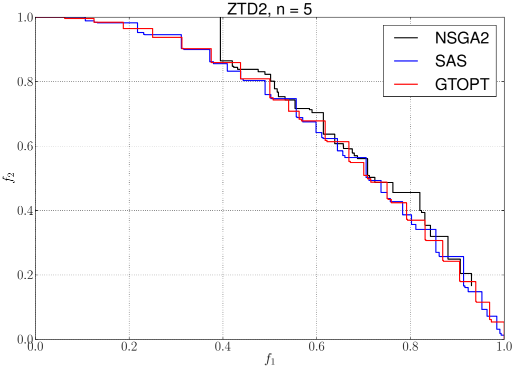
Algorithms	number of evaluations	Q	Time
SAS	**115**	0.032	194.5c
NSGA2	960	0.166	2.5c
GTOpt	211	0.029	1.2c

Table: ZDT2, $n = 5$

Problem ZDT2



Problem ZDT2



Problem ZDT3

$$\min_{x \in Q} \left| F_1, F_2 \right|$$

$$F_1 = x_1$$

$$F_2 = 1 - \sqrt{\frac{F_1}{g}} - \left(\frac{F_1}{g}\right) \sin(10\pi F_1)$$

$$Q = [0, 1]^N$$

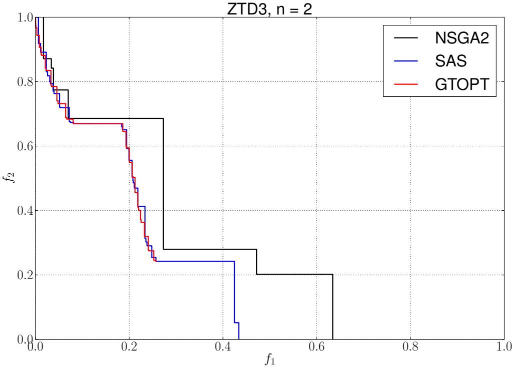
$$g = 1 + 9 \sum_{i=2}^{N} \frac{x_i}{N-1}$$

Problem ZDT3

Algorithms	number of evaluations	Q	Time
SAS	**134**	0.194	206c
NSGA2	240	0.204	1.2c
GTOpt	172	1.015	0.9c

Table: ZDT3, $n = 2$

Problem ZDT3



Conclusion

Properties of proposed algorithm

- Allows to control computational budget
- Quality of approximation is comparable to gradient method with less amount of evaluations of objective functions
- Global Pareto front discovering from the box
- Main weak point is high computational cost (e.g. training large number of surrogate models, global optimization over models).

Thanks for your attention!