

自然中的算法

自然启发算法

<http://www.cs.cmu.edu/~02317/>

Ziv Bar-Joseph
zivbj@cs.cmu.edu
GHC 8006

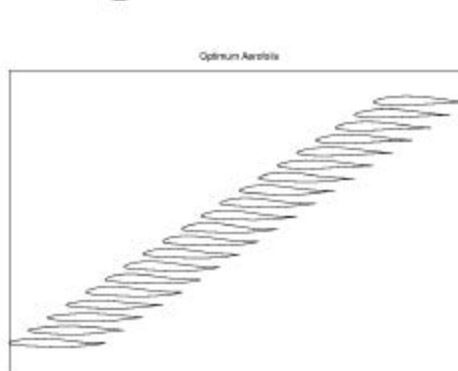
Matt Ruffalo
mruffalo@andrew.cmu.edu
GHC 7715

概览

- **为什么要向自然学习?**
- **自然启发算法 / 从自然习得的算法**
 - **差分进化算法**
 - **其他优化方法**
 - **双向研究**
- **应用**

向自然学习

- 自然界演化出了处理信息处理问题的高效方法
- 模仿这类自然过程的过程通常被称为“自然启发”
- 工程实例：飞机机翼设计



(另一个) 工程实例：子弹 列车

列车车头仿照翠鸟的喙设计，翠鸟能平滑地俯冲入水。
(来源：《大众机械》)



优化

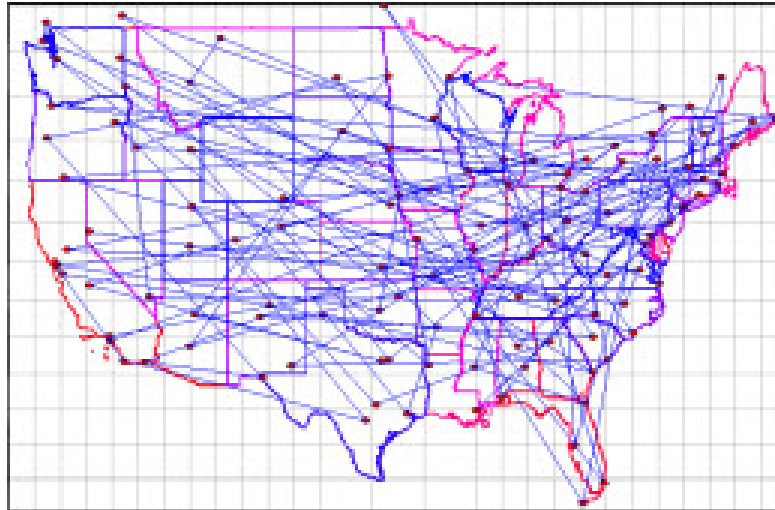
- 使某事物尽可能完善、实用或高效的行为、过程或方法论。（韦氏词典）
- 鸟类：最小化阻力。
- 考虑如下形式的优化问题：

$$\textit{Min} \quad \{ f(x) \}$$

$$\textit{s.t.} \quad x \in S \subset R^n$$

优化问题： 示例

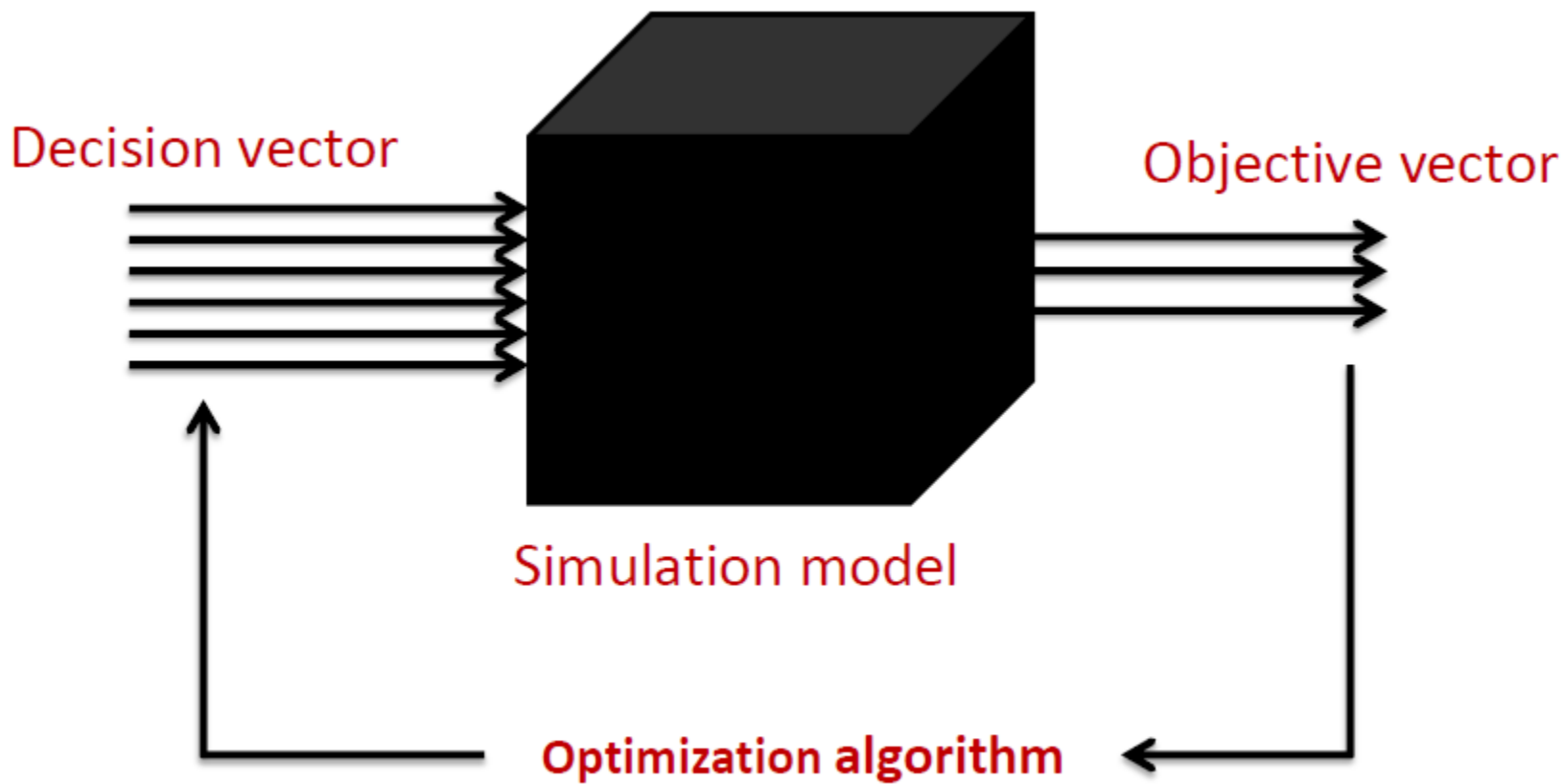
访问全部 50 个州首府的最快 / 最省路线



常见优化问题的特征

- 目标函数与约束函数可能不可微。
- 约束是非线性的。
- 搜索空间离散 / 不连续。
- 混合变量（整数、实数、布尔等）。
- 约束与变量数量庞大。
- 目标函数可能是多峰的，存在多个最优解。
- 难以用闭式求解，计算代价高。

迭代求解优化问题

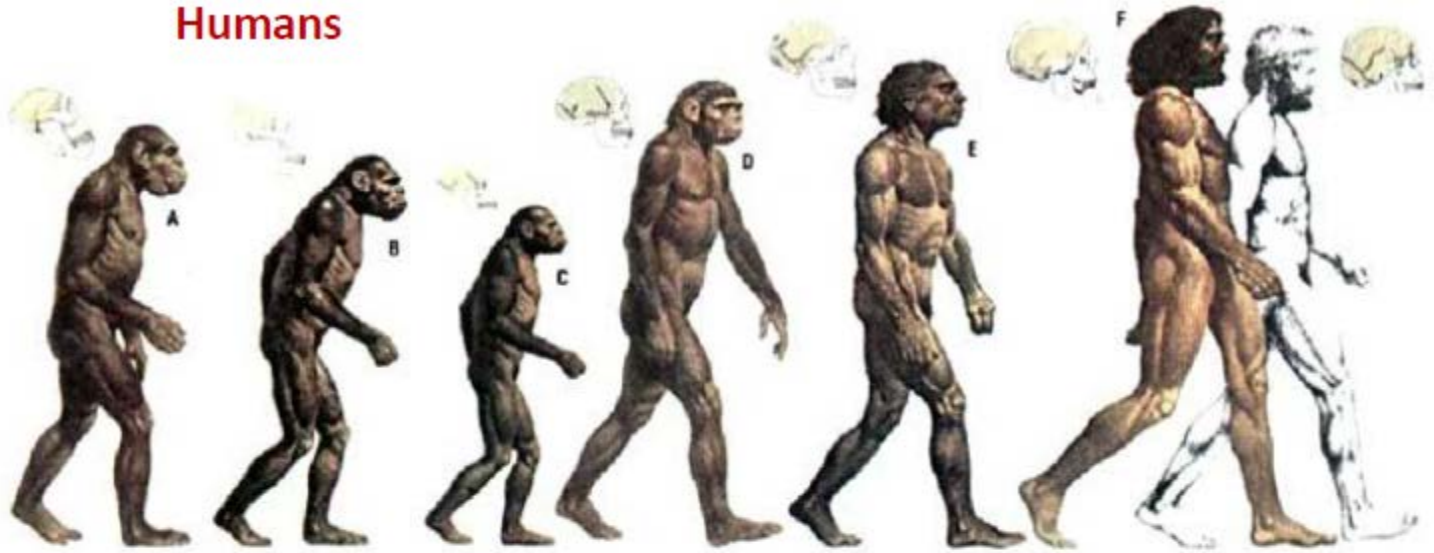


求解优化问题

- 不同类型的问题需要不同的方法。
- 常常陷入局部最优（缺乏全局视角）。
- 某些方法（例如基于梯度下降的回归）需要知道目标函数与约束的一阶/二阶导数。

进化

Humans



Macintosh

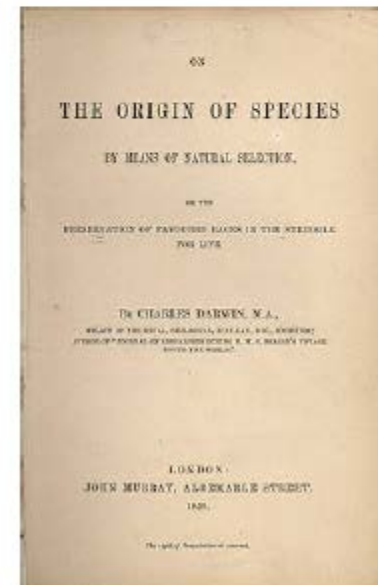


进化算法

- 后代通过繁殖、变异等方式产生。
- 自然选择——一种有导向的搜索过程
- 适应环境的个体得以存活、繁殖，并把遗传性状传给后代
- 种群适应环境；变异随时间累积，产生新物种



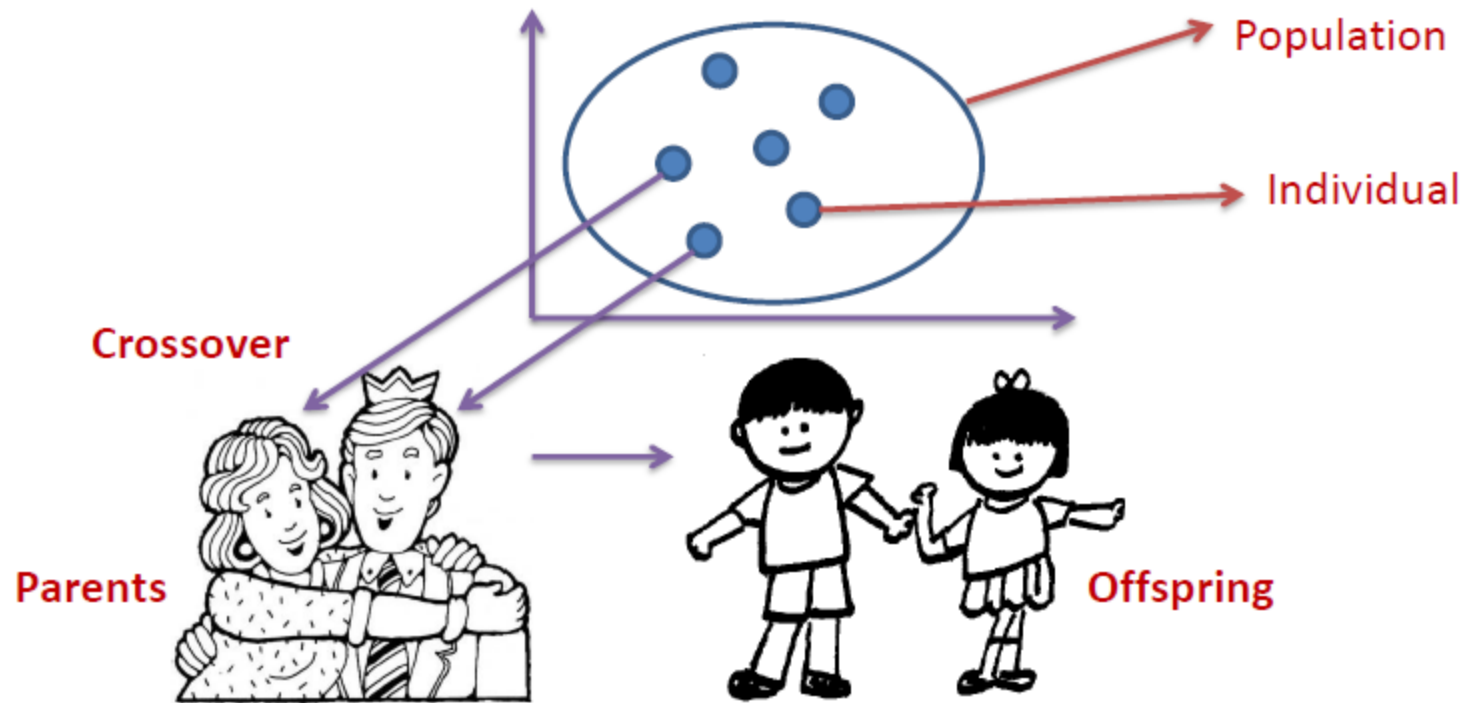
Charles
Darwin



进化算法

术语

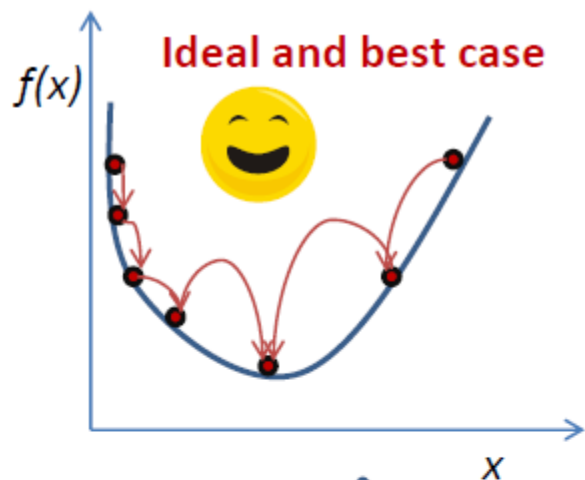
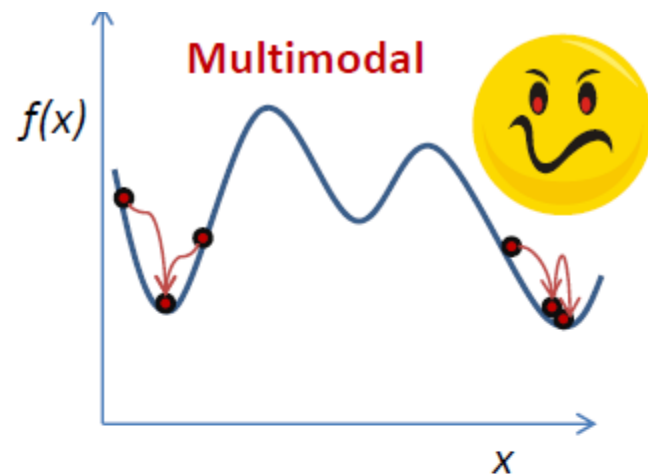
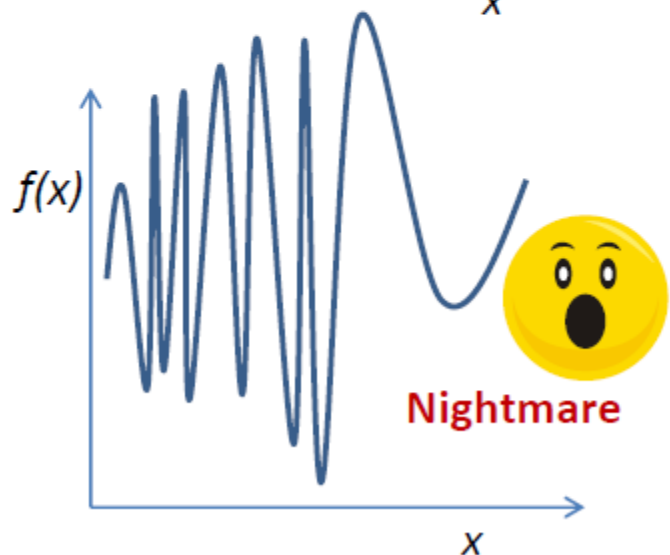
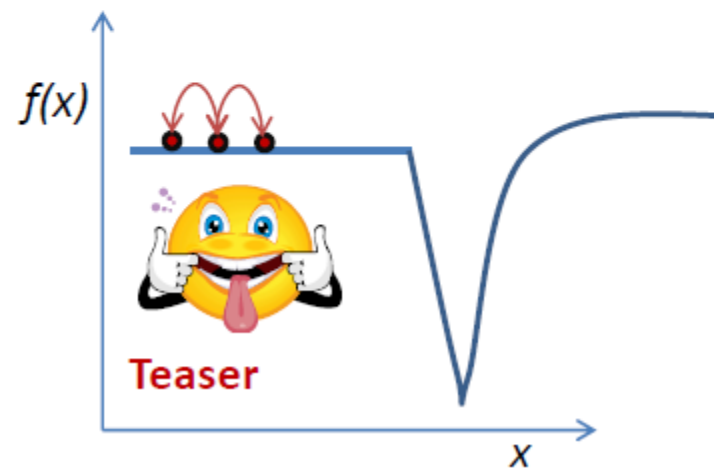
1. 个体——携带遗传信息（染色体），由其在搜索空间中的状态与适应度（目标函数值）刻画。
2. 种群——个体的集合，遗传算子作用于其上。
3. 适应度函数——“适应度函数”常作为目标函数的同义词使用。
4. 代——进化算法的（自然）时间单位，即一次迭代步。



总体思路

- 选择——轮盘赌、锦标赛、稳态选择等
- 其动机是保留最优个体（复制多份）并淘汰最差个体
- 交叉——模拟二进制交叉、线性交叉、混合交叉等
- 通过同时考虑多个个体生成新解——在全局范围内搜索新的、有望更优的解
- 变异——多项式变异、随机变异等
- 保持种群多样性——010110 → 010100（按位变异）

进化方法 vs. 基于梯度下降的方法

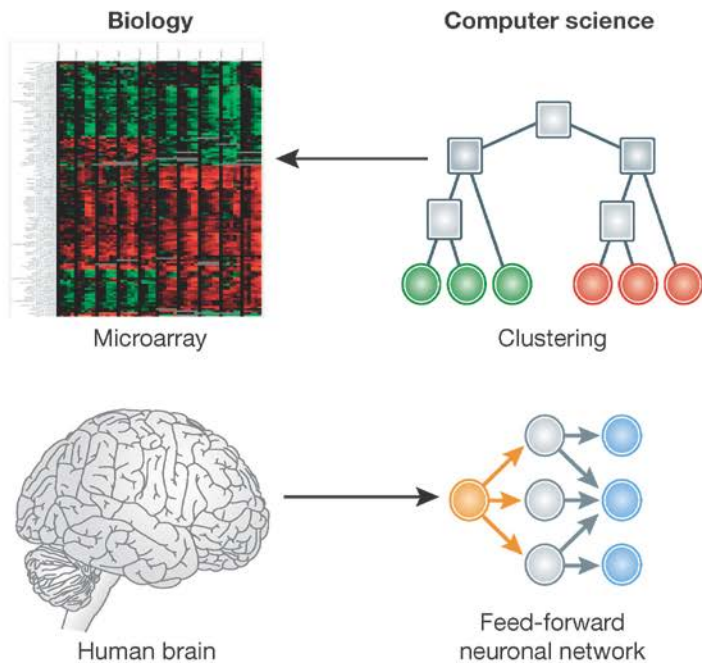


局限性

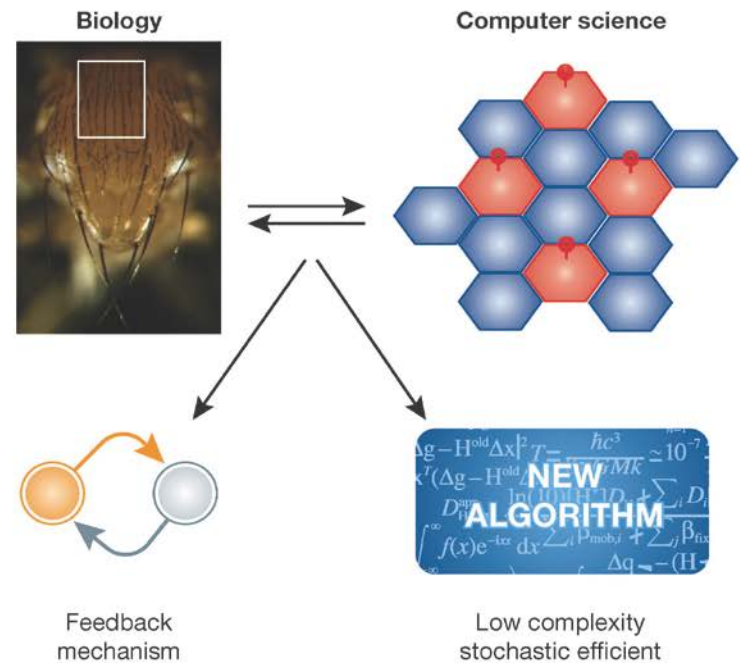
- 无法保证在有限时间内找到最优解
- 关于收敛性的保证相对较少
- 计算开销可能很大

双向研究

A Traditional studies

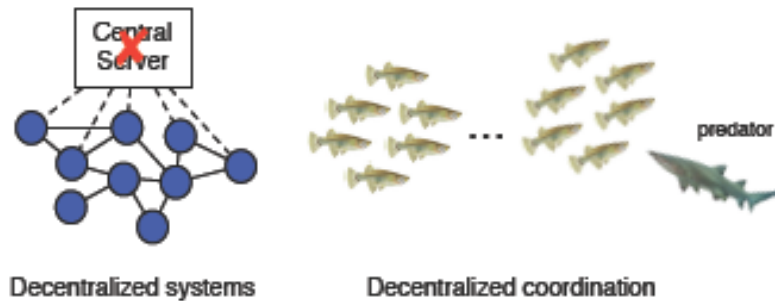


B Computational thinking

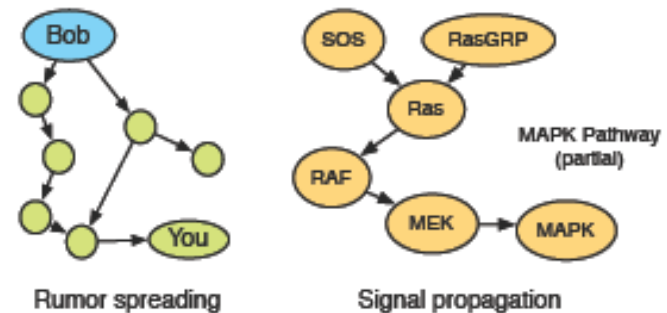


自然中的算法：计算机科学与生物学的共通原理

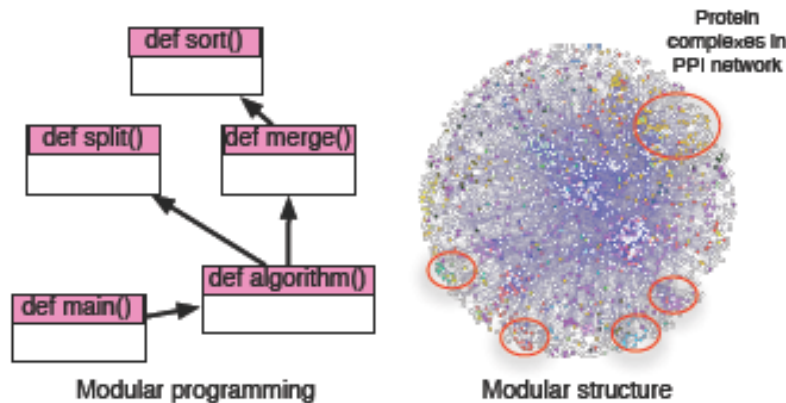
A) Distributed computing



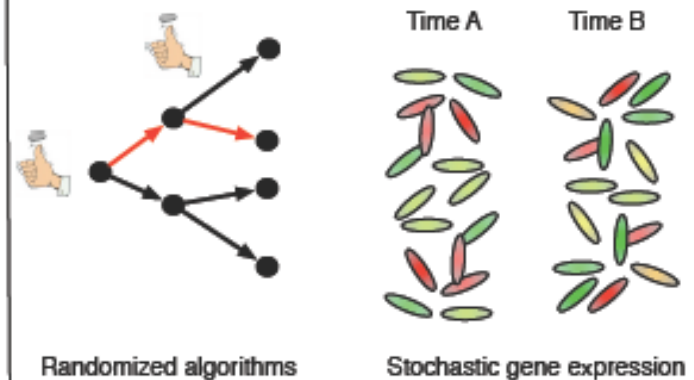
B) Network processes



C) Reusable components



D) Randomness and stochasticity



自然中的算法

优化

什么是优化？

- 从已定义集合中选取一个元素，
使给定准则最大化（或最小化）

输入集合通常是
良定义的

待求值的公式
潜在解通常
易于使用

在许多情况下
优化涉及
在一组潜在解上搜索

形式化定义

- 一种数学表述

- 给定：一个从某个集合 A 映射到实数的函数 $f: A \rightarrow \mathbb{R}$

- 求解：寻找 A 中的元素 x_0 ，使得对 A 中所有的 x 都有 $f(x_0) \leq f(x)$ （“极小化”）；

- 或使得对 A 中所有的 x 都有 $f(x_0) \geq f(x)$ （“极大化”）。

挑战

- **问题 / 解可能是：**
 - **高维的**
 - **存在大量局部最优**
 - **位于巨大的搜索空间中**

示例

- 最小化从生产工厂到仓库的运输成本
- 旅行商问题
- 布置传感器以使有效信息最大化
- 确定一系列药物的给药时机以获得最大疗效
- 为城市交通网络寻找最优的红—黄—绿信号配时
- 等等.....

极小值搜索算法

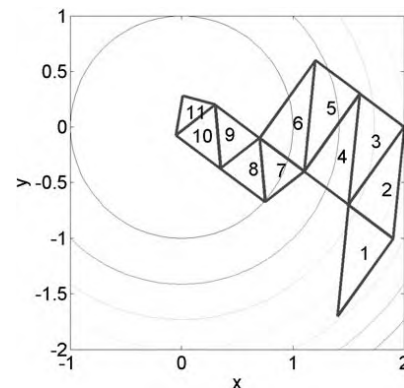
1. 穷举搜索 = 暴力搜索



2. 解析优化

$$\nabla f(\mathbf{x}) = 0, \quad \frac{\partial^2 f}{\partial^2} - ?$$

3. Nelder-Mead 下山单纯形法



4. 基于搜索方法的优化 (坐标搜索法、最速下降算法、牛顿法等.....)

主要方向

- **线性规划是一类凸规划，研究目标函数 f 为线性、且约束集合仅用线性等式与不等式来刻画的情况。**
- **非线性规划研究一般情形：目标函数或约束（或两者）含有非线性部分。**

主要方向

- **整数规划研究部分或全部变量被限制取整数值的线性规划。它不是凸问题，通常比常规线性规划困难得多。**

主要方向

- 启发式与元启发式对被优化问题几乎不做假设。
通常
启发式并不保证一定能找到最优解；但另一方面，
启发
式被用于为许多复杂的优化问题寻找近似解。

启发式方法

- 启发式算法

- 启发式算法不保证能找到最优解。
- 启发式算法甚至未必存在“最坏表现”的界。
- 然而在实践中，启发式算法（简称启发式）往往很成功。

为什么不使用精确方法？

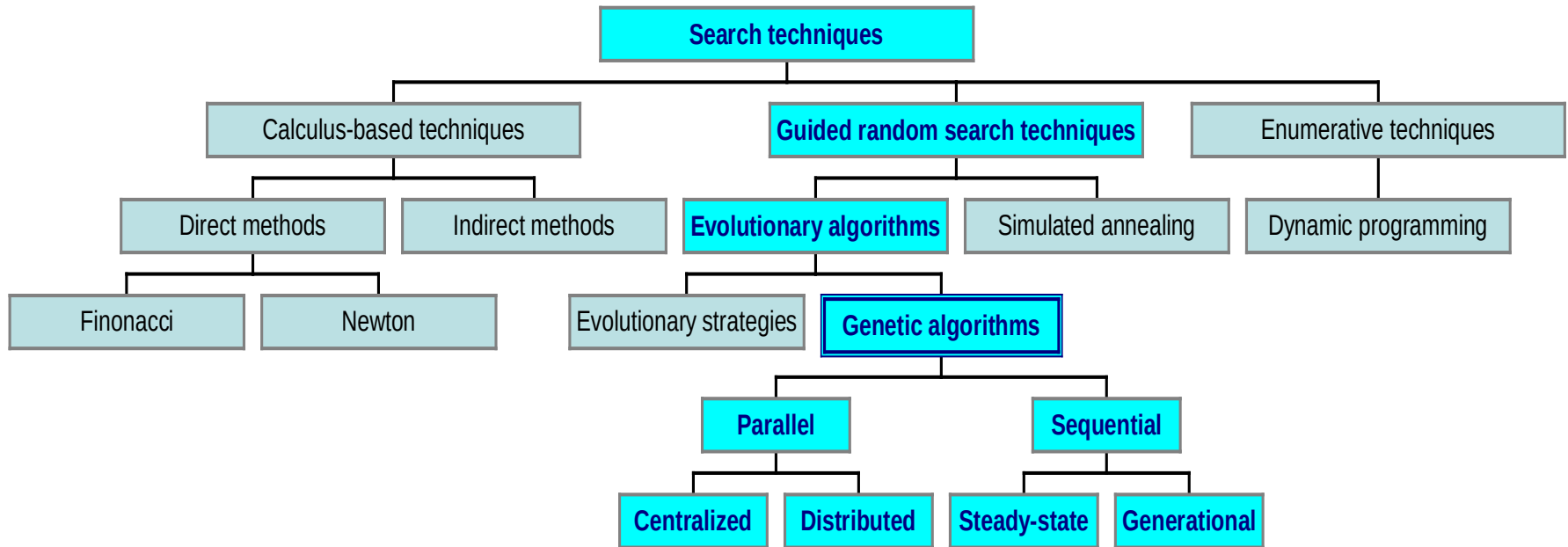
- **运行时间**

- 依问题而定，精确方法的运行时间可能是指数级的

- **真实世界问题与形式化问题之间的联系可能很弱**

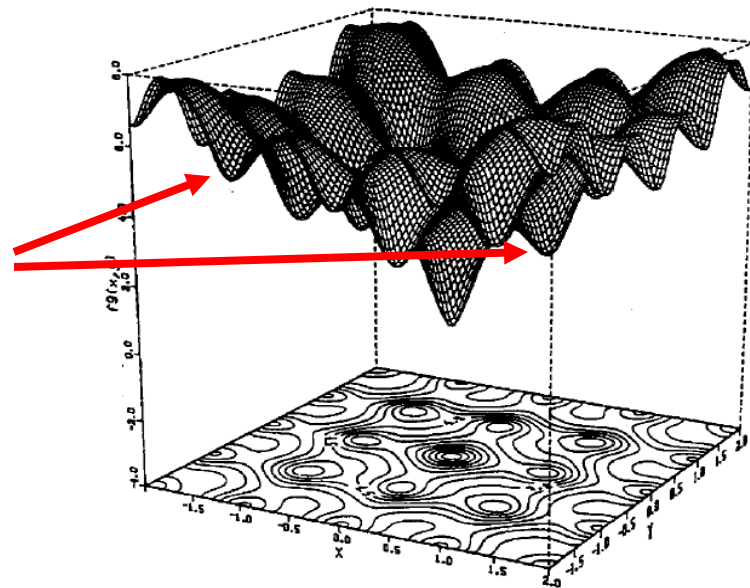
- 有时你无法恰当地建立一个涵盖真实问题所有方面的数学问题。如果所求的并不是那个真正的问题，那么得到一个（或多个）启发式解，可能比得到形式化问题的最优解更有用。

搜索技术的分类



受生物学启发的 优化算法

多次局部搜索可能收敛到
同一个局部极小值！



自然优化方法
并非万能，但.....

模拟退火
(Kirkpatrick 等人, 1983)

遗传算法
(Holland, 1975)

进化算法
(Schwefel, 1995)

粒子群优化
(Parsopoulos 与 Vrahatis, 2002)

无需导数、搜索空间大、“基于自然”

自然中的算法

遗传算法

历史

遗传算法的历史

- 早在 1962 年, John Holland 关于自适应系统的研究
为后来的发展奠定了基础。
- 到 1975 年, Holland 与他的学生和同事们出版了
《自然与人工系统中的适应性》一书。

遗传算法的历史

- 20 世纪 80 年代前期到中期，遗传算法被应用到广泛的学科领域。
- 1992 年，John Koza 用遗传算法来演化出能够完成特定任务的程序。他把这种方法称为“遗传编程”(GP)。

什么是遗传算法

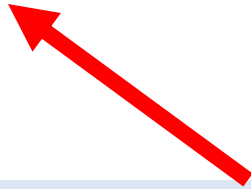
- **遗传算法（GA）是计算中使用的一种搜索技术，用于寻找优化问题与搜索问题的真实解或近似解。**
- **遗传算法被归类为全局搜索启发式。**
- **遗传算法是进化算法中的一个特定类别，它使用受进化生物学启发的技术，如遗传、变异、选择和交叉（也称重组）。**

什么是遗传算法

- **进化通常从随机生成的个体种群开始，并逐代进行。**
- **在每一代中，都要评估种群中每个个体的适应度，并根据适应度从当前种群中选出若干个体，加以修改以形成新的种群。**

什么是遗传算法

- 新种群用于算法的下一迭代。
- 当已经产生最大代数，或种群达到了令人满意的适应度水平时，算法终止。



**没有收敛准则，
也没有保证！**

术语表

- 个体——任意可能的解
- 种群——所有个体的集合
- 适应度——我们正在优化的目标函数（每个个体都有一个适应度）
- 性状——个体可能具有的方面（特征）
- 基因组——一个个体的全部染色体（性状）的集合。

基本遗传算法

- 从一个大的“种群”开始，它由随机生成的问题的“尝试解”构成
- 反复执行以下操作：
 - 评估每一个尝试解
 - （按概率）保留最优解的一个子集
 - 用这些解生成新的种群
- 当得到令人满意的解时（或时间耗尽时）退出

示例： MAXONE 问题

假设我们要最大化
一个 l 位二进制串中 1 的个数

这是一个平凡的问题吗？

看起来也许是，因为我们可以提前
知道答案

不过，我们也可以把它理解为最大化
正确答案的个数，每个正确答案
编码为 1，对应 l 道困难的 yes/no 问题

示例（续）

- 个体（自然地）被编码为一个 l 位的二进制串
- 对 MAXONE 问题，候选解的适应度 f 就是其遗传编码中 1 的个数
- 我们从 n 个随机串组成的种群开始。
假设 $l = 10$, $n = 6$

示例（初始化）

我们抛一枚均匀硬币 60 次，得到如下初始种群：

$$s_1 = 1111010101 \quad f(s_1) = 7$$

$$s_2 = 0111000101 \quad f(s_2) = 5$$

$$s_3 = 1110110101 \quad f(s_3) = 7$$

$$s_4 = 0100010011 \quad f(s_4) = 4$$

$$s_5 = 1110111101 \quad f(s_5) = 8$$

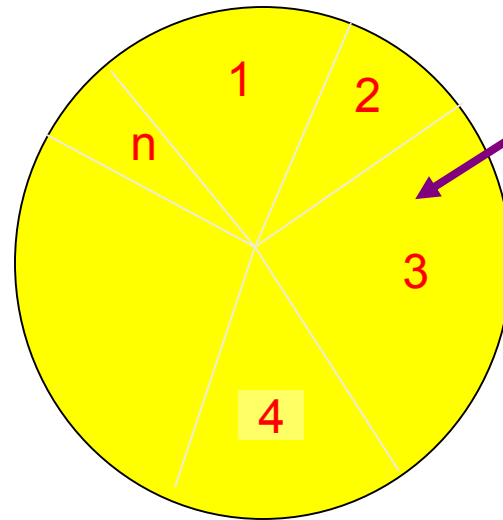
$$s_6 = 0100110000 \quad f(s_6) = 3$$

第 1 步：选择

我们（用一枚有偏硬币）随机地选出一部分个体，依据是它们的适应度：

个体 i 被选中的
概率

$$\frac{f(i)}{\sum_i f(i)}$$



面积为
成正比
与适应度
值

被选出的集合

假设在执行选择之后，我们得到如下种群：

$$s_1' = 1111010101 (s_1)$$

$$s_2' = 1110110101 (s_3)$$

$$s_3' = 1110111101 (s_5)$$

$$s_4' = 0111000101 (s_2)$$

$$s_5' = 0100010011 (s_4)$$

$$s_6' = 1110111101 (s_5)$$

第 2 步：交叉

- 接下来我们让字符串配对以进行交叉。对每一对配偶，我们首先（按某个预先给定的概率，例如 0.6）决定是否真的执行交叉
- 如果我们决定真的执行交叉，就随机选取交叉点，例如 2 和 5

交叉结果

交叉前:

$$s_1' = 11\textcolor{red}{11}010101 \quad s_2' = 11\textcolor{blue}{10}110101$$

交叉后:

$$s_1'' = 11\textcolor{blue}{10}110101 \quad s_2'' = 11\textcolor{red}{11}010101$$

第 3 步：变异

最后一步是施加随机变异：对于我们复制到新种群的每一位，都允许一个很小的出错概率（例如 0.1）。

初始字符串

$s_1'' = 1110110101$

$s_2'' = 1111010101$

$s_3'' = 1110111101$

$s_4'' = 0111000101$

$s_5'' = 0100011101$

$s_6'' = 1110110011$

变异后

$s_1''' = 1110100101$

$s_2''' = 1111110100$

$s_3''' = 1110101111$

$s_4''' = 0111000101$

$s_5''' = 0100011101$

$s_6''' = 1110110001$

然后，开始迭代.....

**在一代之内，种群的总适应度
从 34 变为 37，提高了约 9%**

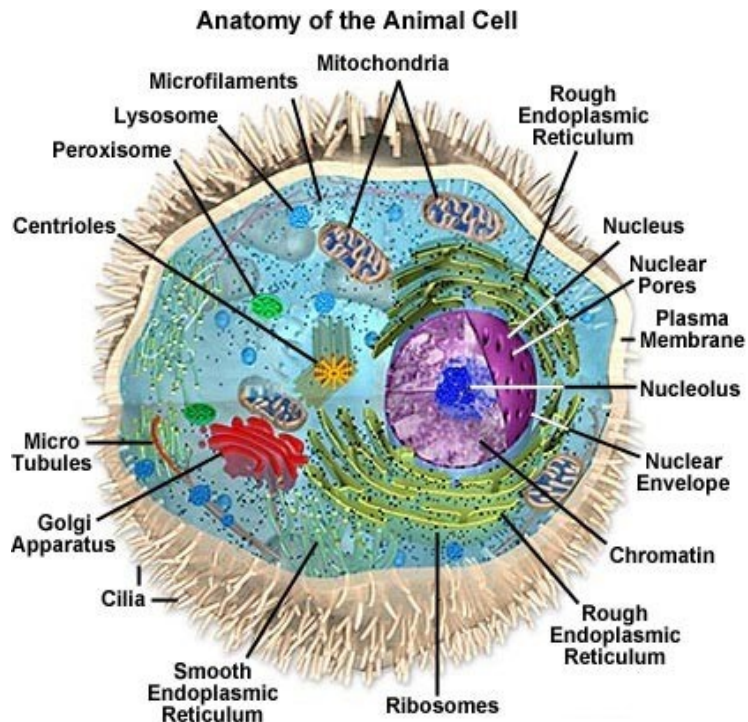
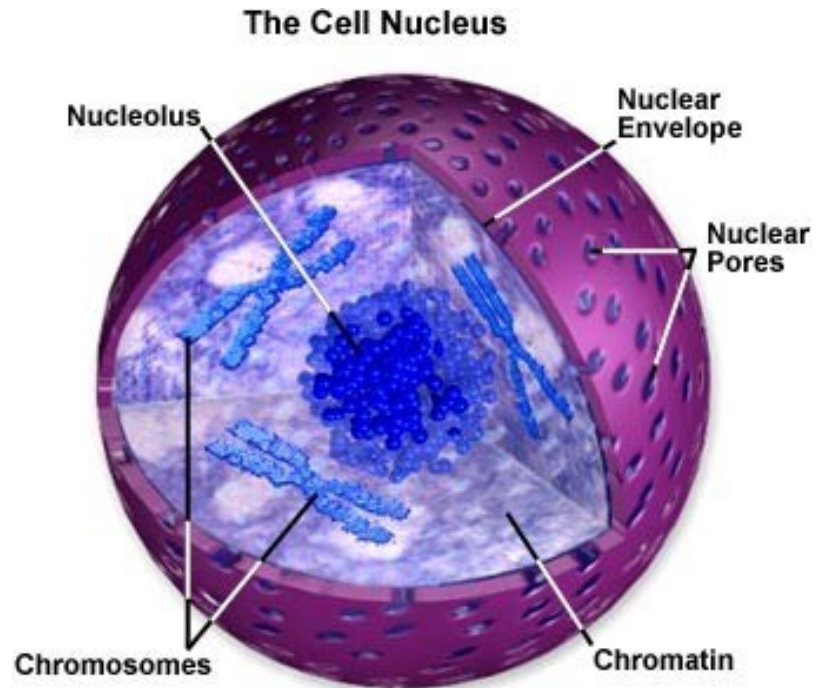
**此时，我们从头重复同样的过程，
直到满足某个停止准则**

生物学动机

生物学背景

“细胞”

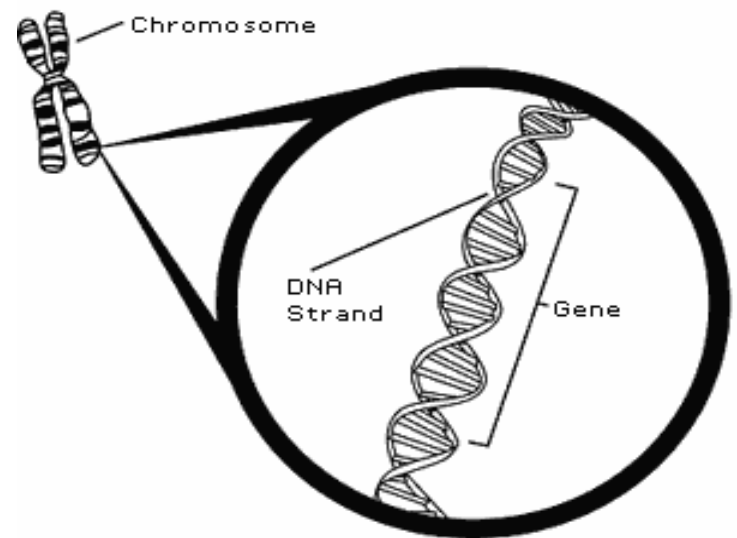
- 每个动物细胞都是由许多小型 “工厂” 共同组成的复合体。
- 细胞核位于细胞的中心。
- 细胞核中含有遗传信息



生物学背景

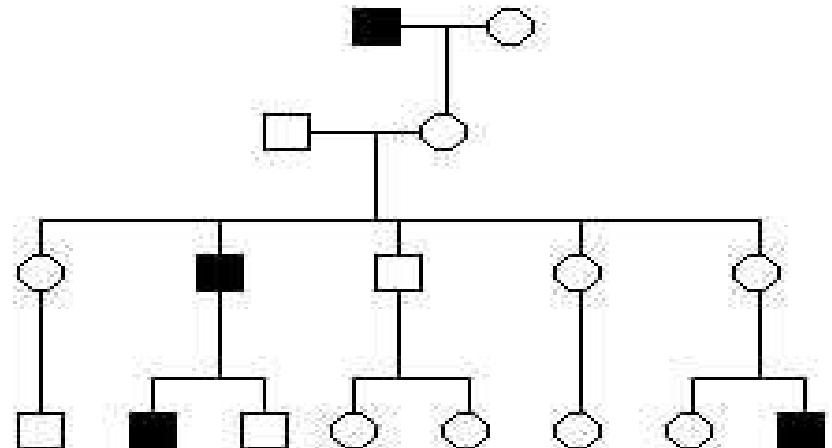
“染色体”

- 遗传信息存储在染色体中
- 每条染色体由 DNA 构成
- 基因编码在染色体上
- 基因编码蛋白质
- 每个基因在染色体上都有唯一的位置



生物学背景：基因型与表现型

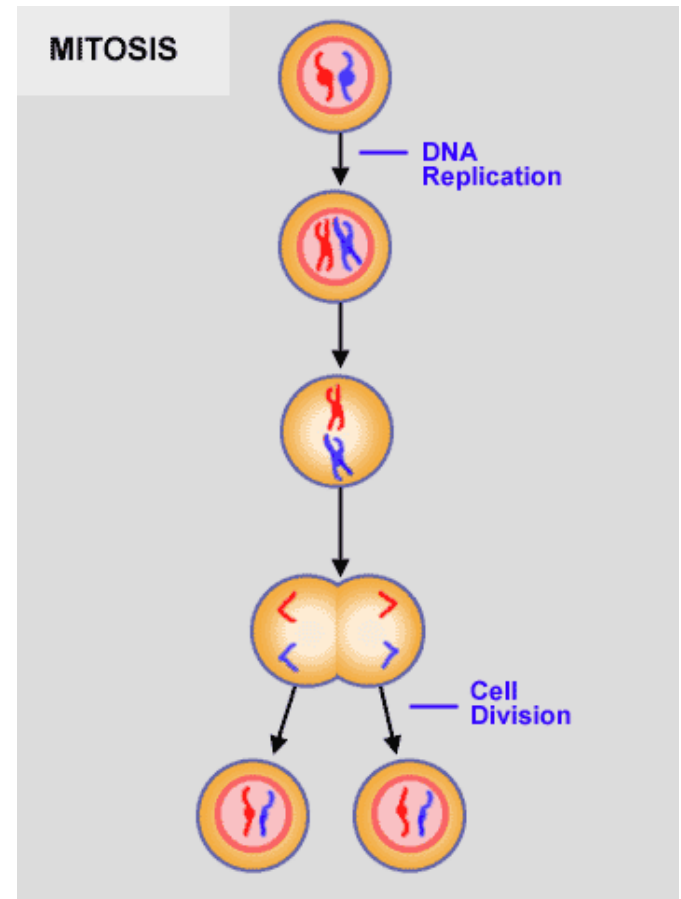
- 基因的全部组合称为基因型
- 基因型决定表现型（眼睛颜色、身高、疾病易感性）
- 表现型会受到底层遗传编码变化的影响



生物学背景

“繁殖”

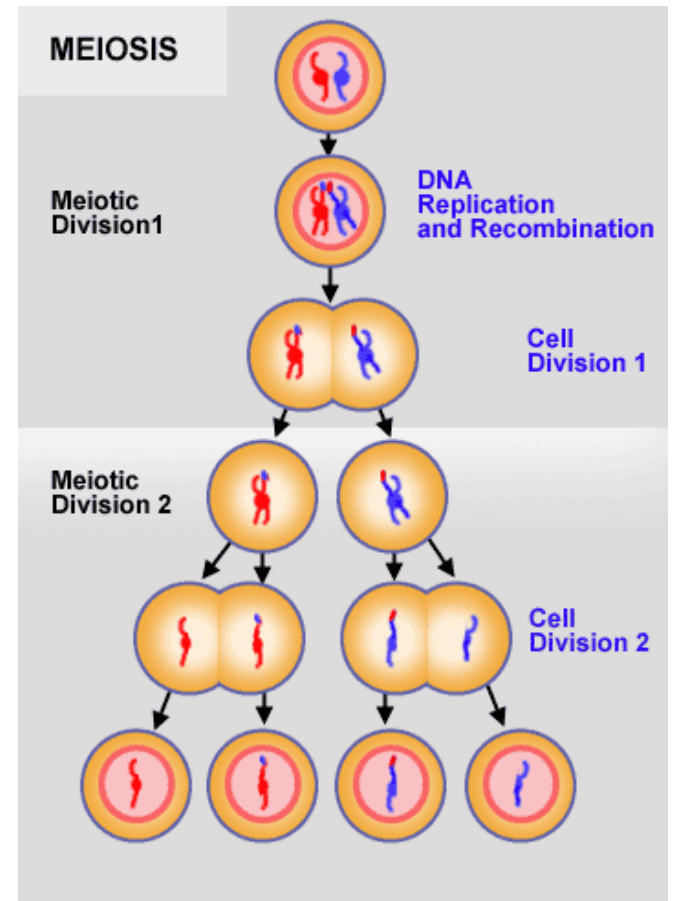
- 遗传信息的复制
 - 有丝分裂
 - 减数分裂
- 有丝分裂是把相同的遗传信息复制给新的后代：不发生信息的交换
- 有丝分裂是多细胞结构（如器官）生长的正常方式。



生物学背景

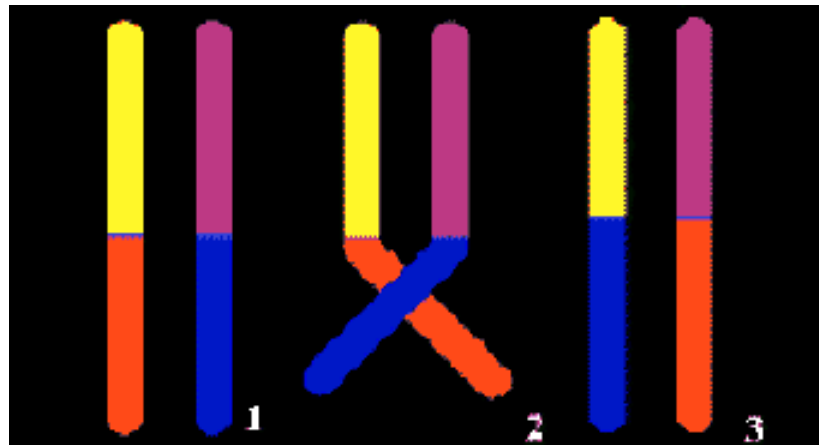
繁殖

- 减数分裂是有性繁殖的基础
- 减数分裂之后产生 2 个配子
- 繁殖时两个配子结合成合子，它将发育成新的个体
- 交叉产生新的基因型



变异

- 任何复制过程都可能出错，因此单点变异相当常见。
- 其他类型的错误也可能发生，包括影响较长区段的错误，例如缺失、倒位、替换等也可能发生。



“自然选择”

- **物种起源：“有利变异的保存与不利变异的摒弃。”**
- **出生的个体多于能够存活的个体，因此存在着持续的生存竞争。**
- **具有优势的个体拥有更大的存活机会：即适者生存。**

遗传算法算子

- 表示方法
- 选择方法
- 繁殖方法

常用表示方法

- 二进制串。
- 整数数组（通常有界）
- 字母数组
-

选择方法

有许多不同的策略可以用来选择被复制到下一代的个体

选择方法

- 轮盘赌选择。
- 精英选择。
- 适应度比例选择。
- 尺度变换选择。
- 排序选择。
-

轮盘赌选择

- 从概念上说，这可以表示为一个轮盘游戏——每个个体分得轮盘上的一个扇区，适应度更高的个体分得的扇区更大。

轮盘赌选择

编号	字符串	适应度	百分比
1	01101	169	14.4
2	11000	576	49.2
3	01000	64	5.5
4	10011	361	30.9
总计		1170	100.0

其他选择方法

- **精英选择：**
每一代只选择适应度最高的成员。
- **截断选择：**
只选择目标函数值高于某个阈值的个体。

繁殖方法

- 主要有两种方法：
 - 交叉
 - 变异

繁殖方法： 交叉

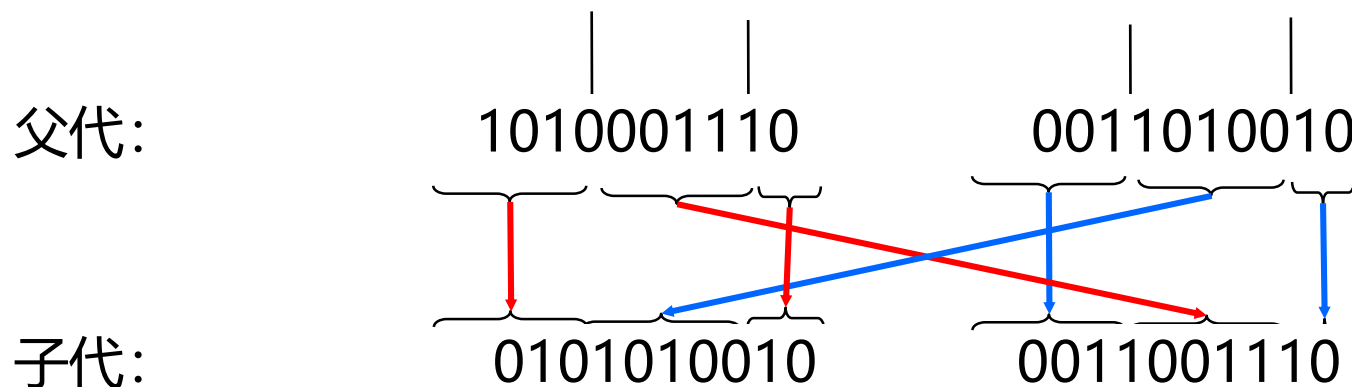
- 两个父代产生两个子代
- 两种做法：
 1. 把两个父代的染色体复制到下一代
 2. 把两个父代随机重组
(交叉) 以形成新的子代

几种可能的交叉 策略

- 随机选择一个点进行交叉
- 多点交叉
- 均匀交叉

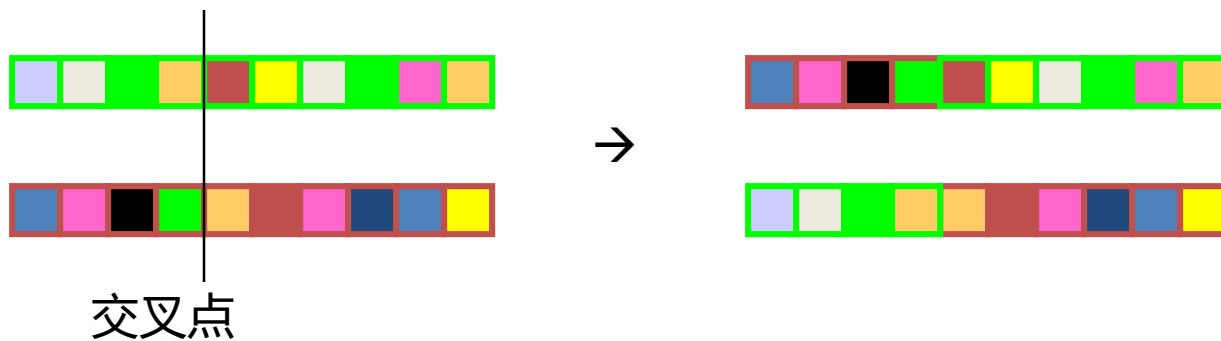
两点交叉

- 避免出现染色体首尾两端的基因总是被拆开的情况

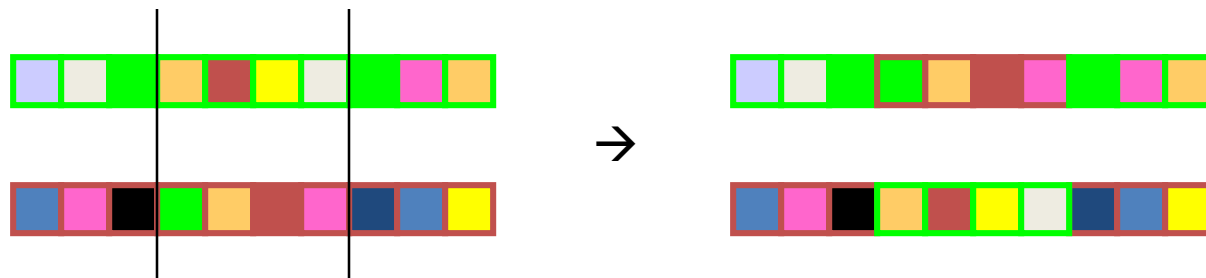


交叉

- 单点交叉



- 两点交叉（多点交叉）



均匀交叉

- 随机选择一个子集
- 该子集取自父代 1，其余各位取自父代 2。

子集：BAABBAABBB

(随机生成)

父代：1010001110

0011010010

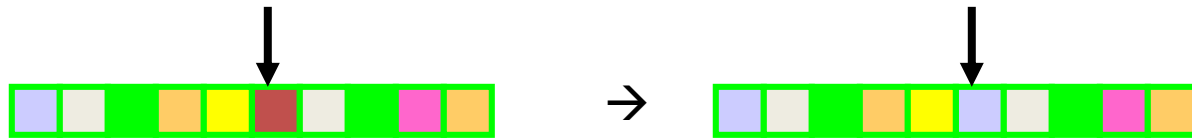
子代：0011001010

1010010110

繁殖方法：

变异

- 由单个父代产生新的子代



一个（稍复杂些的） 例子

旅行商问题：

为给定的城市集合寻找一条回路，使得

- 每个城市只被访问一次**
- 旅行的总距离最小**

表示

**表示是一个有序的城市编号列表，
称为基于顺序的遗传算法。**

1) 伦敦 3) 达尼丁 5) 北京 7) 东京
2) 威尼斯 4) 新加坡 6) 凤凰城 8) 维多利亚

城市列表 1 (3 5 7 2 1 6 4 8)

城市列表 2 (2 5 7 6 8 1 3 4)

交叉

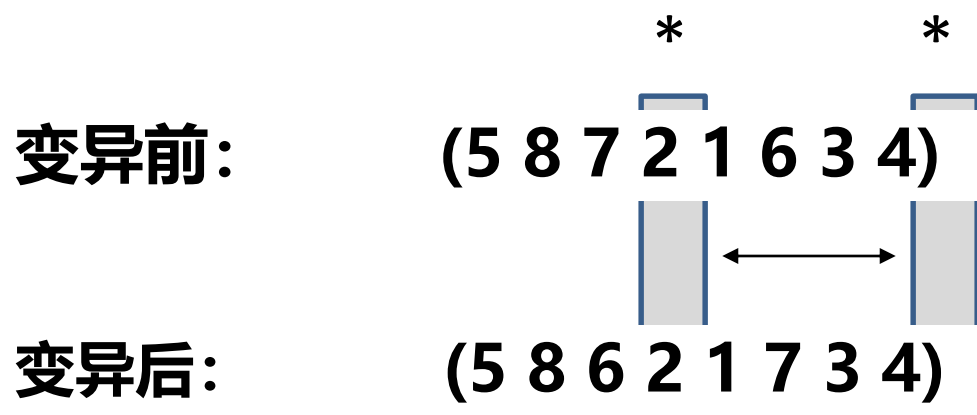
交叉结合了倒位与重组：

		*		*			
父代 1	(3	5	7	2	1	6	4 8)
父代 2	(2	5	7	6	8	1	3 4)
子代	(5	8	7	2	1	6	3 4)

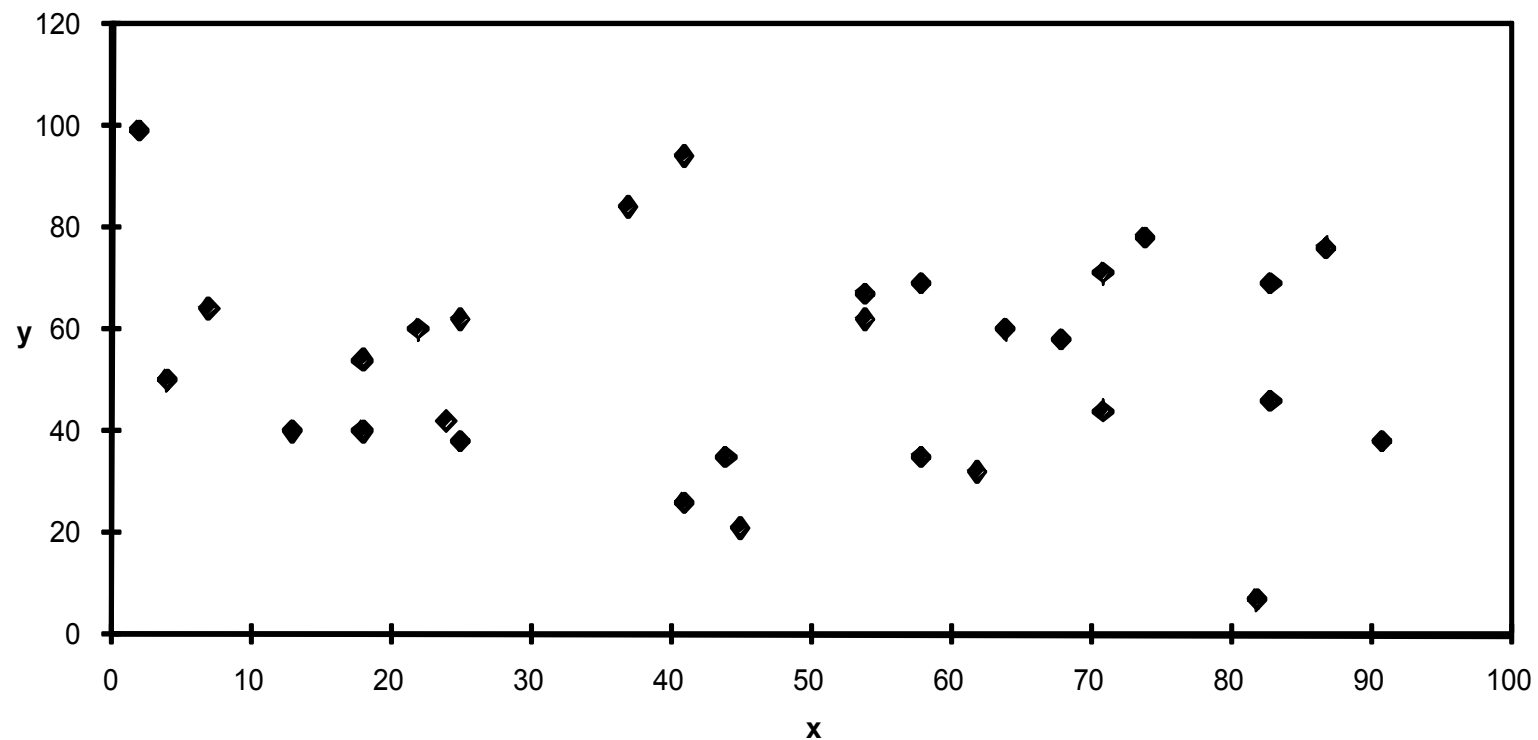
该算子称为 Order1 交叉。

变异

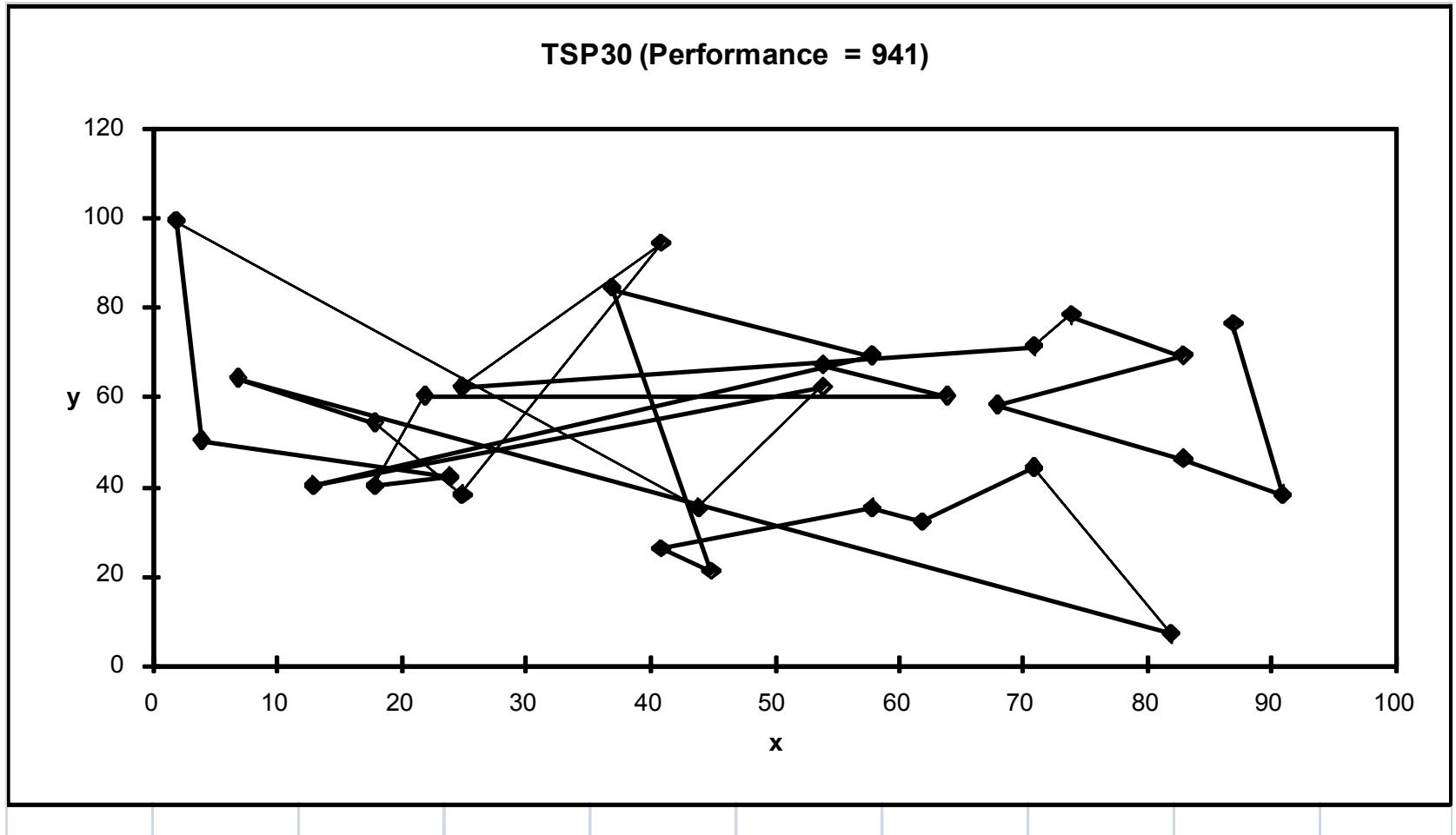
变异就是对列表重新排序：



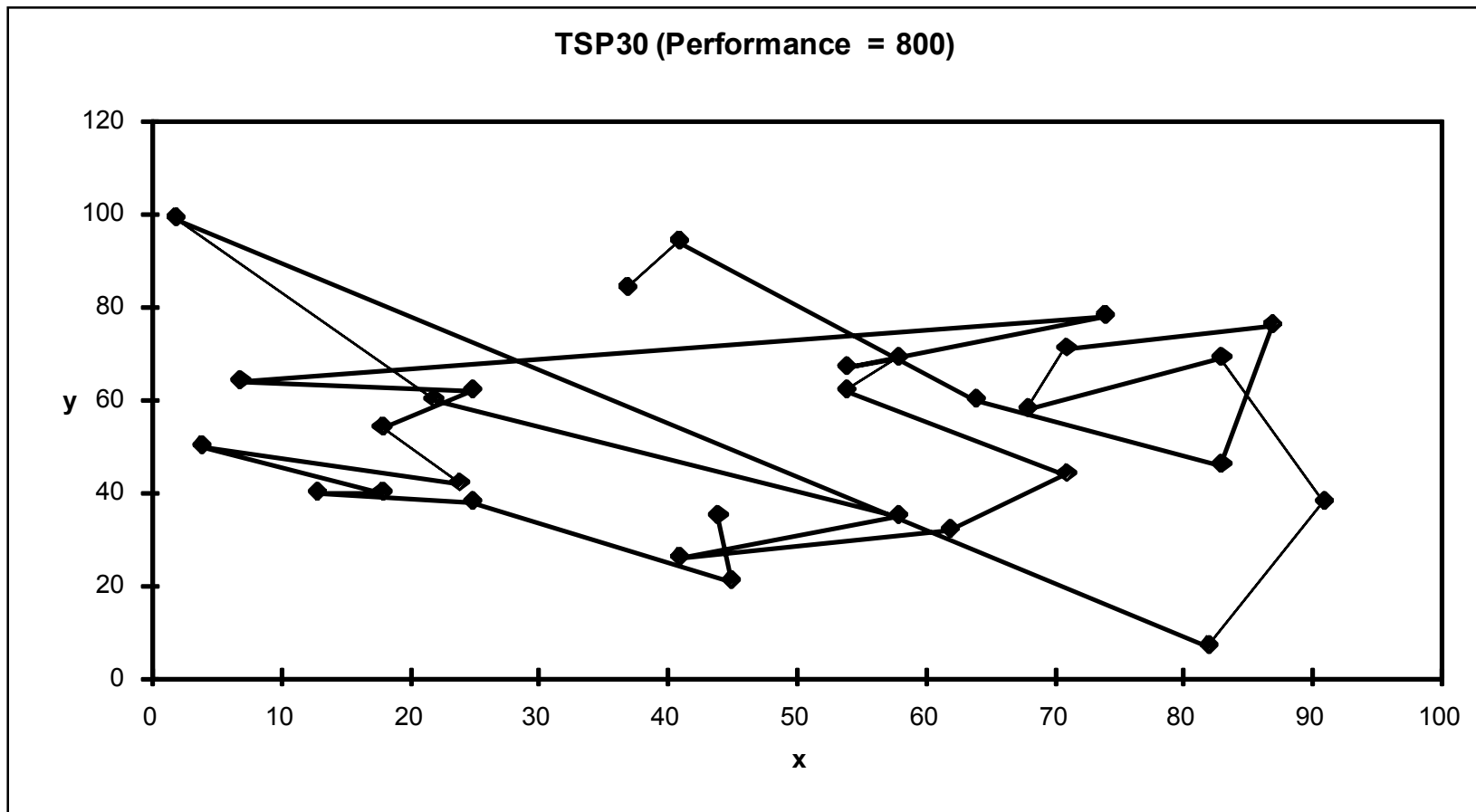
TSP 示例：30 个城市



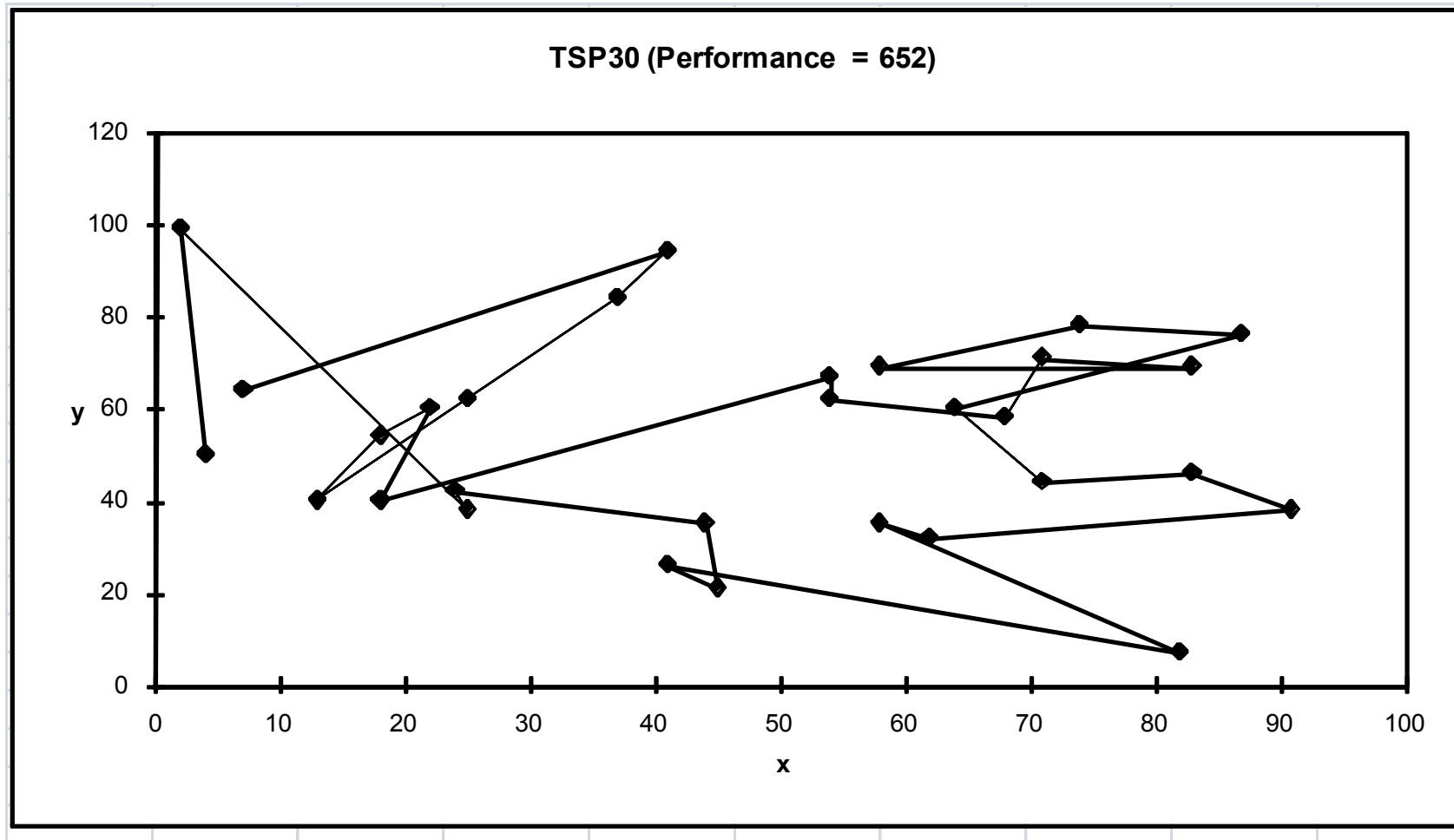
解 (距离 = 941)



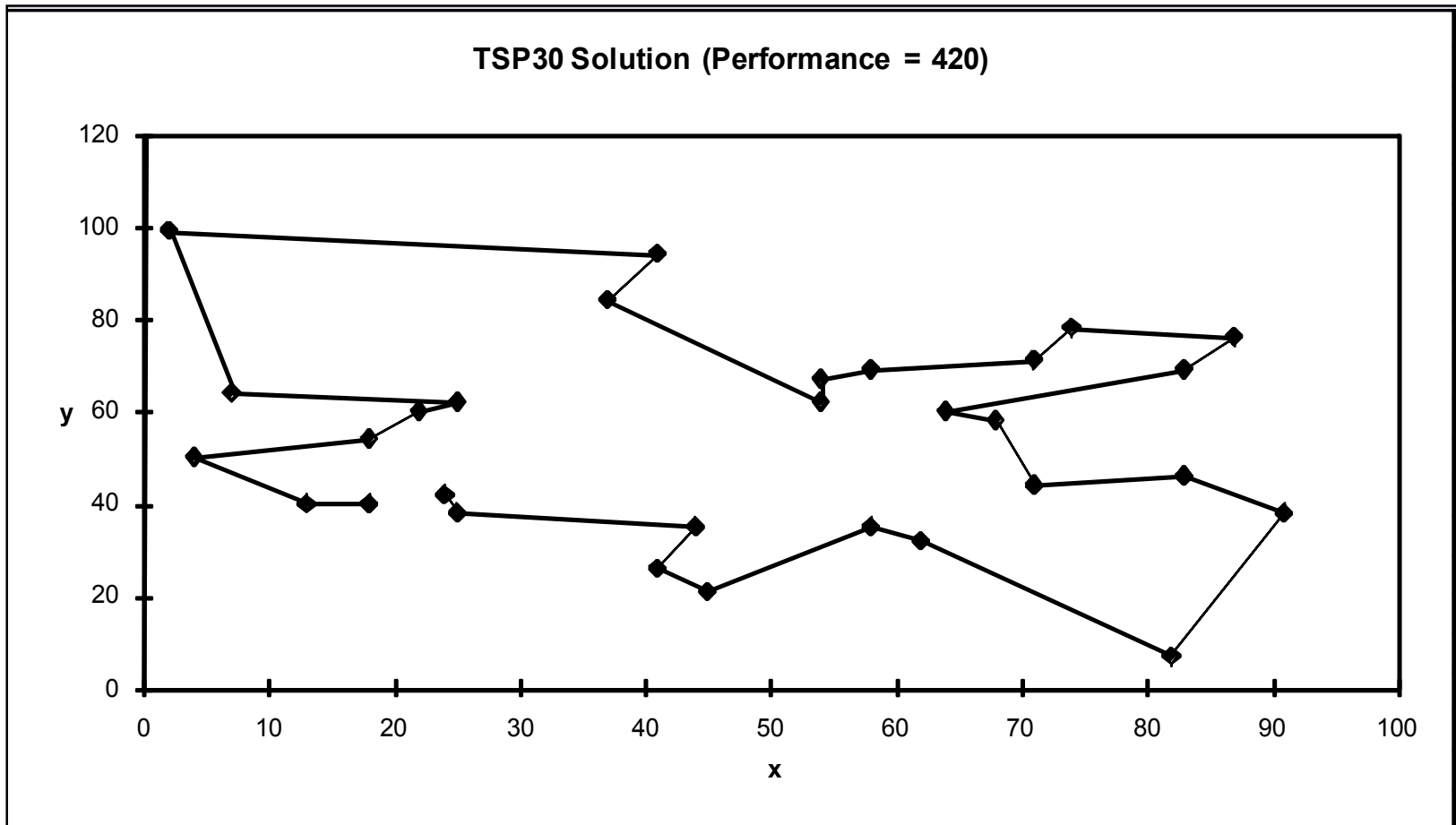
解 (距离 = 800)



解 (距离 = 652)



最优解 (距离 = 420)



遗传算法的应用

领域	应用类型
控制	天然气管道、导弹规避
设计	飞行器设计、键盘布局、通信网络
博弈	扑克、跳棋
安全	加密与解密
机器人	轨迹规划

遗传算法的优点

- 概念易于理解
- 模块化，与应用分离
- 支持多目标优化
- 总是有解；且解会随时间变好。
- 易于利用先前的或备选的解
- 是构建混合应用的灵活积木。

机器人生命体的 自动化设计与制造

生物生命掌控着自身的繁殖方式.....
但这种设计与制造的自主性尚未在人工系统中实现.....本文报告了一种计算与实验相结合的方法的结果：简单的机电系统通过仿真，由基本的构造块（杆件、作动器与人工神经元）演化而来；随后用机器人技术把“最适应”的机器制造出来.....我们由此实现了设计与建造的自主：在“有限宇宙”的物理仿真中演化，并与自动制造相结合。