

پردازش تکاملی

تقویت روش های تکاملی Motivations

دانشگاه صنعتی مالک اشتر

مجتمع دانشگاهی فن آوری اطلاعات و امنیت

زمستان ۱۳۹۲

Motivation 1

- الگوریتم های تکاملی روشهای پارامتریک بکار می برد
- اپراتور جهش و نرخ جهش
- اپراتور ترکیب و نرخ ترکیب
- مکانیزم انتخاب و سخت بودن آن (رده بندی کردن مسابقات)
- اندازه جمعیت
- مقادیر پارامتر خوب در کارایی بالا موثر است.
- سوال ۱: چطور برای پارامتر مقادیر خوب بیابیم؟

Motivation 2

- پارامترهای تکاملی جامد هستند (در طی اجرا ثابت هستند) اما

الگوریتم تکاملی دینامیکی است، فرآیند آن تطبیقی است
بنابراین

مقادیر پارامترهای بهینه ممکن است در طی اجرا تغییر
یابند.

سوال ۲: چطور مقادیر پارامترها تغییر می نمایند؟

Parameter tuning

- **تطبیق پارامترها:** روش سنتی تست و مقایسه مقادیر متفاوت قبل از اجرای «واقعی» مشکلات
- خطاهای کاربران در تنظیمات میتواند منبع خطا باشد یا کارایی نیمه بهینه شود.
- هزینه زمان بالا.
- انفعال پارامترها: جامعیت جستجو عملی نیست.
- مقادیر خوب ممکن است در زمان اجرا ناصحیح شوند.

Parameter control

- پارامتر کنترل: تنظیم مقادیر on-line, در طی

اجرای واقعی

- $P=P(t)$ زمانبندی تغییرات- زمان از قبل تعیین شده

- استفاده از بازخورد فرآیند جستجو

- کدگذاری پارامترها در کروموزوم و تکیه بر انتخاب

طبیعی

مشکلات:

- یافتن P بهینه مشکل است و یافتن $P(t)$ بهینه سخت تر

- بازهم مکانیزم تعریف-کاربر، چطور بهینه سازی کنیم؟

مثال

حل مثال:

$$\begin{aligned}
 & \min f(x_1, \dots, x_n) - \\
 & \text{for } i = 1, \dots, n \quad L_i \leq x_i \leq U_i - \\
 & \text{کران ها} \\
 & \text{for } i = 1, \dots, q \quad g_i(x) \leq 0 - \\
 & \text{نا برابری محدودیت ها} \\
 & \text{for } i = q+1, \dots, m \quad h_i(x) = 0 - \\
 & \text{برابری محدودیت ها}
 \end{aligned}$$

الگوریتم:

- (x_1, \dots, x_n) نمایش مقادیر واقعی
- محاسبه میانگین تقاطع
- $x'_i = x_i + N(0, \sigma)$ جهش گوسی
- انحراف معیار σ مقدار گام جهش نامیده میشود

تغییر اندازه جهش : option1

• تغییر ثابت σ بوسیله تابع $\sigma(t)$

$$\sigma(t) = 1 - 0.9 \times \frac{t}{T}$$

$0 \leq t \leq T$ تعداد نسل جاری

ویژگی ها :

تغییر σ مستقل از فرآیند جستجو است

قدرت کنترل کاربر بوسیله σ در فرمول بالا

σ کاملاً قابل پیش بینی است

یک σ داده شده روی همه موجودیت های جمعیت کار

می کند

تغییر اندازه جهش : option2

جایگزین کردن ثابت σ با تابع $\sigma(t)$ و Update بعد از هر n گام با $1/5$ قانون (cf. ES chapter)

$$\sigma(t) = \begin{cases} \sigma(t-n)/c & \text{if } p_s > 1/5 \\ \sigma(t-n) \cdot c & \text{if } p_s < 1/5 \\ \sigma(t-n) & \text{otherwise} \end{cases}$$

ویژگی ها :

تغییرات در σ مبنایی است روی بازخورد از پیشرفت جستجو
تقریبا کنترل کاربر بوسیله σ در فرمول بالا
 σ قابل پیش بینی نیست

یک σ داده شده روی همه موجودیت های جمعیت کار می کند

تغییر اندازه جهش : option3

- تخصیص یک σ خصوصی برای هر موجودیت
- $(x_1, \dots, x_n, \sigma)$ جادادن σ در کروموزوم
- کاربرد اپراتورهای مختلف از x_i' و σ

$$\sigma' = \sigma \times e^{N(0, \tau)}$$

$$x'_i = x_i + N(0, \sigma')$$

ویژگی ها :

تغییرات در σ انتخاب طبیعی را نتیجه می دهد
تقریبا هیچ کنترلی روی σ وجود ندارد
 σ قابل پیش بینی نیست

یک σ داده شده روی یک موجودیت کار میکند

تغییر اندازه جهش : option4

- تخصیص یک σ خصوصی به هر متغیر برای هر موجودیت
- $(x_1, \dots, x_n, \sigma)$ جادادن σ' در کروموزوم ها
 $(x_1, \dots, x_n, \sigma_1, \dots, \sigma_n)$
- کاربرد اپراتورهای مختلف x'_i و σ'_i
 $\sigma'_i = \sigma_i \times e^{N(0, \tau)}$
 $x'_i = x_i + N(0, \sigma'_i)$ ویژگی ها :
- تغییرات در σ_i انتخاب طبیعی را نتیجه می دهد
- تقریباً هیچ کنترلی روی σ_i وجود ندارد
- σ_i قابل پیش بینی نیست
- یک σ_i داده شده روی یک ژن از یک موجودیت کار میکند

Example cont'd

inequality constraints

for $i = 1, \dots, q$

Constraints

$g_i(x) \leq 0$ —

equality constraints

for $i = q+1, \dots, m$

$h_i(x) = 0$ —

are handled by penalties:

$$eval(x) = f(x) + W \times penalty(x)$$

$$penalty(x) = \sum_{j=1}^m \begin{cases} 1 & \text{for violated constraint} \\ 0 & \text{for satisfied constraint} \end{cases} \quad \text{where}$$

Varying penalty: option 1

Replace the constant W by a function $W(t)$

$$W(t) = (C \times t)^\alpha$$

$0 \leq t \leq T$ is the current generation number

ویژگی ها:

تغییرات W مستقل از پیشرفت جستجو می باشد.

یک کنترل قوی کاربر W در فرمول بالا

W کاملاً قابل پیش بینی است

یک W داده شده روی همه موجودیت ها کار می کند.

Varying penalty: option 2

Replace the constant W by $W(t)$ updated in each generation

$$W(t+1) = \begin{cases} \beta \times W(t) & \text{if last } k \text{ champions all feasible} \\ \gamma \times W(t) & \text{if last } k \text{ champions all infeasible} \\ W(t) & \text{otherwise} \end{cases}$$

$\beta < 1, \gamma > 1, \beta \times \gamma \neq 1$ champion: best of its generation

ویژگی ها:

تغییرات W بر اساس بازخورد پیشرفت جستجو است.

برخی کنترل کاربرا W در فرمول بالا است

W قابل پیش بینی نیست

W داده شده روی همه موجودیت ها جمعیت کار می کند.

Varying penalty: option 3

Assign a personal W to each individual
Incorporate this W into the chromosome: (x_1, \dots, x_n, W)

Apply variation operators to x_i 's and W

Alert:

$$eval((x, W)) = f(x) + W \times penalty(x)$$

while for mutation step sizes we had

$$eval((x, \sigma)) = f(x)$$

this option is thus sensitive “cheating” \Rightarrow makes
no sense

- اشکال مختلف پارامتر کنترلی متمایز میشوند با :
- ویژگی های اولیه:
- چه مولفه ای را در EA تغییر دهیم.
- چطور این تغییر ساخته میشود
- ویژگی های ثانویه:
- پشتیبان گیری از تغییرات **evidence/data**
- تغییرات **level/scope**

- هر کدام از مولفه های EA عملاً میتوانند پارامتری شوند بدین ترتیب در جنبش کنترل میشوند

نمایش

تابع ارزیابی

اپراتورهای مختلف

انتخاب اپراتور (والد یا انتخاب جفت)

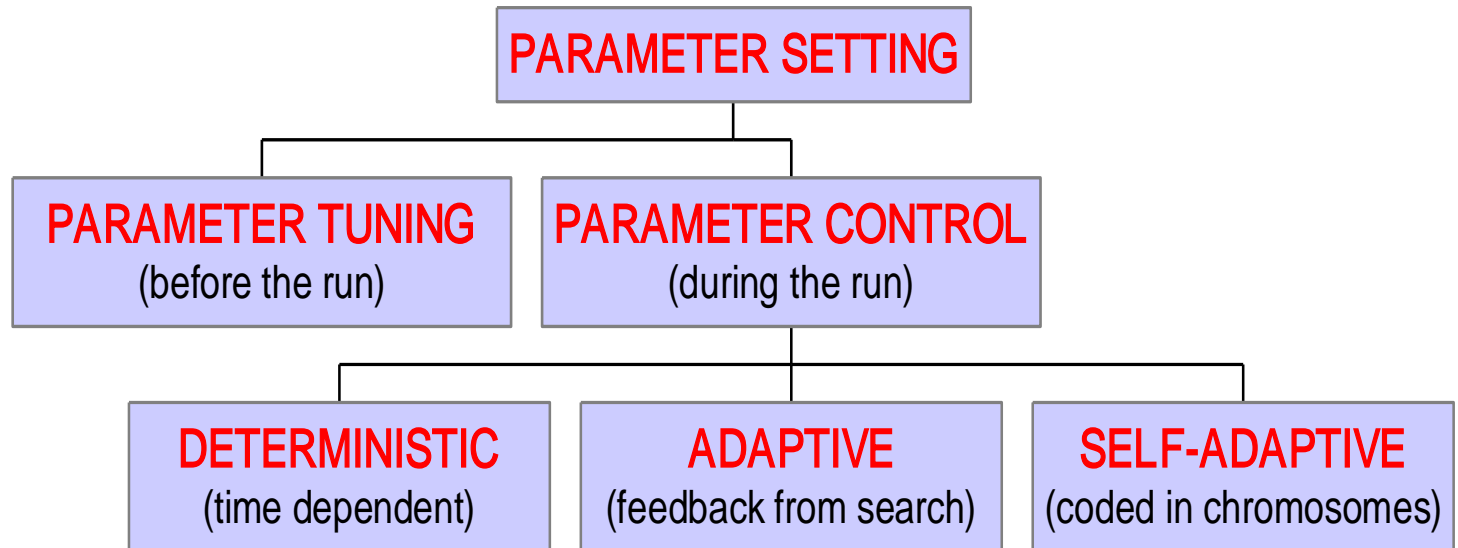
جایگزینی اپراتور (بقا یا انتخاب محیط)

جمعیت (اندازه، توپولوژی)

چطور

- ۳ نوع عمده از پارامترهای کنترل:
- **deterministic (قطعی)**
برخی قوانین پارامترهای استراتژی را اصلاح می نماید بدون بازخورد گرفتن از جستجو
- **adaptive (وفقی)**
بازخورد قوانین مبتنی بر برخی سنجش ها پیشرفت جستجو را مونیتور می نماید
- **self-adaptative (خود انطباق)**
- مقادیر پارامترها همراه با جوابها بدست می آید، کدگذاری روی کروموزوم ها دستخوش انتخاب و دگرگونی است

Global taxonomy



ملاک آگاهی تغییرات

- تغییرات پارامتر ممکن است بر پایه :
- **time or nr. of evaluations** (deterministic control)
nr کنترل قطعی زمان یا ارزیابی
- **population statistics** (adaptive control)
 - کنترل وفقی - جمعیت آماری
 - ساخت پیشروی
 - تنوع جمعیت
 - توزیع ژن
- نسبت مناسب موجودیت های داده شده با مقدار مناسب (adaptive or self-adaptive control)

- گواهی مطلق:

تغییر رویداد از پیش تعریف شده و افزایش p_m با ۱۰٪ گوناگونی جمعیتی که جهت آنها زیر آستانه X بوده و مقدار تغییر ثابت باشد.

- گواهی نسبی:

- مقایسه مقادیر جوابهایی که ایجاد شده اند و افزایش p_m اگر کیفیت بالا باشد تولد آن با نرخ بالا جهش می باشد که جهت و مقدار آن ثابت نمی باشد.

قلمرو/سطح

پارامتر میتواند اثر روی سطوح متفاوت داشته باشد:

محیط (تابع مناسب)

جمعیت

موجودیت

بخشی از موجودیت

توجه : مولفه داده شده (پارامتر) احتمال دارد تعیین شده باشد
بنابراین : قلمرو/سطح یک مشتق می باشد یا ویژگی ثانویه در رویه
طبقه بندی می باشد.

Refined taxonomy

Combinations of types and evidences ●

Possible: + ●

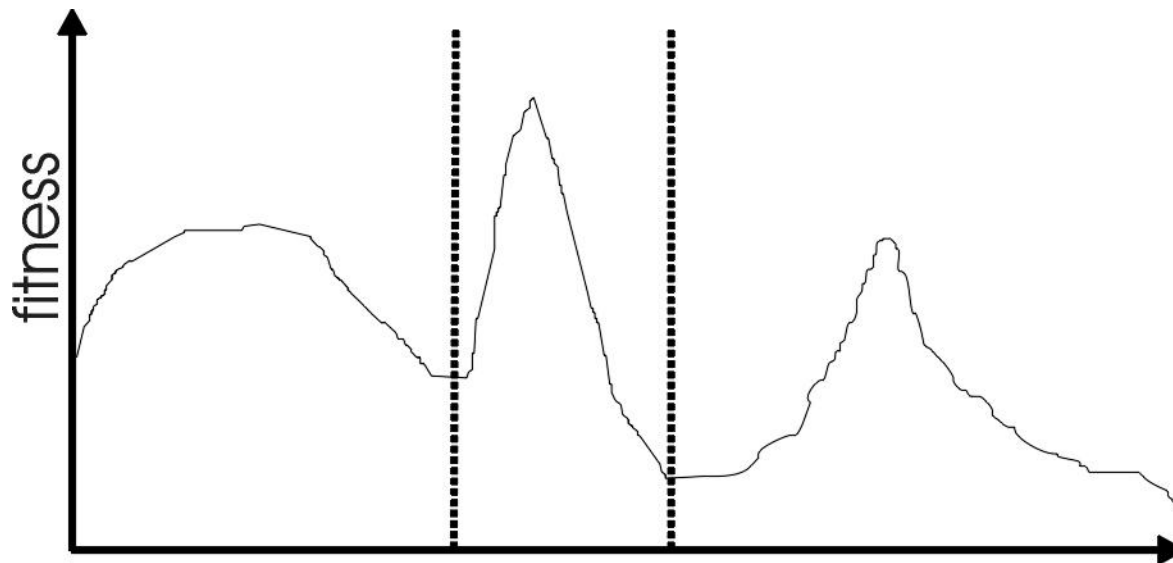
Impossible: - ●

	Deterministic	Adaptive	Self-adaptive
Absolute	+	+	-
Relative	-	+	+

- پارانترهای کنترل پیشنهاد می کنند امکان استفاده مقادیر مناسب در مراحل گوناگون جستجو
- پارامترهای کنترل Adaptive and self-adaptive به کاربران
 - پیشنهاد می کنند نجات از میزان سازی پارامتر "liberation" from parameter tuning
 - نمایندگی وظیفه تنظیم پارامتر در فرآیند تکاملی
 - مطلب قبلی یک وظیفه مضاعف برای EA دارد: حل مسئله + اندازه گیری خودش (بالاسری)
- اطمینان، غیر حساسی در EA برای فرضیات مختلف
- اگر تعداد پارامترها افزایش یابد بوسیله (self)adaptation
- برای meta-parameters این روش ها را معرفی می نمائیم

Motivation 1: Multimodality

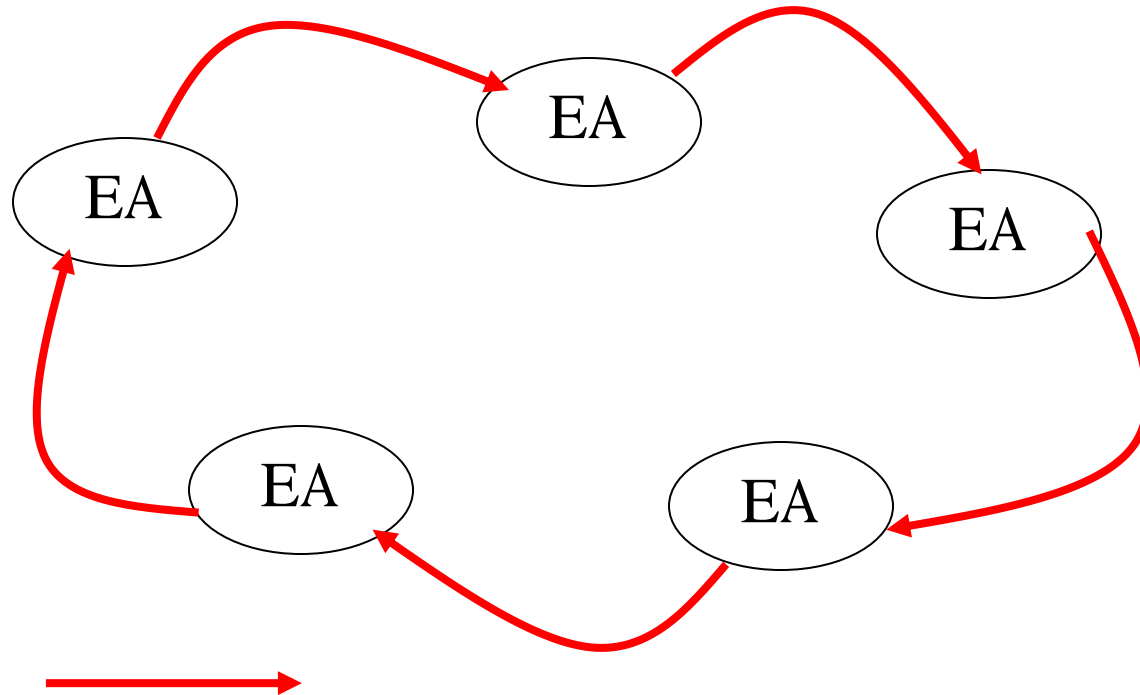
اکثراً در مسائلی که بیشتر از یک جواب محلی بهینه دارند استفاده می شوند



توده ژنتیکی: Motivation 2

- جمعیت محدود با (panmictic) عمومی مخلوط و انتخاب سراجام همگرایی دور یک موجودیت بهینه
- اغلب ممکن است بخواهیم یکی نمائیم چندین قله را
- این میتواند کمک کند بهینه سازی کلی را وقتی که
- بخشی بینه شود که بیشترین حفره را جذب نماید

Implicit 1: “Island” Model Parallel EAs



مهاجرت متناوب از جوابهای موجود بین جمعیت

Island Model EAs contd:

- اجرای مضاعف جمعیتها بصورت موازی، در برخی از انواع ساختار ارتباطات (معمولا یک حلقه یا هلال)
- بعد از یک (معمولا ثابت) تعداد نسلها (یک دوره) ، موجودیت ها باهمسایگی ها مبادله می شوند و این تکرار میشود تا شرایط بازدید متوقف شود.
- غیر منصفانه است سیستم های موازی/خوشه بندی را القاء کنیم

Island Model Parameters 1

- در هر جزیره نمی توانیم اپراتورهای گوناگونی بکار ببریم.
- چطور اغلب موجودیت ها مبادله می شوند؟
- همچنین سریع هستند و همه می پرند با توجه به جوابهای یکسان
- همچنین آهسته اند و زمان را بیهوده تلف می کنند.
 - اکثر نویسندگان بکار میبرند محدوده ۲۵ تا ۱۵۰ ژن
 - میتوانیم انجام دهیم بصورت وفقی (توقف هر پرش زمانی که
 - هیچ پیشرفت برای ماندن ۲۵ ژن)

Island Model Parameters 2

How many, which individuals to exchange ? •

usually $\sim 2-5$, but depends on population size. —

more sub populations usually gives better results —

but there can be a “critical mass” i.e. minimum size
of each sub population needed

Martin et al found that better to exchange —

randomly selected individuals than best

can select random/worst individuals to replace —

Explicit 1: Fitness Sharing

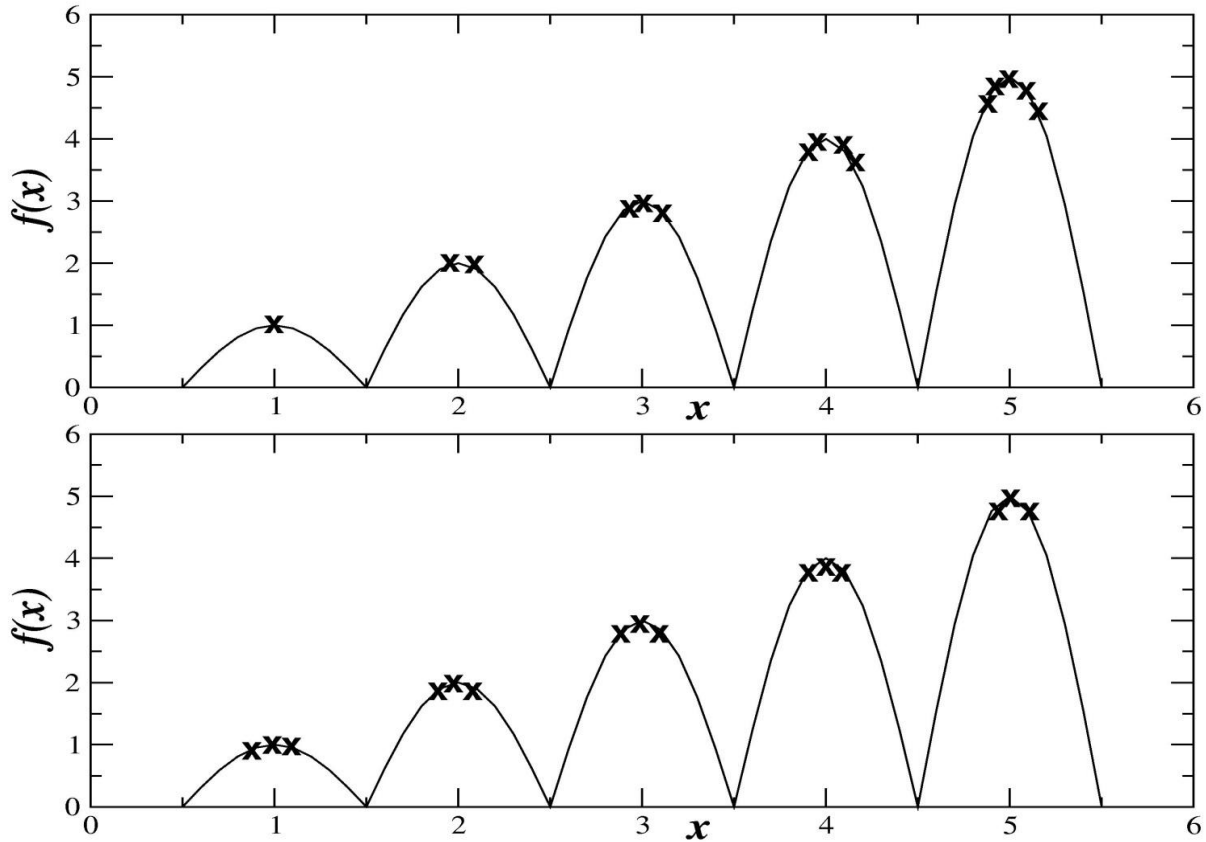
- Restricts the number of individuals within a given niche by “sharing” their fitness, so as to allocate individuals to niches **in proportion to the niche fitness**
- need to set the size of the niche σ_{share} in either genotype or phenotype space
- run EA as normal but after each gen set

$$sh(d) = \begin{cases} 1 - d / \sigma & d < \sigma \\ 0 & \text{otherwise} \end{cases}$$
$$f'(i) = \frac{f(i)}{\sum_{j=1}^{\mu} sh(d(i, j))}$$

Explicit 2: Crowding

- Attempts to distribute individuals **evenly** amongst niches
- relies on the assumption that offspring will tend to be close to parents
- uses a distance metric in ph/g enotype space
- randomly shuffle and pair parents, produce 2 offspring
- 2 parent/offspring tournaments - pair so that $d(p1,o1)+d(p2,o2) < d(p1,o2) + d(p2,o1)$

Fitness Sharing vs. Crowding



Multi-Objective Problems (MOPs)

- Wide range of problems can be categorised by the presence of a number of n possibly conflicting objectives:
 - buying a car: speed vs. price vs. reliability –
 - engineering design: lightness vs strength –
- Two part problem:
 - finding set of good solutions –
 - choice of best for particular application –

MOPs 1: Conventional approaches

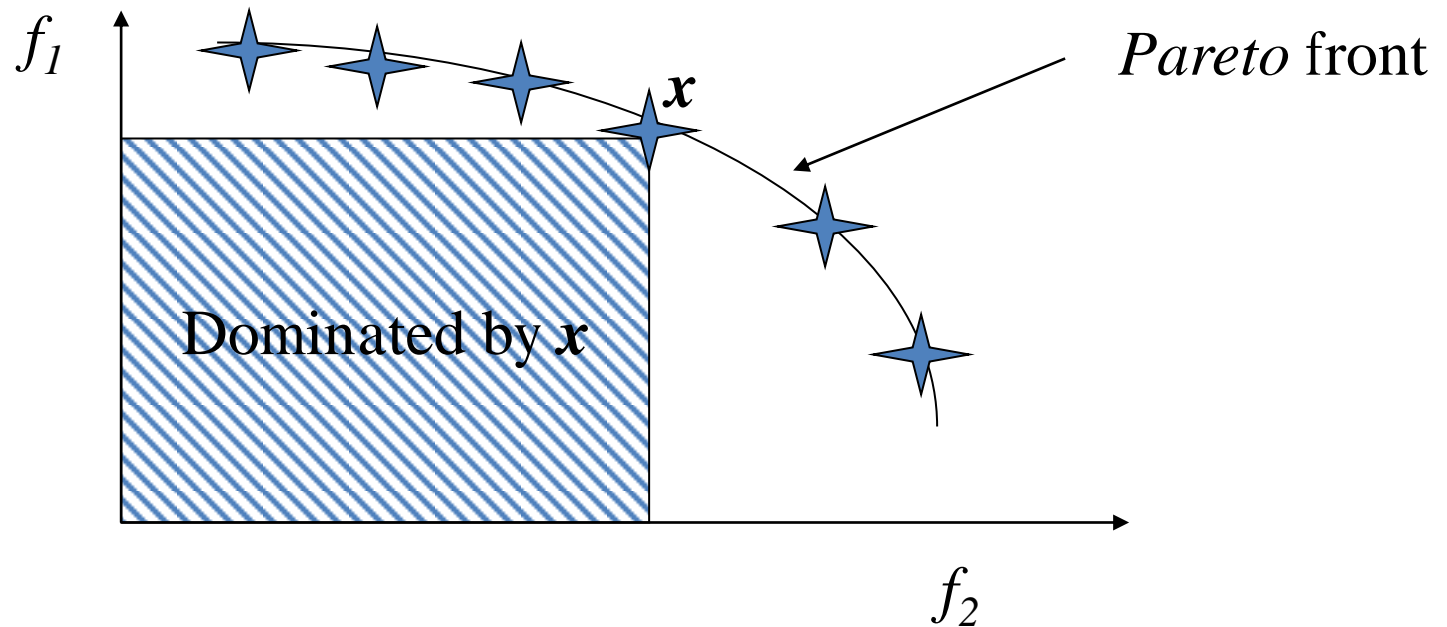
rely on using a weighting of objective •
function values to give a single scalar
objective function which can then be
optimised:

$$f'(x) = \sum_{i=1}^n w_i f_i(x)$$

to find other solutions have to re- •
optimise with different w_i .

MOPs 2: Dominance

we say x dominates y if it is at least as good on all criteria and **better** on at least one



MOPs 3: Advantages of EC approach

- Population-based nature of search means you can *simultaneously* search for set of points approximating Pareto front
- Don't have to make guesses about which combinations of weights might be useful
- Makes no assumptions about shape of Pareto front - can be convex / discontinuous etc

MOPs 4: Requirements of EC approach

- Way of assigning fitness, usually based on dominance –
- Preservation of diverse set of points
- similarities to multi-modal problems –
- Remembering all the non-dominated points you've seen usually using elitism or an archive –

MOPs 5: Fitness Assignment

- Could use aggregating approach and change weights during evolution
 - no guarantees –
- Different parts of pop use different criteria
 - e.g. VEGA, but no guarantee of diversity –
- Dominance
 - ranking or depth based –
 - fitness related to whole population –

MOPs 6: Diversity Maintenance

Usually done by niching techniques •
such as:

fitness sharing –
adding amount to fitness based on inverse –
distance to nearest neighbour
(minimisation)

(adaptively) dividing search space into –
boxes and counting occupancy

All rely on some distance metric in •
genotype / phenotype space

MOPs 7: Remembering Good Points

- Could just use elitist algorithm
e.g. $(\mu + \lambda)$ replacement –
- Common to maintain an archive of non-dominated points
some algorithms use this as second –
population that can be in recombination etc
- others divide archive into regions too
e.g. PAES

Summary

	ES	EP	GA
Representation	Real-valued	Real-valued	Binary-valued
Fitness is	Objective function value	Scaled objective function value	Scaled objective function value
Self-adaptation	Standard deviations and rotation angles	None (standard-EP), variances (meta-EP), correlation coefficients	None
Mutation	Gaussian, main operator	Gaussian, only operator	Bit-inversion, background operator
Recombination	Discrete and intermediate, sexual and panmictic, important for self-adaptation	None	z -point crossover, uniform crossover, only sexual, main operator

Selection	Deterministic, extinctive or based on preservation	Probabilistic, extinctive	Probabilistic, based on preservation
Constraints	Arbitrary inequality constraints	None	Simple bounds by encoding mechanism
Theory	Convergence rate for special cases, (1+1)-ES, (1 ⁺ , λ)-ES, global convergence for (μ+λ)-ES	Convergence rate for speacial cases, (1+1)-EP, global convergence for (1+1)-EP	Schema processing theory, global convergence for elitist version

Biology	ES	EP	GA
Nucleotide base	—	—	Binary digit $a_i \in IB$
Codon	—	—	—
Gene	Single x_i, σ_i, α_j	Single x_i, ν_i	Bitstring segment $(a_{i1}, \dots, a_{il_x}) \in IB^{l_x}$
Chromosome	Complete vectors $\vec{x}, \vec{\sigma}, \vec{\alpha}$	Complete vectors $\vec{x}, \vec{\nu}$	Complete bitstring \vec{a}
Genotype	Collection of chromosomes $(\vec{x}, \vec{\sigma}, \vec{\alpha})$	Collection of chromosomes $(\vec{x}, \vec{\nu})$	Complete bitstring \vec{a}
Phenotype	Component \vec{x}	Component \vec{x}	Decoded structure $\vec{x} = \Upsilon(\vec{a})$

Table 2.6. Relation of artificial and natural information representation.

Karl Sims, Galápagos



Box insect



Beaded arms



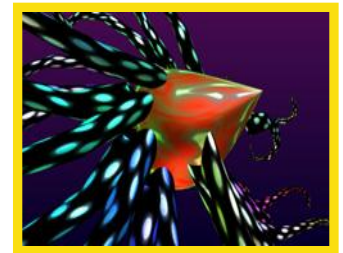
Multipus-green



Jellyfish



Bfly larva



Multipus-purple