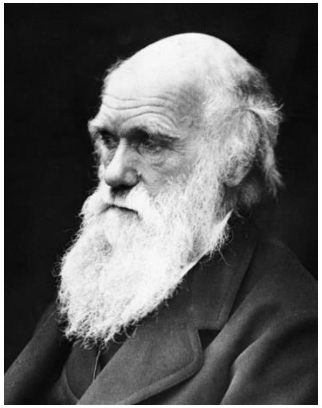
Towards
Safe Evolutionary Optimization

Chao Qian

School of Artificial Intelligence
Nanjing University

http://www.lamda.nju.edu.cn/qianc/
Email: qianc@nju.edu.cn

Evolutionary algorithms



Charles Darwin
1809-1882

C. Darwin, after collecting abundant evidence, developed a theory about how species evolve

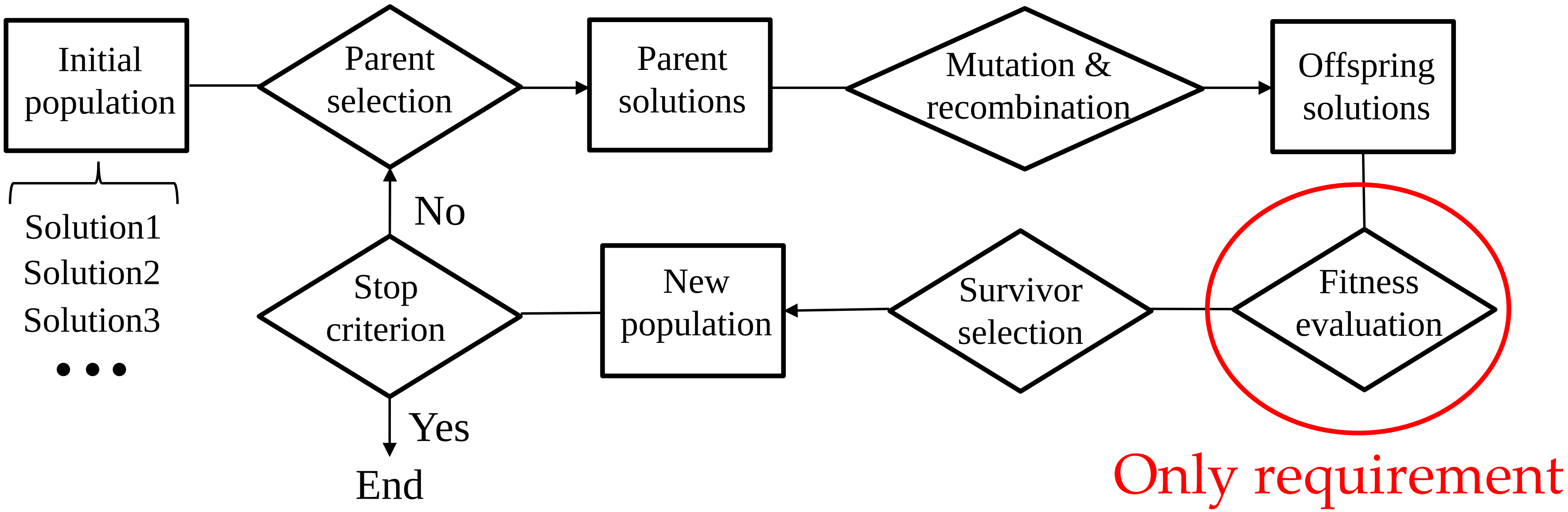
reproduction with variation + nature selection

Evolutionary algorithms: a kind of nature-inspired randomized heuristic optimization algorithms

genetic algorithms, evolutionary strategies, evolutionary programming, particle swarm optimization, ……

Evolutionary algorithms

Common structure of evolutionary algorithms (EAs)

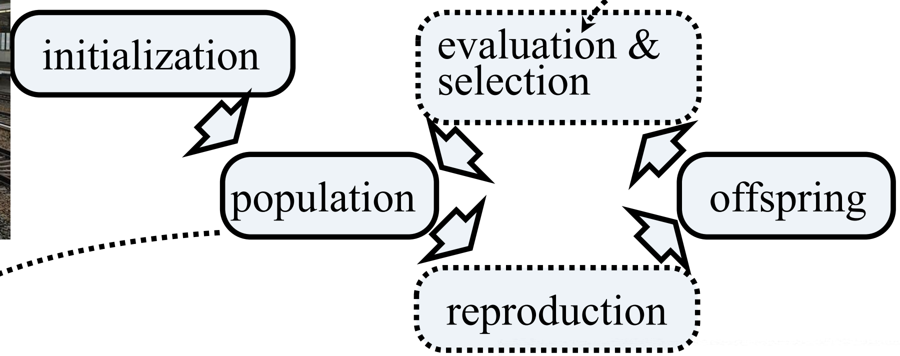


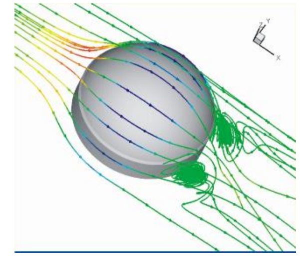
EAs can be applied to solve complex optimization problems

- non-differentiable and non-continuous problems
- problems with multiple objective functions
- problems without explicit objective function formulation

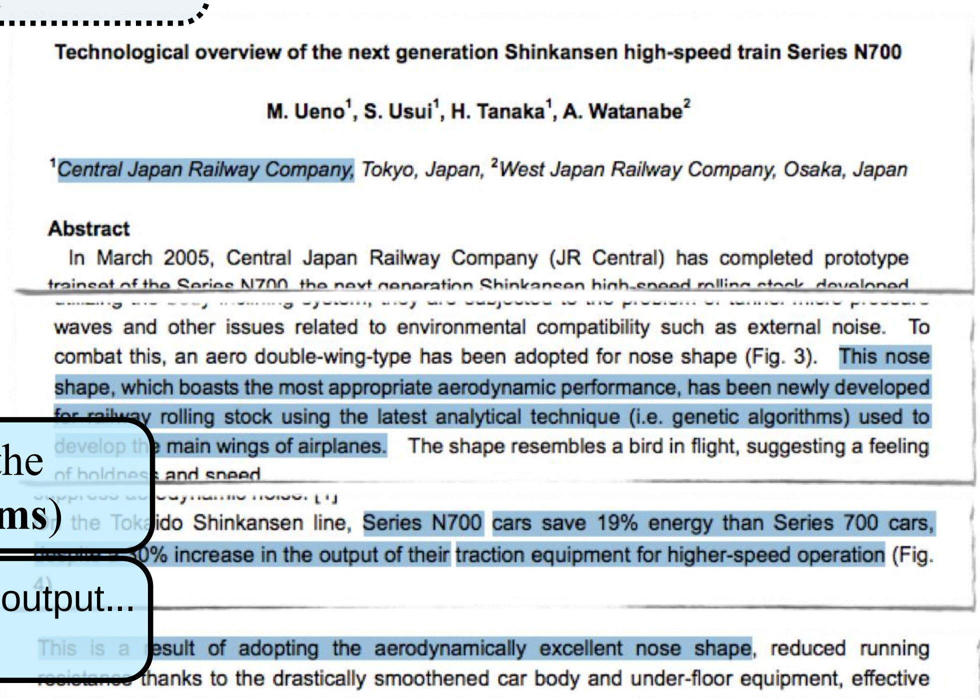
Application - high-speed train head design

Series 700





Series N700



Technological overview of the next generation Shinkansen high-speed train Series N700

M. Ueno[1], S. Usui[1], H. Tanaka[1], A. Watanabe[2]

[1]*Central Japan Railway Company, Tokyo, Japan,* [2]*West Japan Railway Company, Osaka, Japan*

Abstract

In March 2005, Central Japan Railway Company (JR Central) has completed prototype trainset of the Series N700, the next generation Shinkansen high-speed rolling stock, developed

waves and other issues related to environmental compatibility such as external noise. To combat this, an aero double-wing-type has been adopted for nose shape (Fig. 3). This nose shape, which boasts the most appropriate aerodynamic performance, has been newly developed for railway rolling stock using the latest analytical technique (i.e. genetic algorithms) used to develop the main wings of airplanes. The shape resembles a bird in flight, suggesting a feeling of boldness and speed.

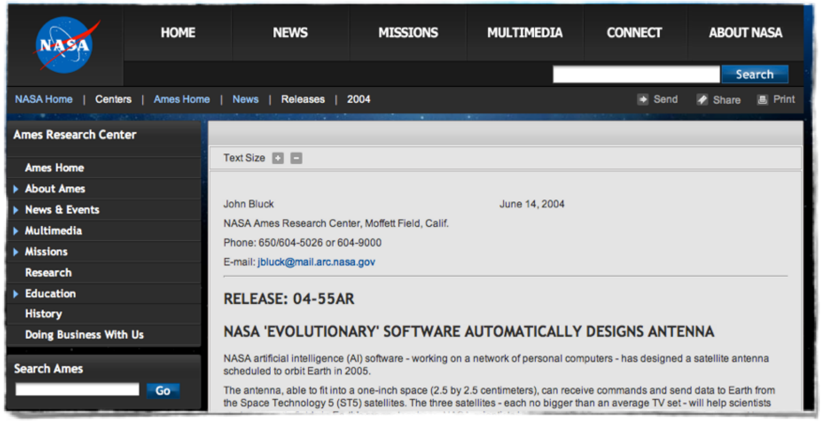
... the Tokaido Shinkansen line, Series N700 cars save 19% energy than Series 700 cars, despite a 30% increase in the output of their traction equipment for higher-speed operation (Fig. 4).

This is a result of adopting the aerodynamically excellent nose shape, reduced running resistance thanks to the drastically smoothened car body and under-floor equipment, effective

this nose ... has been newly developed ... using the latest analytical technique (i.e. **genetic algorithms**)

N700 cars save **19%** energy ... **30%** increase in the output... This is a result of adopting the ... nose shape

Application - antenna design



HOME NEWS MISSIONS MULTIMEDIA CONNECT ABOUT NASA

Search

NASA Home | Centers | Ames Home | News | Releases | 2004

Send Share Print

Ames Research Center

- Ames Home
- About Ames
- News & Events
- Multimedia
- Missions
- Research
- Education
- History
- Doing Business With Us

Search Ames

Go

Text Size

John Bluck June 14, 2004

NASA Ames Research Center, Moffett Field, Calif.

Phone: 650/604-5026 or 604-9000

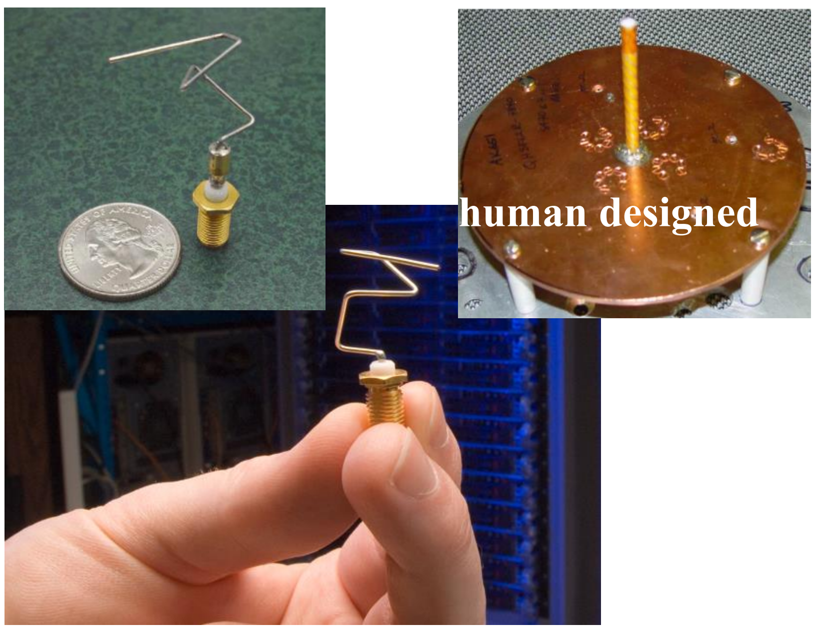
E-mail: jbluck@mail.arc.nasa.gov

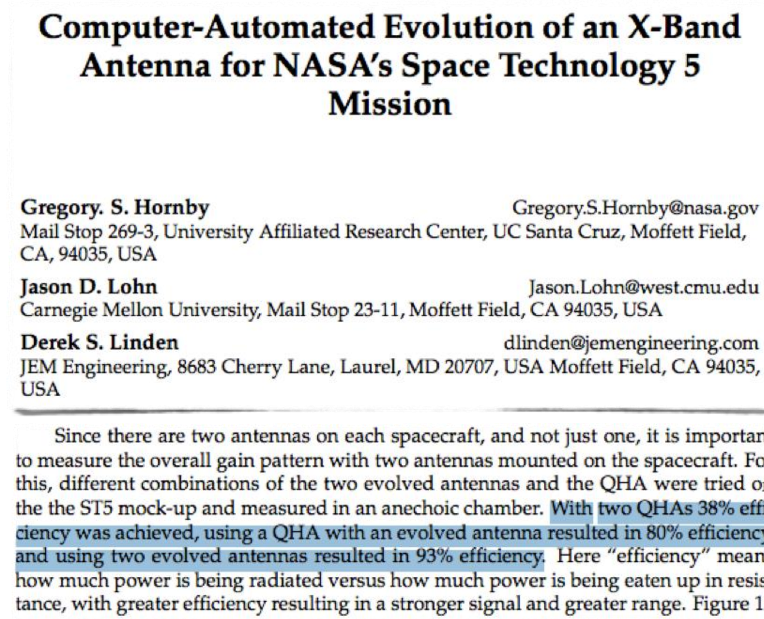
RELEASE: 04-55AR

NASA 'EVOLUTIONARY' SOFTWARE AUTOMATICALLY DESIGNS ANTENNA

NASA artificial intelligence (AI) software - working on a network of personal computers - has designed a satellite antenna scheduled to orbit Earth in 2005.

The antenna, able to fit into a one-inch space (2.5 by 2.5 centimeters), can receive commands and send data to Earth from the Space Technology 5 (ST5) satellites. The three satellites - each no bigger than an average TV set - will help scientists





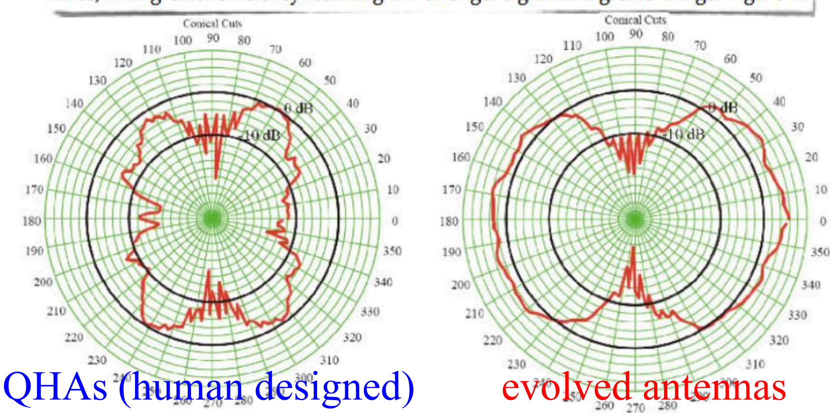
Computer-Automated Evolution of an X-Band Antenna for NASA's Space Technology 5 Mission

Gregory. S. Hornby Gregory.S.Hornby@nasa.gov
Mail Stop 269-3, University Affiliated Research Center, UC Santa Cruz, Moffett Field, CA, 94035, USA

Jason D. Lohn Jason.Lohn@west.cmu.edu
Carnegie Mellon University, Mail Stop 23-11, Moffett Field, CA 94035, USA

Derek S. Linden dlinden@jemengineering.com
JEM Engineering, 8683 Cherry Lane, Laurel, MD 20707, USA Moffett Field, CA 94035, USA

Since there are two antennas on each spacecraft, and not just one, it is important to measure the overall gain pattern with two antennas mounted on the spacecraft. For this, different combinations of the two evolved antennas and the QHA were tried on the the ST5 mock-up and measured in an anechoic chamber. With two QHAs 38% efficiency was achieved, using a QHA with an evolved antenna resulted in 80% efficiency, and using two evolved antennas resulted in 93% efficiency. Here "efficiency" means how much power is being radiated versus how much power is being eaten up in resistance, with greater efficiency resulting in a stronger signal and greater range. Figure 11



QHAs (human designed)
38% efficiency

evolved antennas
93% efficiency

Application - biological evolution

南京大學 | 新闻网

NANJING UNIVERSITY NJU NEWS 南京大学新闻中心主办

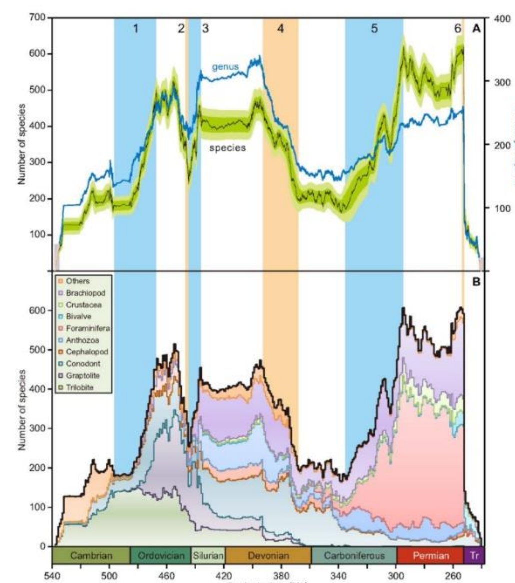
新闻关键字搜索 理论园地 南京大学报

首 页 综合新闻 专题新闻 理论园地 讲话与部署 南雍号 媒体传真 学术动态 影像南大 校园动态 学人视点 南大人

首页 - 综合新闻 2020-01-17 作者：地球科学与工程学院 来源：地球科学与工程学院

《Science》刊登南京大学地球科学与工程学院研究成果：大数据和超算揭秘古生代海洋生物多样性演化

北京时间1月17日，国际权威期刊《Science》以研究长文的形式在线发表了南京大学、中国科学院南京地质古生物所樊隽轩教授、沈树忠院士等的论文“A high-resolution summary of Cambrian to Early Triassic marine invertebrate biodiversity”。该研究利用古生物大数据、超算和遗传算法等全新的方法和手段，基于化石记录重现了生命演化历史，改变了当前对古生代海洋生物多样性演化的认知。



Safe evolutionary algorithms

EAs have yielded encouraging empirical outcomes, but lack theoretical guarantee

Reported results

Data set 1	✓
Data set 2	✓
Data set 3	✓
Data set 4	✓
Data set 5	✓

How about the performance on other data?

Data set 6	?
Data set 7	?
Data set 8	?
Data set 9	?
Data set 10	?

Not safe

Theoretical guarantee: for any instance of a given problem,

$$f(x) \geq \alpha \cdot \mathrm{OPT}$$

function value of the output solution ← $f(x)$

OPT → Optimal function value

Safe evolutionary algorithms

EAs have yielded encouraging empirical outcomes, but lack theoretical guarantee

Reported results

Data set 1	✓
Data set 2	✓
Data set 3	✓
Data set 4	✓
Data set 5	✓

How about the performance on other data?

Data set 6	?
Data set 7	?
Data set 8	?
Data set 9	?
Data set 10	?

Not safe

Can we design “safe” EAs, i.e., EAs with provable approximation guarantee?

Outline

Safe evolutionary optimization?

⇩

Develop running time analysis tools for EAs

Convergence analysis of EAs

Convergence analysis

$\lim_{t\to+\infty} P(\xi_t \in \mathcal{X}^*) = 1$?

An EA that

1. uses global operators
2. preserves the best solution

converges to the optimal solutions

[Rudolph, 1998]

⇨ the probability of finding optimum in each step

$\forall x: P(\xi_{t+1} \in \mathcal{X}^* \mid \xi_t = x) > 0$

⇩

$\lim_{t\to+\infty} P(\xi_t \in \mathcal{X}^*) = 1$

But life is limited! How fast does it converge?

Running time analysis of EAs

Convergence analysis

$\lim_{t\to+\infty} P(\xi_t \in \mathcal{X}^*) = 1$?

Running time analysis

$\tau = \min\{t \geq 0 \mid \xi_t \in \mathcal{X}^*\}$

The leading theoretical aspect

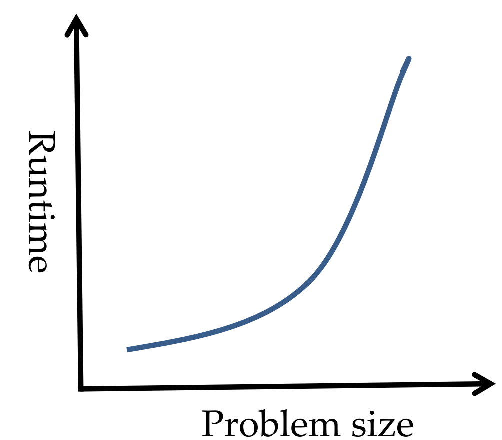
[Auger & Doerr, 2011; Neumann & Witt, 2012]

The number of iterations until finding an optimal or approximate solution for the first time

Running time complexity

- Usually grows with the problem size and expressed in asymptotic notations

  e.g., (1+1)-EA solving LeadingOnes: $O(n^2)$



Running time analysis tools of EAs

- Analyses starting from scratch are quite difficult
- We need general running time analysis tools

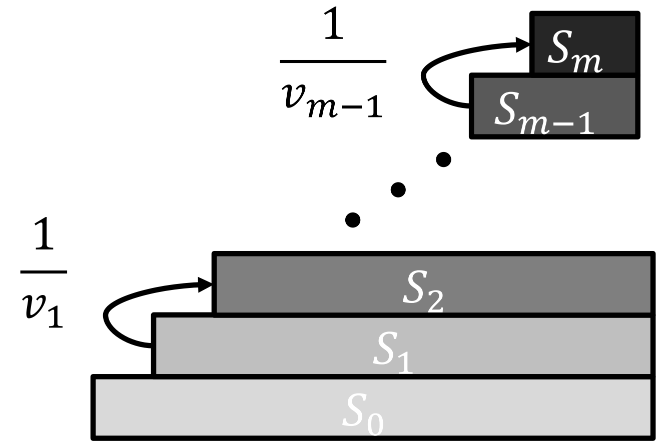
Fitness level method [Droste et al., TCS'02]

➢ Jumping probability lower bound:

$$P(\xi_{t+1} \in \cup_{j=i+1}^{m} S_j | \xi_t \in S_i) \geq v_i$$

⇩

➢ Running time upper bound: $\sum_{i=j}^{m-1} 1/v_i$



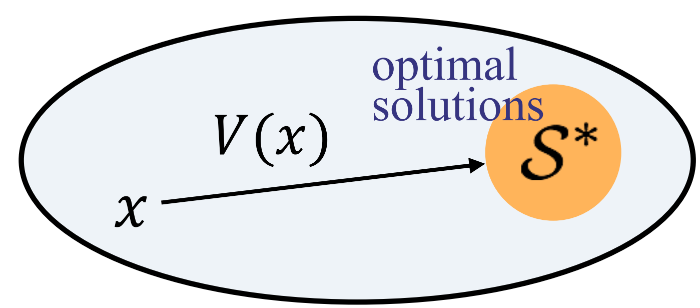
Drift analysis [He & Yao, AIJ'01]

➢ Expected drift in one step:

$$\mathrm{E}[\, V(\xi_t) - V(\xi_{t+1}) \mid \xi_t] \geq c$$

⇩

➢ Running time upper bound: $V(x)/c$



Switch analysis

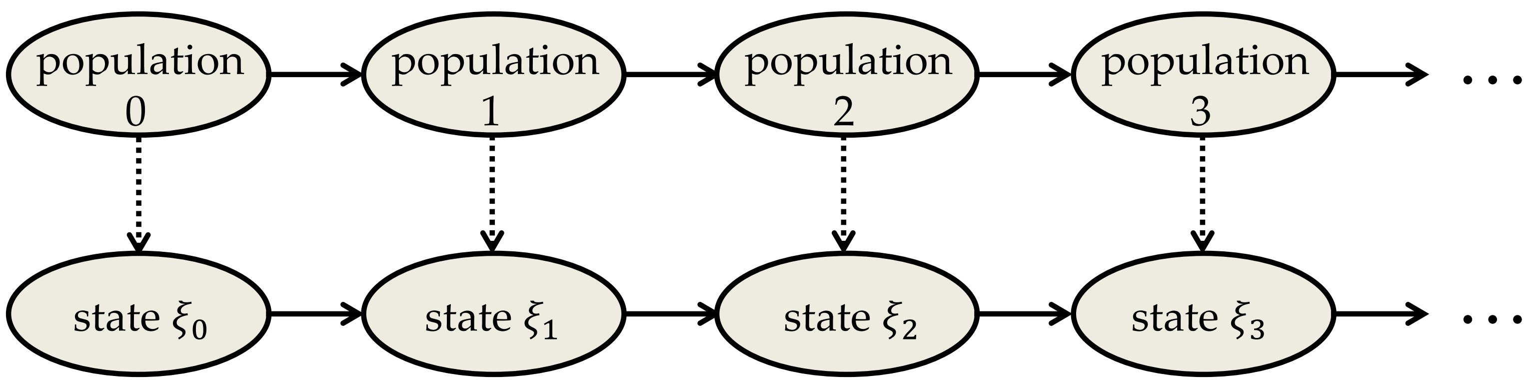
Switch analysis [Yu, Qian & Zhou, TEC'15]

- One step difference with a reference evolutionary process:

$$\forall t: \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \pi_t(x) P(\xi_{t+1} \in \phi^{-1}(y) | \xi_t = x) \mathrm{E}[\tau' | \xi'_0 = y]$$
$$- \sum_{u, y \in \mathcal{Y}} \pi_t^{\phi}(u) P(\xi'_1 \in y | \xi'_0 = u) \mathrm{E}[\tau' | \xi'_1 = y] \leq \rho_t$$

- Running time upper bound: $\mathrm{E}[\tau | \xi_0 \sim \pi_0] \leq \mathrm{E}[\tau' | \xi'_0 \sim \pi_0^{\phi}] + \sum_{t=0}^{+\infty} \rho_t$

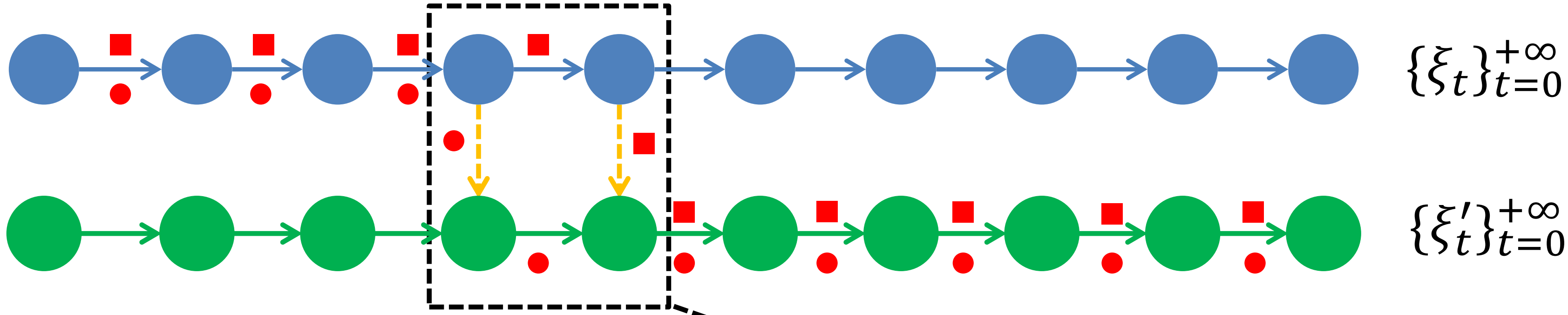
Main idea:



Switch analysis

Main idea [Yu, Qian and Zhou, TEC'15]:

Given EA on the given problem



Reference chain

examine the different behaviors at each step

The expected running time of $\{\xi_t\}_{t=0}^{+\infty}$:

$$\mathrm{E}[\tau] \le (\ge)\ \mathrm{E}[\tau'] + \sum_{t=0}^{+\infty} \rho_t$$

The expected running time of $\{\xi'_t\}_{t=0}^{+\infty}$, easy to analyze

Ability of switch analysis

Fitness level method [Droste et al., TCS'02]

- Jumping probability lower bound:

$$P(\xi_{t+1} \in \cup_{j=i+1}^{m} S_j | \xi_t \in S_i) \geq v_i$$

⇩

- Running time upper bound: $\sum_{i=j}^{m-1} 1/v_i$

Drift analysis [He & Yao, AIJ'01]

- Expected drift in one step:

$$\mathrm{E}[\, V(\xi_t) - V(\xi_{t+1}) \mid \xi_t] \geq c$$

⇩

- Running time upper bound: $V(x)/c$

} Reducible to switch analysis

Switch analysis can derive at least the same tight bound while requiring no more information

Application of switch analysis

Application [Bian, Qian and Tang, IJCAI'18]:

GSEMO	Problem	Previous result	Our result
Bi-objective	LOTZ	$O(n^3)$ [Giel, CEC'03]	$\le 6n^3$
	COCZ	$O(n^2 \log n)$ [Qian et al., AIJ'13]	$\le 3n^2 \log n$
Many-objective	mCOCZ	$O(n^{m+1})$ [Laumanns et al., TEC'04]	$O(n^m)$ for $m > 4$, $O(n^3 \log n)$ for $m = 4$
Approximate analysis	WOMM	—	$1/n$-approximation: $O(n^2(\log_l n + \log_l(w_n/w_1)))$

→ gives the leading constants

→ is asymptotically tighter than

L. Thiele

Member of Academia Europaea

Switch analysis is general and powerful

Outline

Safe evolutionary optimization?

Develop running time analysis tools for EAs

Disclose theoretical properties of EAs
for constrained and noisy optimization

How to deal with constraints when EAs are used for constrained optimization?

The optimization problems in real-world applications often come with constraints

Constrained optimization

General formulation:

$$\min_{x \in \mathcal{X}} \; f(x) \quad \text{(objective function)}$$

$$s.t. \quad g_i(x) = 0, \quad 1 \le i \le q; \quad \text{(equality constraints)}$$

$$h_i(x) \le 0, \quad q + 1 \le i \le m \quad \text{(inequality constraints)}$$

The goal: find a feasible solution minimizing the objective f

a feasible solution: satisfies all constraints

Constraint handling strategies

Two common constraint handling strategies:

- Penalty function [Hadj-Alouane & Bean, OR'97]
  - transform the original constrained optimization problem into an unconstrained optimization problem
- Multi-objective reformulation [Coello Coello, 2002; Cai & Wang, TEC'06]
  - transform the original constrained optimization problem into a bi-objective optimization problem

Penalty function

Main idea [Hadj-Alouane & Bean, OR'97]

1. transform the original constrained optimization problem into an unconstrained optimization problem

constrained

$$\begin{aligned} &min \quad f(x) \\ &s.t. \quad g_i(x) = 0, \quad 1 \le i \le q; \\ &\qquad\ \ h_i(x) \le 0, \quad q+1 \le i \le m \end{aligned}$$

unconstrained

$$min \quad f(x) + \lambda \sum_{i=1}^{m} f_i(x)$$

constraint violation degree

$$f_i(x) = \begin{cases} |g_i(x)| & 1 \le i \le q \\ \max\{0, h_i(x)\} & q+1 \le i \le m \end{cases}$$

Penalty function

Main idea [Hadj-Alouane & Bean, OR'97]

1. transform the original constrained optimization problem into an unconstrained optimization problem

constrained

$$\begin{aligned} &min \quad f(x) \\ &s.t. \quad g_i(x) = 0, \quad 1 \le i \le q; \\ &\qquad\; h_i(x) \le 0, \quad q+1 \le i \le m \end{aligned}$$

unconstrained

$$min \quad f(x) + \lambda \sum_{i=1}^{m} f_i(x)$$

2. employ a single-objective EA to solve the transformed problem

Multi-objective reformulation

Main idea [Coello Coello, 2002; Cai & Wang, TEC'06]

1. transform the original constrained optimization problem into a bi-objective optimization problem

constrained

$$\begin{aligned} & min \;\; f(x) \\ & s.t. \;\; g_i(x) = 0, \quad 1 \le i \le q; \\ & \qquad h_i(x) \le 0, \quad q+1 \le i \le m \end{aligned}$$

bi-objective

$$min \;\; \left(f(x),\; \sum_{i=1}^{m} f_i(x)\right)$$

constraint violation degree

$$f_i(x) = \begin{cases} |g_i(x)| & 1 \le i \le q \\ \max\{0, h_i(x)\} & q+1 \le i \le m \end{cases}$$

Multi-objective optimization

The task: optimize multiple objectives simultaneously

$$min_{x \in \mathcal{X}} \ (f_1(x), f_2(x), \ldots, f_m(x))$$

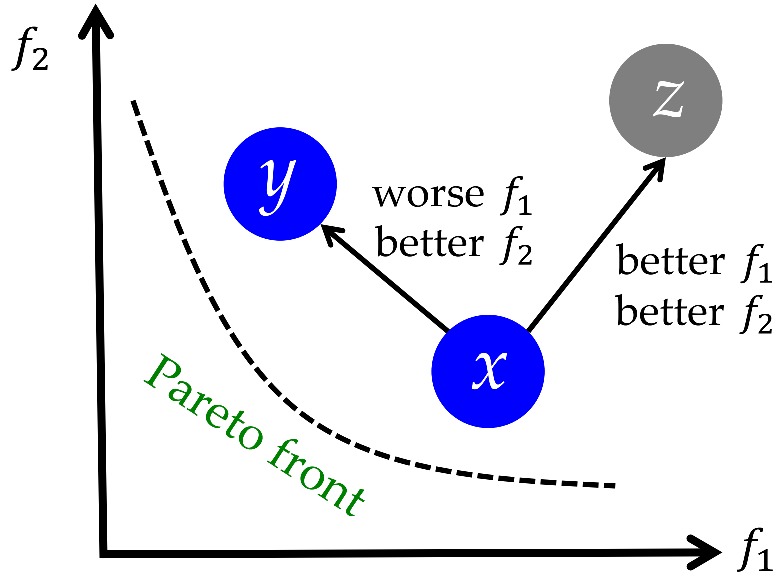
Example: bi-objective minimization

x dominates z :

$$f_1(x) < f_1(z) \wedge f_2(x) < f_2(z)$$

x is incomparable with y :

$$f_1(x) > f_1(y) \wedge f_2(x) < f_2(y)$$



Multi-objective reformulation

Main idea [Coello Coello, 2002; Cai & Wang, TEC'06]

1. transform the original constrained optimization problem into a bi-objective optimization problem

constrained

$$\begin{aligned} &min \quad f(x) \\ &s.t. \quad g_i(x) = 0, \quad 1 \le i \le q; \\ &\qquad\; h_i(x) \le 0, \quad q+1 \le i \le m \end{aligned}$$

bi-objective

$$min \quad \left(f(x),\ \sum_{i=1}^{m} f_i(x)\right)$$

2. employ a multi-objective EA to solve the transformed problem

3. output the feasible solution from the generated non-dominated solution set

constraint violation degree = 0

Constraint handling strategies

Two common constraint handling strategies:

- Penalty function [Hadj-Alouane & Bean, OR'97]
- Multi-objective reformulation [Coello Coello, 2002; Cai & Wang, TEC'06]

Previous empirical studies have shown the superior performance of multi-objective reformulation

It is not yet clear whether multi-objective reformulation can be better in theory

Problems

- Minimum matroid optimization (P-solvable) [Edmonds, MP'71]

  e.g., minimum spanning tree, maximum bipartite matching

Definition 1. Given a matroid (U, S), a rank function $r: 2^U \to \mathbb{N}$ and a weight function $w: U \to \mathbb{N}$, the problem is formulated as

$$min_{\boldsymbol{x} \in \{0,1\}^n} \sum_{i=1}^n w_i x_i \quad s.t. \quad r(\boldsymbol{x}) = r(U)$$

- Minimum cost coverage (NP-hard) [Wolsey, Combinatorica'82]

  e.g., minimum set cover, submodular set cover

Definition 2. Given a monotone submodular function $f: 2^U \to \mathbb{R}$, some value $q \leq f(U)$ and a weight function $w: U \to \mathbb{N}$, the problem is formulated as

$$min_{\boldsymbol{x} \in \{0,1\}^n} \sum_{i=1}^n w_i x_i \quad s.t. \quad f(\boldsymbol{x}) \geq q$$

Theoretical analysis

Penalty function vs. Multi-objective reformulation

[Qian, Yu and Zhou, IJCAI'15]

- Minimum matroid optimization (P-solvable): obtaining an optimal solution

matroid rank — problem size — maximum weight

Penalty function: $\Omega(r^2 n(\log n + \log w_{max}))$

Multi-objective reformulation: $O(rn(\log n + \log w_{max} + r))$

The running time reduces by a factor $min\{\log n + \log w_{max}, r\}$

Theoretical analysis

- Minimum matroid optimization (P-solvable): obtaining an optimal solution

Penalty function: $\Omega(r^2 n(\log n + \log w_{max}))$

Multi-objective reformulation: $O(rn(\log n + \log w_{max} + r))$

The running time reduces by a factor $min\{\log n + \log w_{max}, r\}$

- Minimum cost coverage (NP-hard): obtaining a H_q-approximate solution

Penalty function: exponential w.r.t. $n, q, \log w_{max}$

Multi-objective reformulation: $O(qn(\log n + \log w_{max} + q))$

The running time reduces exponentially

polynomial

Theoretical analysis

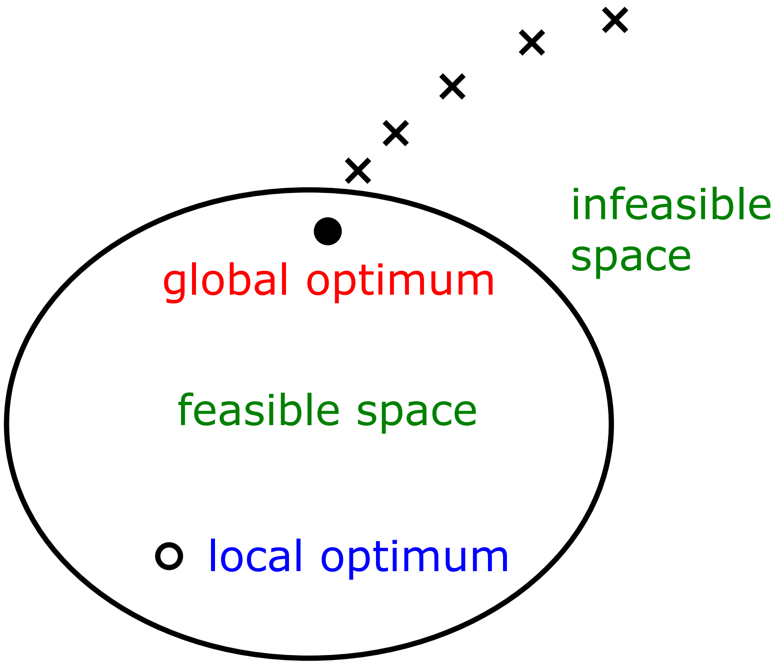
Findings from the analysis:

Penalty function

- the penalty prefers feasible solutions
- get trapped in the local optimum, which is far from the global optimum

Multi-objective reformulation

- the constraint violation objective allows infeasible solutions
- follow a short path from infeasible to feasible to find good solutions



General problem classes

[Qian et al., AIJ'19]

- Constrained submodular approximately monotone maximization

  Multi-objective reformulation: $$f(x) \geq \left(1 - \frac{1}{e}\right) \cdot (\mathrm{OPT} - k\epsilon)$$

- Constrained monotone approximately submodular maximization

  Multi-objective reformulation: $$f(x) \geq \frac{1}{1 + \frac{2k\epsilon}{1-\epsilon}} \left(1 - \frac{1}{e}\left(\frac{1-\epsilon}{1+\epsilon}\right)^k\right) \cdot \mathrm{OPT}$$

Achieve the best known polynomial-time approximation guarantee [Krause et al., JMLR'08] [Horel & Singer, NIPS'16]

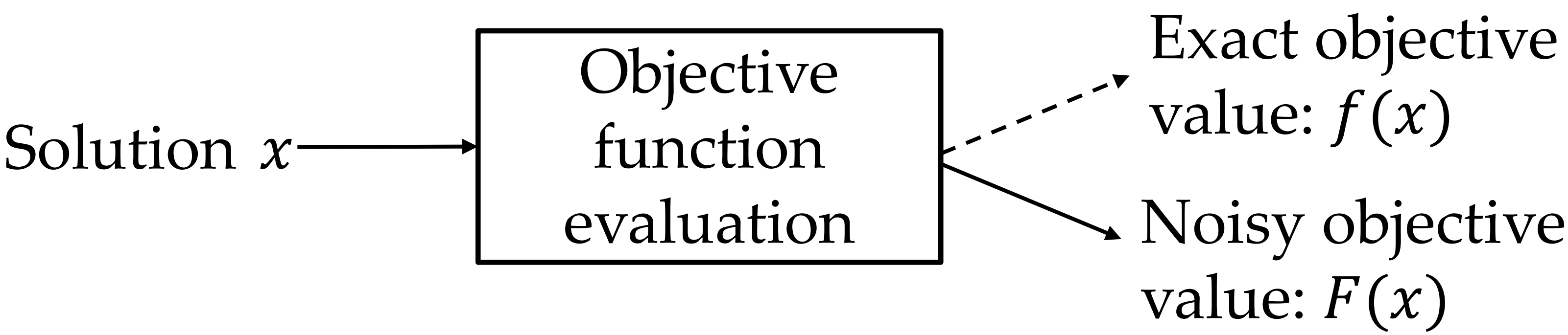
How to improve the robustness when EAs are used for noisy optimization?

The optimization problems in real-world applications often come with noise

Noisy optimization

The objective evaluation is often disturbed by noise

e.g., a prediction model is evaluated only on a limited amount of data



e.g., $f(x) + N(0, \sigma^2)$

The influence of noise

It was believed that noise makes evolutionary optimization harder

many noise handling strategies have been proposed
[Jin & Branke, TEC'05; Goh & Tan, TEC'07]

Some empirical observations have shown that noise can have a positive impact on the performance of local search
[Selman et al., AAAI'94; Hoos & Stutzle, JAR'00]

Can noise make evolutionary optimization easier?

Theoretical analysis

A sufficient condition: noise is helpful [Qian, Yu and Zhou, ECJ'18]

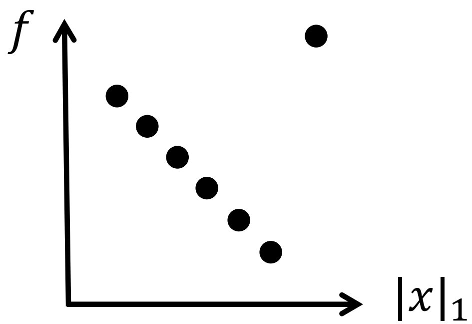
Theorem 1. *For an EA $\mathcal{A}$ optimizing a problem f, which can be modeled by a deceptive Markov chain, if*

$$\forall x \notin \mathcal{X}_0 : P_\xi^t(x, \mathcal{X}_0) = \sum\nolimits_{x' \cap \mathcal{S}^* \neq \emptyset} P_{var}(x, x'), \tag{6}$$

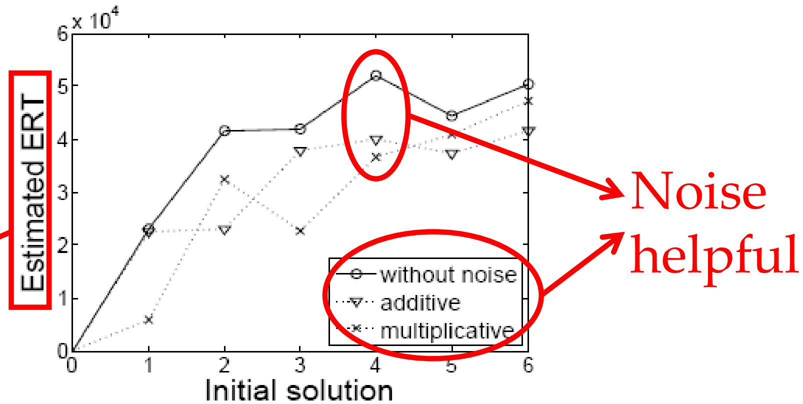
then noise makes f easier for $\mathcal{A}$.

Intuitively, noise can bring some randomness to help the EA escape from local optima

Example: (1+n)-EA on the Trap problem



Running time



Noise handling strategies

Noise is harmful in most cases

Two commonly used noise handling strategies:

- Re-evaluation [Arnold & Beyer, TEC'02; Jin & Branke, TEC'05]
  - every time we access the fitness of a solution by evaluation

smooth noise

- Threshold selection [Markon et al., CEC'01; Bartz-Beielstein & Markon, CEC'02]
  - an offspring solution is accepted only if its fitness is larger than that of the parent solution by at least a threshold τ

reduce the risk of accepting a bad solution due to noise

Theoretical analysis

the range of noise level such that the running time is polynomial

Example:

(1+1)-EA

OneMax

one-bit noise

Noise handling strategies	PNT
single-evaluation	$[0, 1 - \frac{1}{\Theta(poly(n))}]$
single-evaluation & $\tau > 0$	$[0, 0]$
re-evaluation	$[0, \Theta(\frac{\log n}{n})]$ (Droste, 2004)
re-evaluation & $\tau = 1$	$[0, 1]$
re-evaluation & $\tau = 2$	$[\frac{1}{\Theta(poly(n))}, 1 - \frac{1}{\Theta(poly(n))}]$
re-evaluation & $\tau > 2$	$\emptyset$

combining re-evaluation with proper threshold selection is better [Qian, Yu and Zhou, ECJ'18]

Outline

Safe evolutionary optimization?

Develop running time analysis tools for EAs

Disclose theoretical properties of EAs
for constrained and noisy optimization

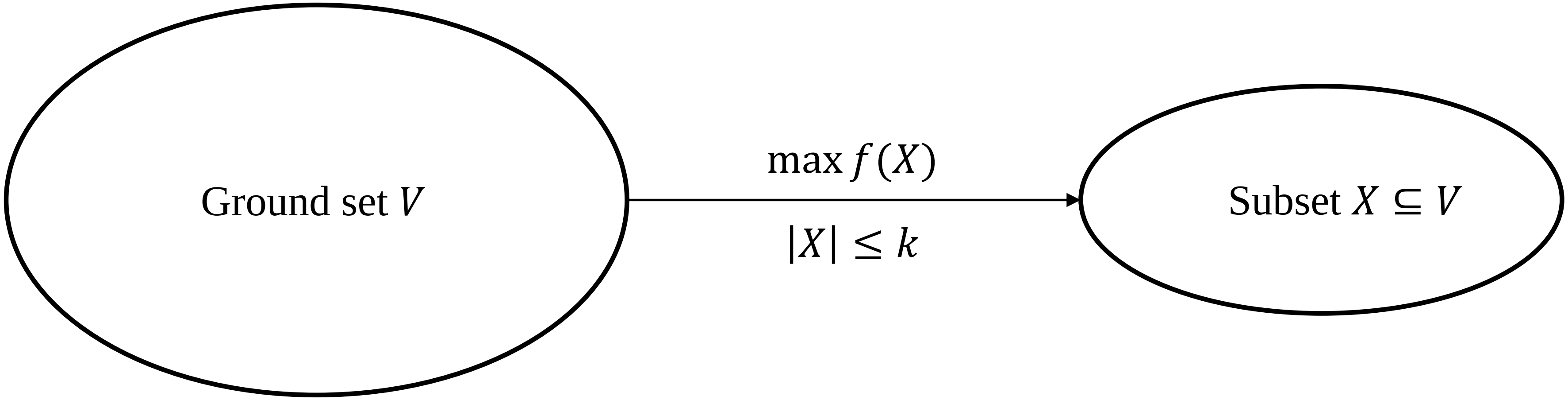
Propose EAs with good polynomial-time
approximation guarantees for (noisy) subset selection

Subset selection

Subset selection is to select a subset of size k from a total set of n items for optimizing some objective function

Formally stated: given all items $V = \{v_1, \ldots, v_n\}$, an objective function $f: 2^V \to \mathrm{R}$ and a budget k, to find a subset $X \subseteq V$ such that

$$max_{X \subseteq V} \quad f(X) \quad s.t. \quad |X| \le k.$$



Application - sparse regression

Sparse regression [Tropp, TIT'04] : select a few observation variables to best approximate the predictor variable by linear regression

observation variables

predictor variable z

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.03	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.93	0.56

a subset X of observation variables

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.03	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.93	0.56

Item v_i: an observation variable

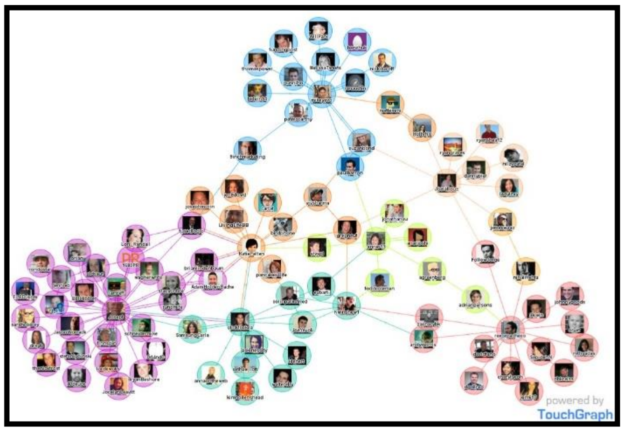
variance

mean squared error

Objective f: squared multiple correlation $R^2_{z,X} = \dfrac{\mathrm{Var}(z) - \mathrm{MSE}_{z,X}}{\mathrm{Var}(z)}$

Application - influence maximization

Influence maximization [Kempe et al., KDD’03] : select a subset of users from a social network to maximize its influence spread

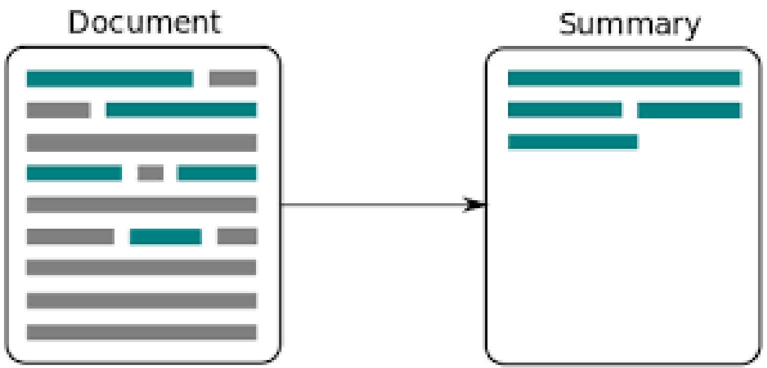


Item v_i: a social network user

Objective f: influence spread, measured by the expected number of social network users activated by diffusion

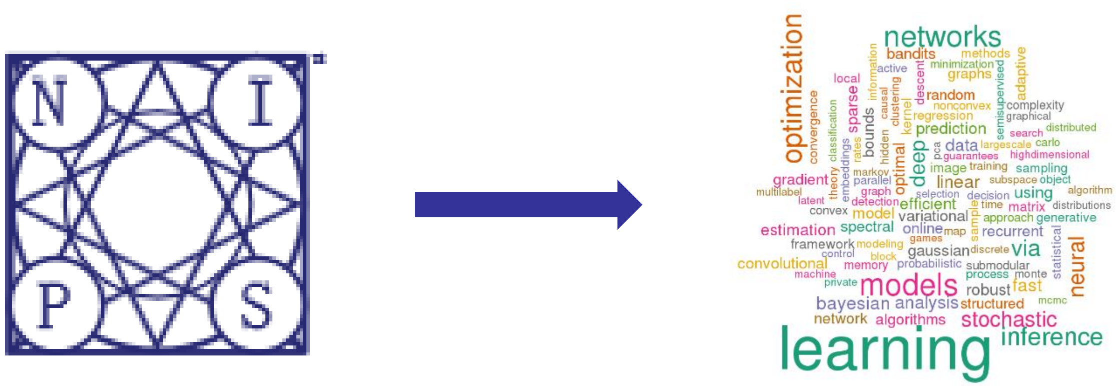
Application - document summarization

Document summarization [Lin & Bilmes, ACL'11]: select a few sentences to best summarize the documents



Item v_i: a sentence

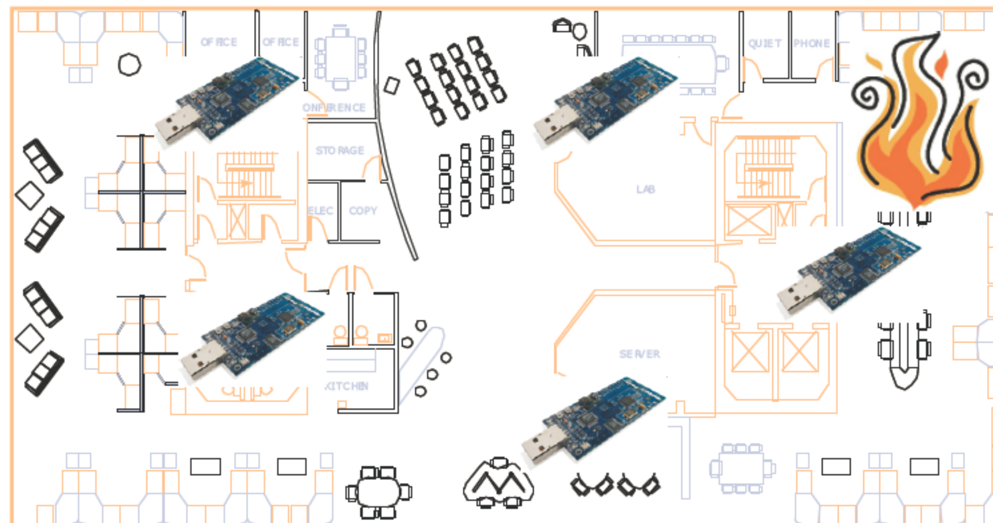
Objective f: summary quality



Application - sensor placement

Sensor placement [Krause & Guestrin, IJCAI'09 Tutorial] : select a few places to install sensors such that the information gathered is maximized

Water contamination detection



Fire detection

Item v_i: a place to install a sensor

Objective f: entropy

Subset selection

Sparse regression

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.08	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.98	0.56

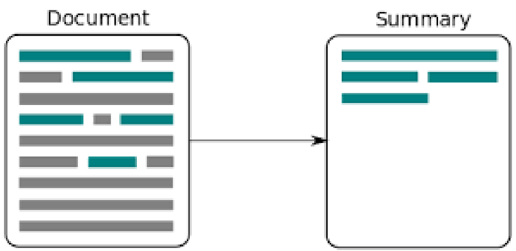
Machine learning

Data mining

Influence maximization

Subset selection

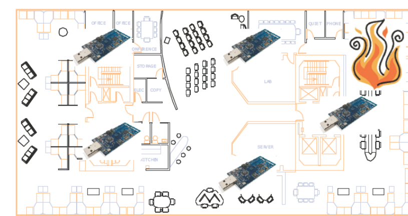
Document summarization

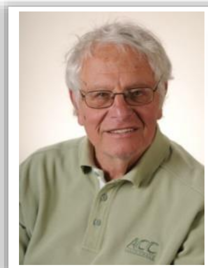


Natural language processing

Networks

Sensor placement





George Nemhauser

John Von Neumann Theory Prize

[Mathematical Programming 1978]

f : monotone and submodular

The greedy algorithm:

$(1-1/e)$-approximation

Best Paper/Test of Time Award:

[Kempe et al., KDD'03]

[Das & Kempe, ICML'11]

[Iyer & Bilmes, NIPS'13]

POSS algorithm

POSS algorithm [Qian, Yu and Zhou, NIPS'15]

Transformation:

$$max_{X \subseteq V} \; f(X) \quad s.t. \quad |X| \leq k$$ original

$$\Downarrow$$

$$min_{X \subseteq V} \; (-f(X), |X|)$$ bi-objective

Algorithm 1 POSS

Input: all variables $V = \{X_1, \ldots, X_n\}$, a given objective f and an integer parameter $k \in [1, n]$
Parameter: the number of iterations T
Output: a subset of V with at most k variables
Process:

```
Let s = {0}^n and P = {s}.
Let t = 0.
while t < T do
    Select s from P uniformly at random.
    Generate s' by flipping each bit of s with prob. 1/n.
    Evaluate f_1(s') and f_2(s').
    if ∄z ∈ P such that z ≺ s' then
        Q = {z ∈ P | s' ⪯ z}.
        P = (P \ Q) ∪ {s'}.
    end if
    t = t + 1.
end while
return arg min_{s ∈ P, |s| ≤ k} f_1(s)
```

Initialization: put the special solution $\{0\}^n$ into the population P

Reproduction: pick a solution $\boldsymbol{x}$ randomly from P, and flip each bit of $\boldsymbol{x}$ with prob. $1/n$ to generate a new solution

Evaluation & Updating: if the new solution is not dominated, put it into P and weed out bad solutions

Output: select the best feasible solution

Sparse regression

Sparse regression: given all observation variables $V=\{v_1,\ldots,v_n\}$, a predictor variable z and a budget k, to find a subset $X \subseteq V$ such that

$$max_{X\subseteq V} \quad R^2_{z,X} = \frac{\mathrm{Var}(z) - \mathrm{MSE}_{z,X}}{\mathrm{Var}(z)} \quad s.t. \quad |X| \le k$$

$\mathrm{Var}(z)$: variance of z

$\mathrm{MSE}_{z,X}$: mean squared error of predicting z by using observation variables in X

observation variables

predictor variable z

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.03	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.93	0.56

a subset X of observation variables

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.03	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.93	0.56

Experimental results

the size constraint: $k = 8$ **the number of iterations of POSS: $2ek^2n$**

exhaustive search — greedy algorithms — relaxation methods

Data set	OPT	POSS	FR	FoBa	OMP	RFE	MCP
housing	.7437±.0297	.7437±.0297	.7429±.0300•	.7423±.0301•	.7415±.0300•	.7388±.0304•	.7354±.0297•
eunite2001	.8484±.0132	.8482±.0132	.8348±.0143•	.8442±.0144•	.8349±.0150•	.8424±.0153•	.8320±.0150•
svmguide3	.2705±.0255	.2701±.0257	.2615±.0260•	.2601±.0279•	.2557±.0270•	.2136±.0325•	.2397±.0237•
ionosphere	.5995±.0326	.5990±.0329	.5920±.0352•	.5929±.0346•	.5921±.0353•	.5832±.0415•	.5740±.0348•
sonar	–	.5365±.0410	.5171±.0440•	.5138±.0432•	.5112±.0425•	.4321±.0636•	.4496±.0482•
triazines	–	.4301±.0603	.4150±.0592•	.4107±.0600•	.4073±.0591•	.3615±.0712•	.3793±.0584•
coil2000	–	.0627±.0076	.0624±.0076•	.0619±.0075•	.0619±.0075•	.0363±.0141•	.0570±.0075•
mushrooms	–	.9912±.0020	.9909±.0021•	.9909±.0022•	.9909±.0022•	.6813±.1294•	.8652±.0474•
clean1	–	.4368±.0300	.4169±.0299•	.4145±.0309•	.4132±.0315•	.1596±.0562•	.3563±.0364•
w5a	–	.3376±.0267	.3319±.0247•	.3341±.0258•	.3313±.0246•	.3342±.0276•	.2694±.0385•
gisette	–	.7265±.0098	.7001±.0116•	.6747±.0145•	.6731±.0134•	.5360±.0318•	.5709±.0123•
farm-ads	–	.4217±.0100	.4196±.0101•	.4170±.0113•	.4170±.0113•	–	.3771±.0110•
POSS: win/tie/loss		–	12/0/0	12/0/0	12/0/0	11/0/0	12/0/0

• denotes that POSS is significantly better by the t-test with confidence level 0.05

POSS is significantly better than all the compared state-of-the art algorithms on all data sets

Theoretical analysis

POSS can achieve the optimal polynomial-time approximation guarantee

Theorem 1. For subset selection with monotone objective functions, POSS using $E[T] \leq 2ek^2n$ finds a solution X with $|X| \leq k$ and

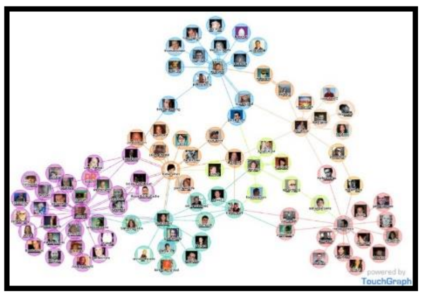
$$f(X) \geq (1 - e^{-\gamma}) \cdot OPT.$$

the optimal polynomial-time approximation guarantee for monotone f [Harshaw et al., ICML'19]

Noise

Previous analyses often assume that the exact value of the objective function can be accessed

However, in many applications of subset selection, only a noisy value of the objective function can be obtained



Influence maximization

The objective function: the expected number of nodes activated by propagating from X

noise

The average number of nodes activated by a limited number of independent diffusion processes [Kempe et al., KDD'03]

Noise

Previous analyses often assume that the exact value of the objective function can be accessed

However, in many applications of subset selection, only a noisy value of the objective function can be obtained

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.03	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.93	0.56

Sparse regression

	Corr.	Dis.	LR	···	···	AIC.	BIC	RF.
x1	0.28	0.46	1	···	···	0.22	0.63	1
x2	0.31	0.59	0.64	···	···	0.58	0.56	1
x3	0.11	0.02	0.53	···	···	0.43	0.01	1
x4	0.1	0.1	0.64	···	···	0.73	0.92	1
x5	0.02	0.15	0.33	···	···	0.56	0.36	0.78
x6	0.36	0.02	0.01	···	···	0.32	0.02	0.22
x7	0.2	0.2	0.21	···	···	0.21	0.02	0.11
x8	0.1	0.03	0.32	···	···	0.33	0.51	0.44
x9	0.32	0.1	0.2	···	···	0.06	0.66	0
x10	0.24	0	0.02	···	···	0.6	0.03	0.33
x11	0.12	0.45	0.44	···	···	0.64	0.45	1
x12	0.36	0.58	0.12	···	···	0.73	0.58	0.67
x13	0.2	0.02	0.24	···	···	0.34	0.02	0.89
x14	0.24	0.92	0.33	···	···	0.24	0.93	0.56

The objective function: the mean squared error of prediction by X

noise

How about the performance for noisy subset selection?

Noisy subset selection

Subset selection: given $V = \{v_1, \ldots, v_n\}$, an objective function $f: 2^V \to R$ and a budget k, to find a subset $X \subseteq V$ such that

$$max_{X \subseteq V} \quad f(X) \quad s.t. \quad |X| \leq k$$

exact objective value

noisy objective value

Noise

- Multiplicative: $(1-\epsilon) \cdot f(X) \leq F(X) \leq (1+\epsilon) \cdot f(X)$
- Additive: $f(X) - \epsilon \leq F(X) \leq f(X) + \epsilon$

Applications: influence maximization, sparse regression

maximizing information gain in graphical models [Chen et al., COLT'15]

crowdsourced image collection summarization [Singla et al., AAAI'16]

Theoretical analysis

Approximation guarantee of POSS

noise strength

$\varepsilon \leq 1/k$ for a constant approximation ratio

Under multiplicative noise:

$$f(X) \geq \frac{1}{1+\frac{2\epsilon k}{(1-\epsilon)\gamma}}\left(1-\left(\frac{1-\epsilon}{1+\epsilon}\right)^k\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT$$

Under additive noise:

$$f(X) \geq \left(1-\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT - \left(\frac{2k}{\gamma}-\frac{2k}{\gamma}e^{-\gamma}\right)\epsilon$$

Noiseless approximation guarantee [Qian, Yu and Zhou, NIPS'15]

$$f(X) \geq \left(1-\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT \geq (1-e^{-\gamma})\cdot OPT$$

a constant approximation ratio

The performance degrades largely in noisy environments

PONSS algorithm

Threshold selection has theoretically been shown to be robust against noise [Qian, Yu and Zhou, ECJ'18]

$$f(X) \geq f(Y) \longrightarrow f(X) \geq f(Y) + \theta$$

POSS

"better"

$$X \preccurlyeq Y \Leftrightarrow \begin{cases} f(X) \geq f(Y) \\ |X| \leq |Y| \end{cases}$$

Conservative comparison

PONSS [Qian et al., NIPS'17]

Multiplicative:

$$X \preccurlyeq Y \Leftrightarrow \begin{cases} f(X) \geq \dfrac{1+\theta}{1-\theta} f(Y) \\ |X| \leq |Y| \end{cases}$$

Additive:

$$X \preccurlyeq Y \Leftrightarrow \begin{cases} f(X) \geq f(Y) + 2\theta \\ |X| \leq |Y| \end{cases}$$

Theoretical analysis

Under multiplicative noise:

PONSS $f(X) \geq \frac{1-\epsilon}{1+\epsilon}\left(1-\left(1-\frac{\gamma}{k}\right)^k\right) \cdot OPT$ Significantly better

POSS $f(X) \geq \frac{1}{1+\frac{2\epsilon k}{(1-\epsilon)\gamma}}\left(1-\left(\frac{1-\epsilon}{1+\epsilon}\right)^k\left(1-\frac{\gamma}{k}\right)^k\right) \cdot OPT$

When $\gamma = 1$ (submodular), ϵ is a constant

PONSS a constant approximation ratio

POSS $\Theta(1/k)$ approximation ratio

Theoretical analysis

Under multiplicative noise:

PONSS $$f(X) \geq \frac{1-\epsilon}{1+\epsilon}\left(1-\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT$$ Significantly better

POSS $$f(X) \geq \frac{1}{1+\frac{2\epsilon k}{(1-\epsilon)\gamma}}\left(1-\left(\frac{1-\epsilon}{1+\epsilon}\right)^k\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT$$

Under additive noise:

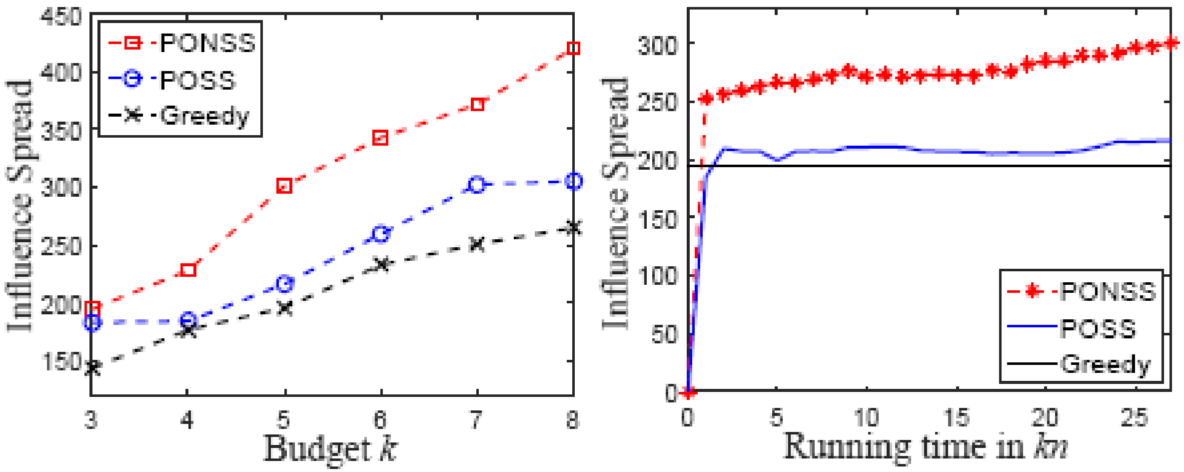
PONSS $$f(X) \geq \left(1-\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT - 2\epsilon$$ better

POSS $$f(X) \geq \left(1-\left(1-\frac{\gamma}{k}\right)^k\right)\cdot OPT - \left(\frac{2k}{\gamma}-\frac{2k}{\gamma}e^{-\gamma}\right)\epsilon$$

Experimental results - influence maximization

PONSS (red line) vs. POSS (blue line) vs. Greedy (black line):

- Noisy evaluation: the average of 10 independent Monte Carlo simulations
- The output solution: the average of 10,000 independent Monte Carlo simulations

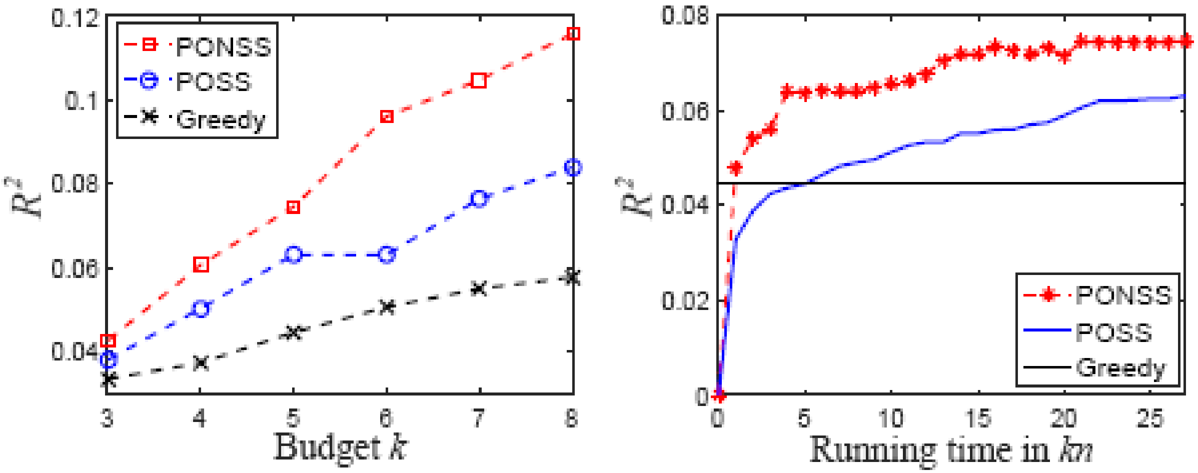


(b) *Weibo* (10,000 #nodes, 162,371 #edges)

Experimental results - sparse regression

PONSS (red line) vs. POSS (blue line) vs. Greedy (black line):

- Noisy evaluation: a random sample of 1,000 instances
- The output solution: the whole data set



(a) *protein* (24,387 #inst, 357 #feat)

Conclusion

Safe evolutionary optimization?

Develop running time analysis tools for EAs

Disclose theoretical properties of EAs
for constrained and noisy optimization

Propose EAs with good polynomial-time
approximation guarantees for (noisy) subset selection

Yes

Collaborators:

Nanjing University:

Chao Bian

Yang Yu

Zhi-Hua Zhou

SUSTech:

Ke Tang

Xin Yao

For details

- Y. Yu, C. Qian, Z.-H. Zhou. Switch analysis for running time analysis of evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 2015, 19(6): 777-792.
- C. Qian, Y. Yu, Z.-H. Zhou. On constrained Boolean Pareto optimization. In: *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI'15)*, Buenos Aires, Argentina, 2015.
- C. Qian, Y. Yu, Z.-H. Zhou. Subset selection by Pareto optimization. In: *Advances in Neural Information Processing Systems 28 (NIPS'15)*, Montreal, Canada, 2015.
- C. Qian, J.-C. Shi, Y. Yu, K. Tang, Z.-H. Zhou. Subset selection under noise. In: *Advances in Neural Information Processing Systems 30 (NIPS'17)*, Long Beach, CA, 2017.
- C. Bian, C. Qian, K. Tang. A general approach to running time analysis of multi-objective evolutionary algorithms. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI'18)*, Stockholm, Sweden, 2018.
- C. Qian, Y. Yu, Z.-H. Zhou. Analyzing evolutionary optimization in noisy environments. *Evolutionary Computation*, 2018, 26(1): 1-41.
- C. Qian, Y. Yu, K. Tang, X. Yao, Z.-H. Zhou. Maximizing submodular or monotone approximately submodular functions by multi-objective evolutionary algorithms. *Artificial Intelligence*, 2019, 275: 279-294.

Codes available at http://www.lamda.nju.edu.cn/qianc/

THANK YOU !