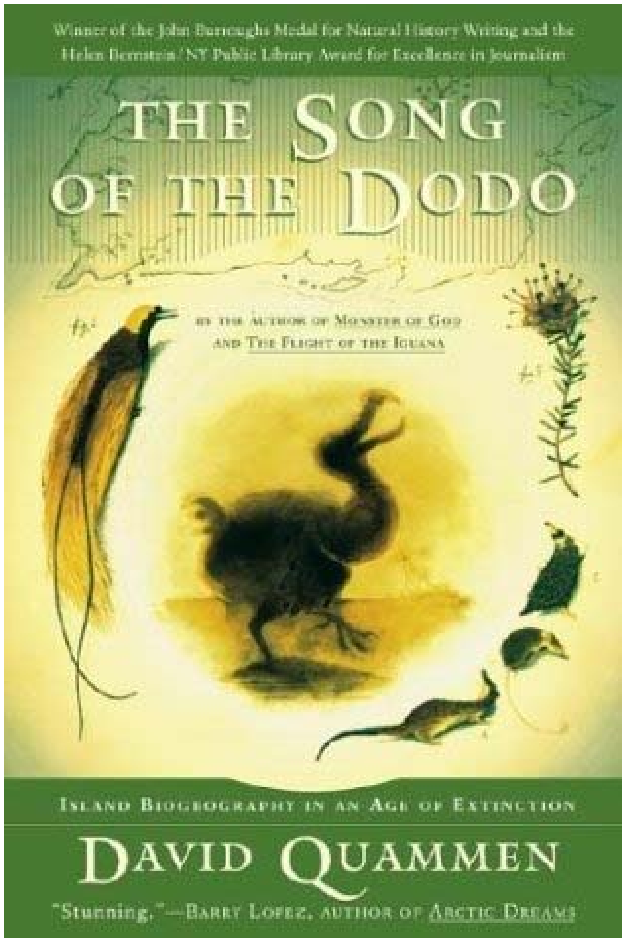
Biogeography-Based Optimization

Dan Simon
Cleveland State University
Fall 2008

Outline

1. Biogeography
2. Optimization
3. Other Population-Based Optimizers
4. Benchmark Functions & Results
5. Sensor Selection & Results
6. Conclusion

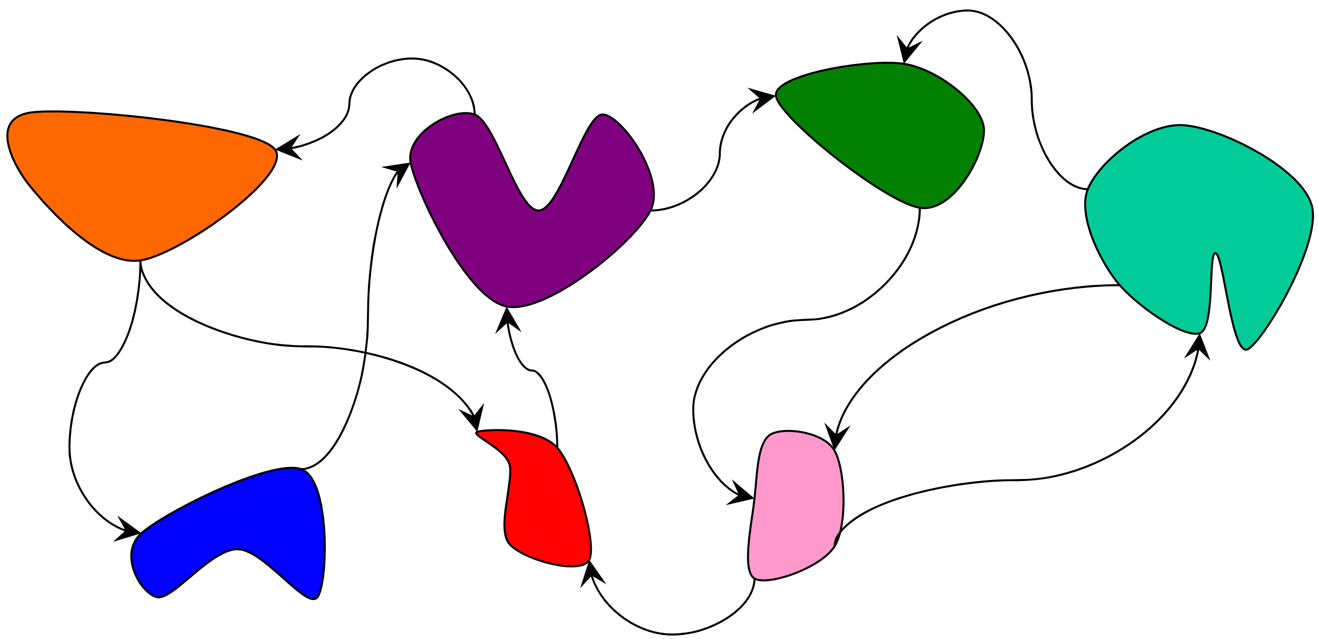
Biogeography



The study of the geographic distribution of biological organisms

- Mauritius
- 1600s

Biogeography



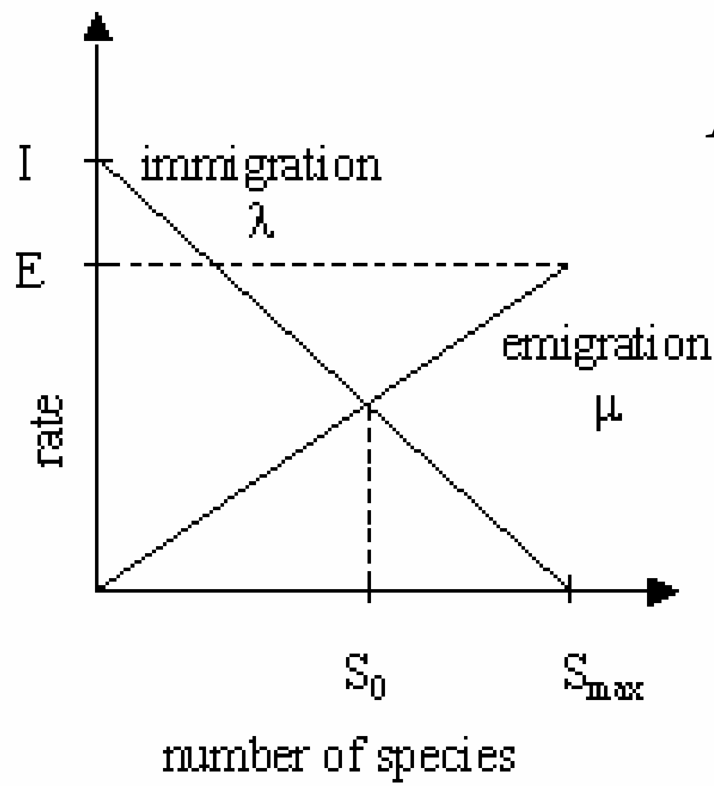
Species migrate between "islands" via flotsam, wind, flying, swimming, …

Biogeography

- Habitat Suitability Index (HSI): Some islands are more suitable for habitation than others

- Suitability Index Variables (SIVs): Habitability is related to features such as rainfall, topography, diversity of vegetation, temperature, etc.

Biogeography



As habitat suitability improves:

- The species count increases
- Emigration increases (more species exit the habitat)
- Immigration decreases (fewer species come into the habitat)

Biogeography

P_s = probability that habitat contains S species

$$\overbrace{}^{S \text{ species at time } t\text{, and no migration occurred}}$$

$$P_s(t+\Delta t) = P_s(t)\overbrace{(1-\lambda_s\Delta t - \mu_s\Delta t)}^{S \text{ species at time } t\text{, and no migration occurred}} + \underbrace{P_{s-1}(t)\lambda_{s-1}\Delta t}_{S-1 \text{ species at time } t\text{, and 1 species immigrated}} + \underbrace{P_{s+1}(t)\mu_{s+1}\Delta t}_{S+1 \text{ species at time } t\text{, and 1 species emmigrated}}$$

Biogeography

Convert the difference equation into a differential equation

$$\dot{P}_s = \begin{cases} -(\lambda_s + \mu_s)P_s + \mu_{s+1}P_{s+1} & S = 0 \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} + \mu_{s+1}P_{s+1} & S = 1, \cdots, S_{\max} - 1 \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} & S = S_{\max} \end{cases}$$

($S_{\max}+1$) coupled differential equations that can be combined into a single matrix equation.

Biogeography

$$\dot{P} = AP$$

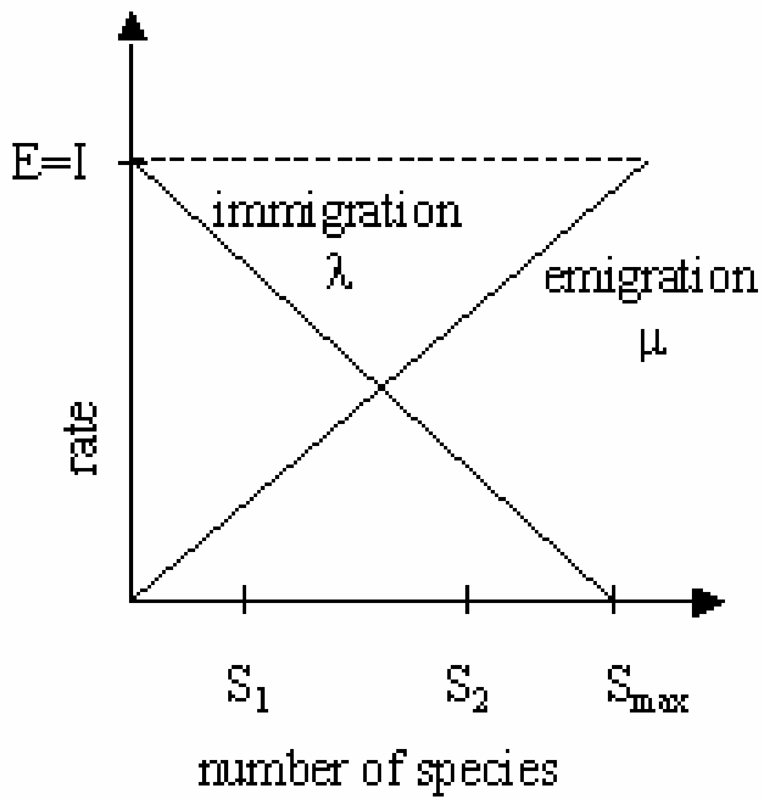
$$A = \begin{bmatrix} -(\lambda_0 + \mu_0) & \mu_1 & 0 & \cdots & 0 \\ \lambda_0 & -(\lambda_1 + \mu_1) & \mu_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \lambda_{n-2} & -(\lambda_{n-1} + \mu_{n-1}) & \mu_n \\ 0 & \cdots & 0 & \lambda_{n-1} & -(\lambda_n + \mu_n) \end{bmatrix}$$

Biogeography

Suppose E=I. Then $\mu_k = Ek/n$

$\lambda_k = E(1-k/n)$

where $k =$ species count, $n = S_{\max}$



Biogeography

$$A = E\begin{bmatrix} -1 & 1/n & 0 & \dots & 0 \\ n/n & -1 & 2/n & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & 2/n & -1 & n/n \\ 0 & \dots & 0 & 1/n & -1 \end{bmatrix}$$

$$= EA'$$

$$\dot{P} = AP$$

So the population reaches equilibrium when P is equal to the eigenvector corresponding to the zero eigenvalue of A

Biogeography

$$P(\infty) = \frac{v}{\sum v_i}$$

$$v = \begin{bmatrix} v_1 & \cdots & v_{n+1} \end{bmatrix}^T$$

$$v_i = \begin{cases} \dfrac{n!}{(n-1-i)!(i-1)!} & (i = 1, \ldots, \text{ceil}((n+1)/2)) \\ v_{n+2-i} & (i = \text{ceil}((n+1)/2)+1, \ldots, n+1) \end{cases}$$

Biogeography-Based Optimization

1. Initialize a set of solutions to a problem.
2. Compute "fitness" (HSI) for each solution.
3. Compute S, λ, and μ for each solution.
4. Modify habitats (migration) based on λ, μ.
5. Mutatation based on probability.
6. Typically we implement elitism.
7. Go to step 2 for the next iteration if needed.

Benchmark Functions

14 standard benchmark functions were used to evaluate BBO relative to other optimizers.

- Ackley
- Fletcher-Powell
- Griewank
- Penalty Function #1
- Penalty Function #2
- Quartic
- Rastrigin
- Rosenbrock
- Schwefel 1.2
- Schwefel 2.21
- Schwefel 2.22
- Schwefel 2.26
- Sphere
- Step

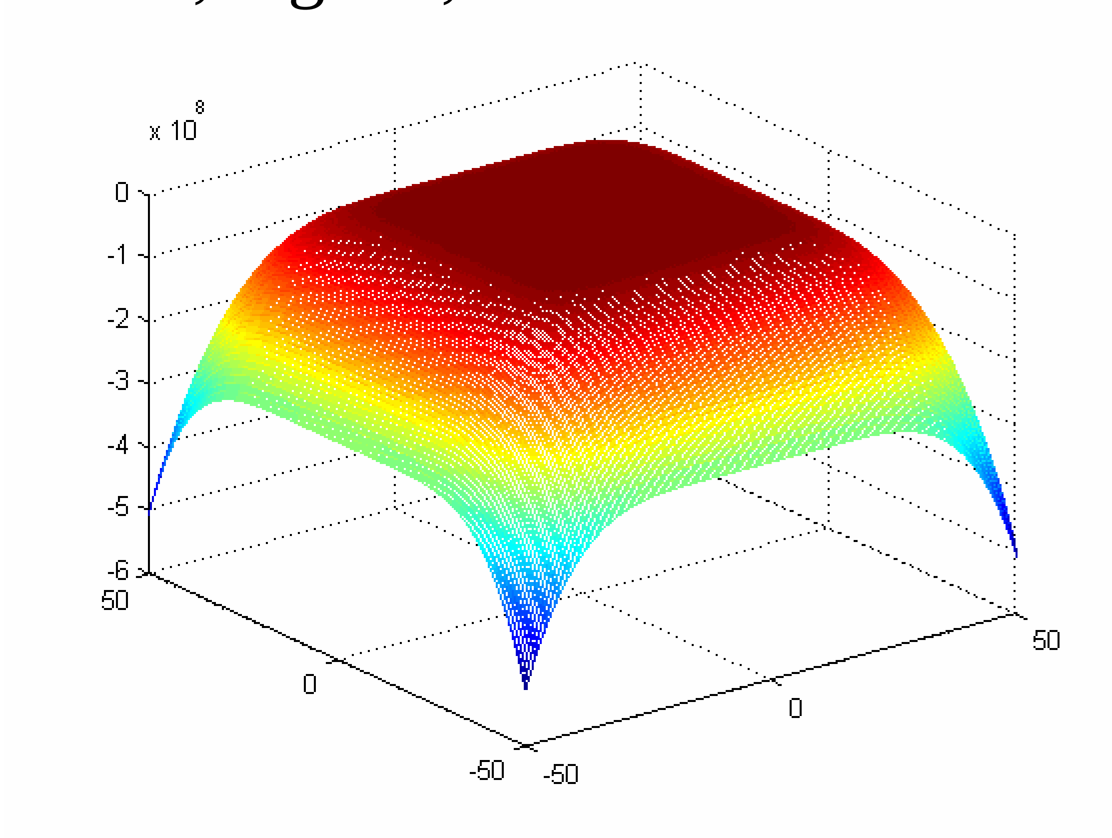
Benchmark Functions

Functions can be categorized as

- Separable or nonseparable – for example, $(x+y)$ vs. xy
- Regular or irregular – for example, $\sin x$ vs. $\mathrm{abs}(x)$
- Unimodal or multimodal – for example, x^2 vs. $\cos x$

Benchmark Functions

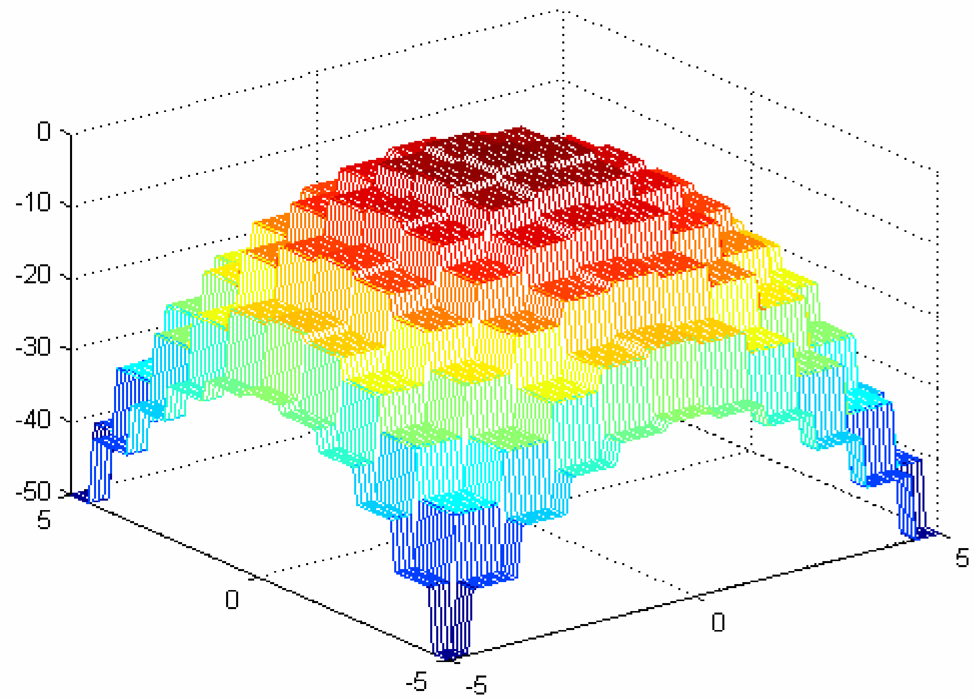
Penalty function #1:
nonseparable, regular, unimodal



Benchmark Functions

Step function: $$f(x) = \sum_{i=1}^{n} [\text{floor}(x_i + 0.5)]^2$$

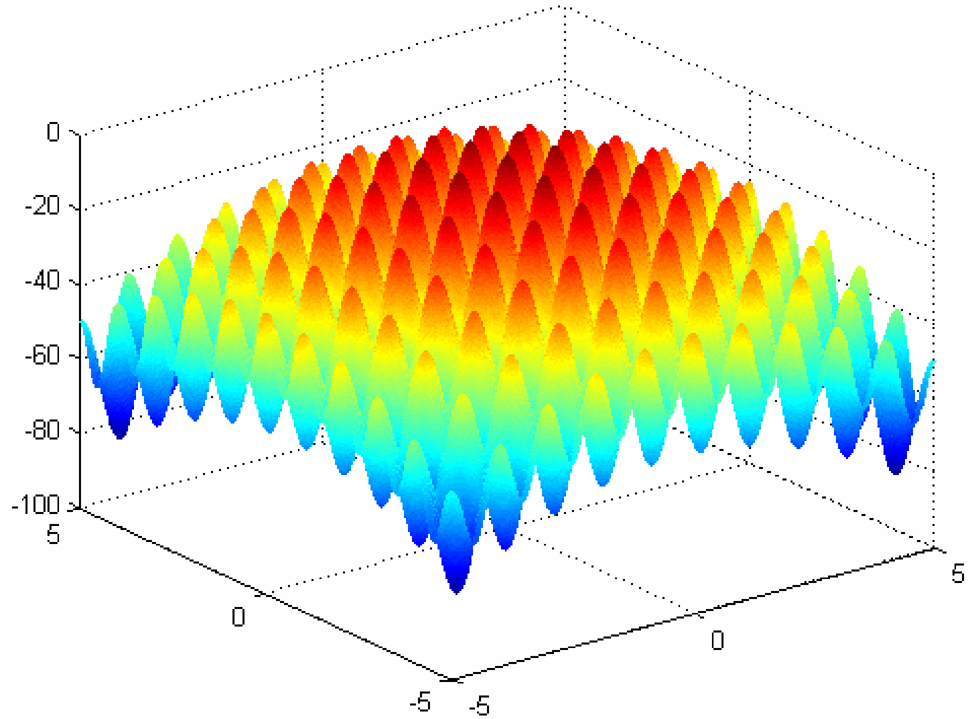
separable, irregular, unimodal



Benchmark Functions

Rastrigin: $f(x) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)]$

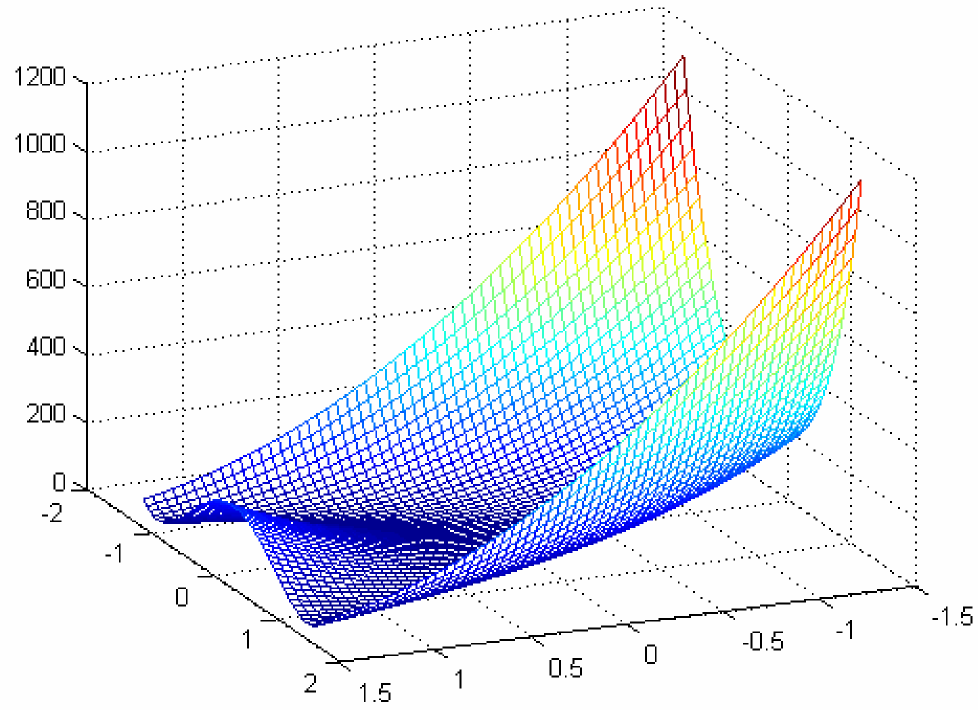
nonseparable, regular, multimodal



Benchmark Functions

Rosenbrock: $f(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$

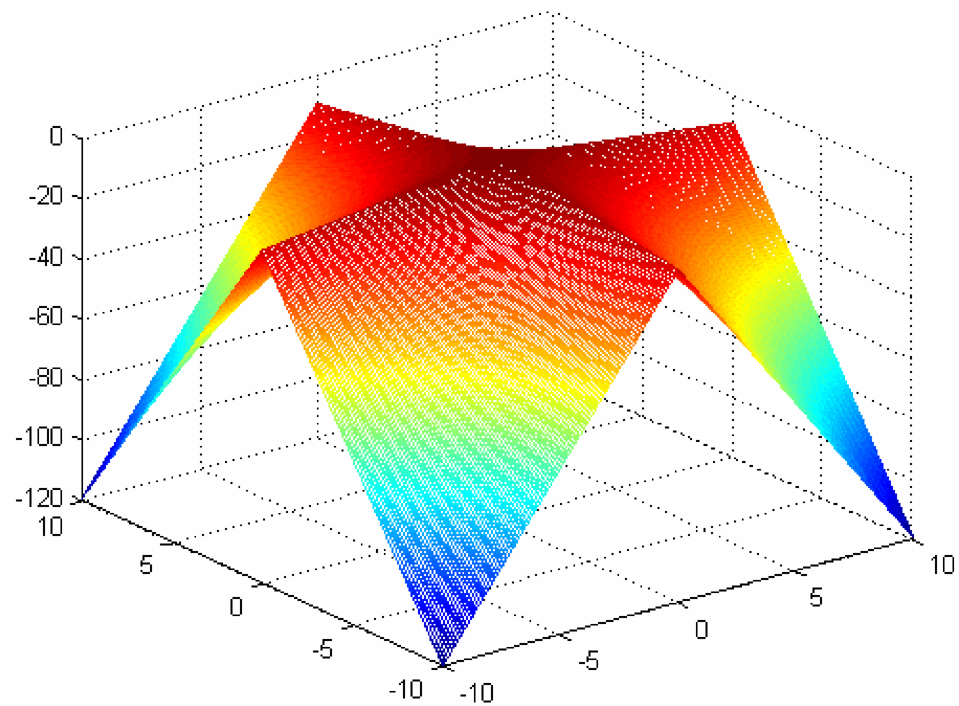
nonseparable, regular, unimodal



Benchmark Functions

Schwefel 2.22: $f(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|$

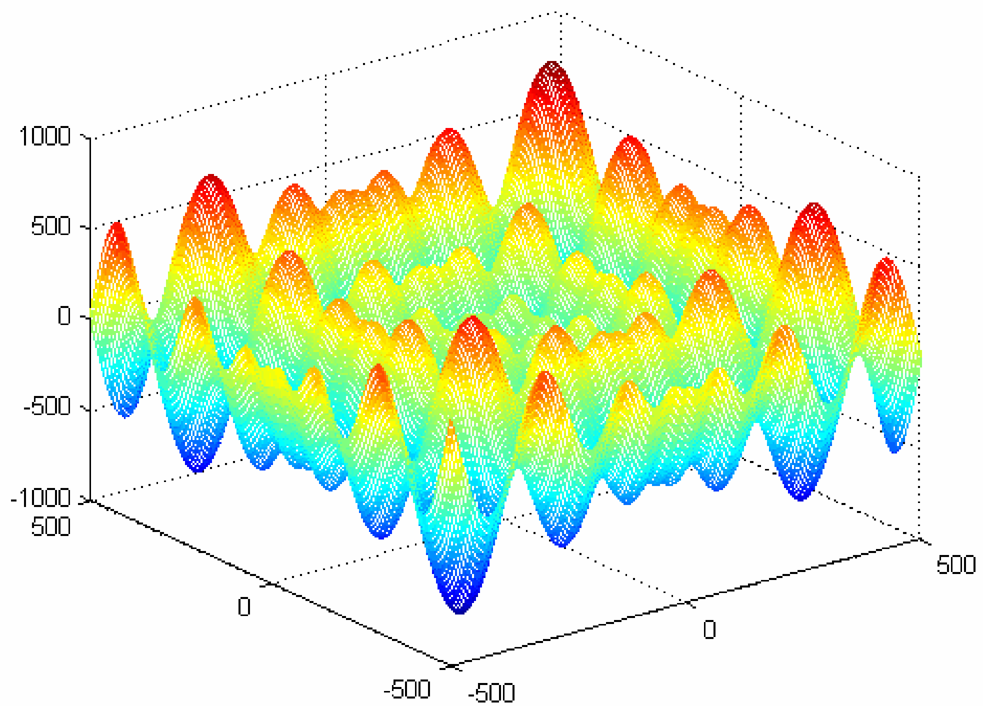
nonseparable, irregular, unimodal



Benchmark Functions

Schwefel 2.26: $f(x) = -\sum_{i=1}^{n} x_i \sin\sqrt{|x_i|}$

separable, irregular, multimodal



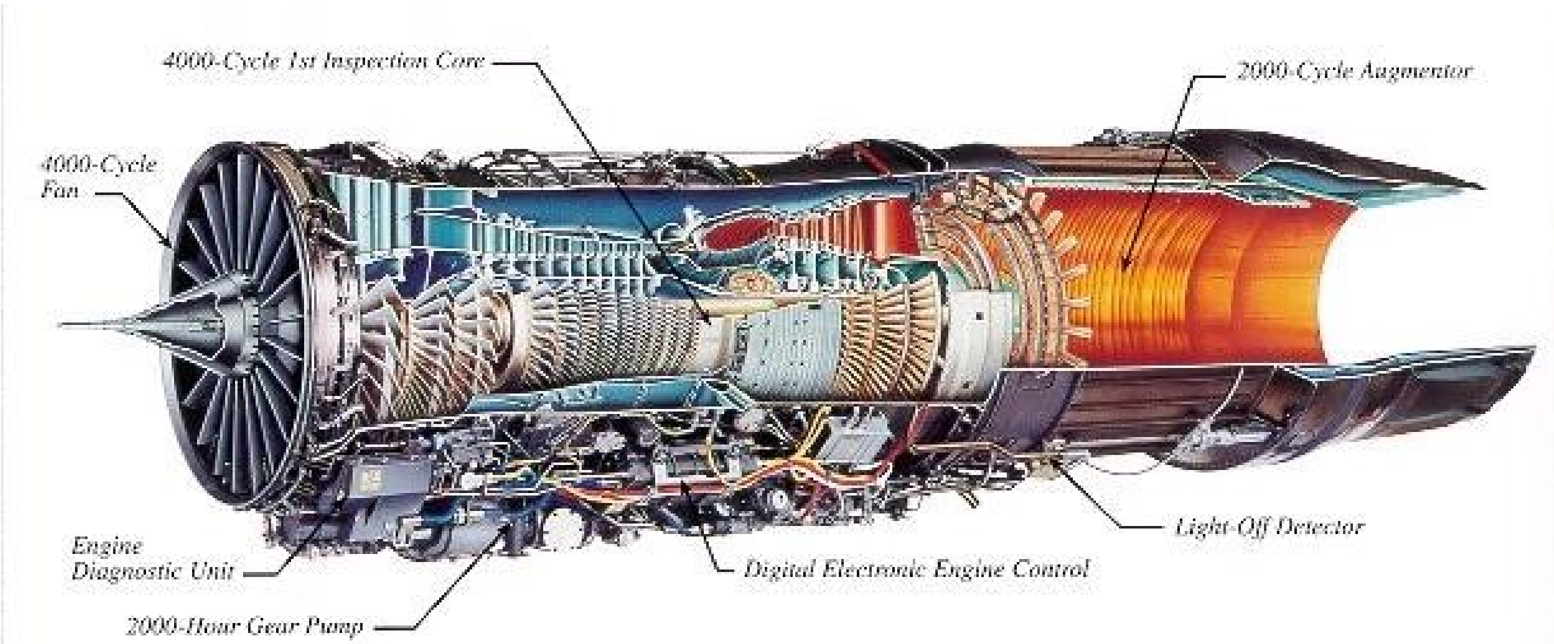
Optimization Algorithms

- Ant colony optimization (ACO)
- Biogeography-based optimization (BBO)
- Differential evolution (DE)
- Evolutionary strategy (ES)
- Genetic algorithm (GA)
- Population-based incremental learning (PBIL)
- Particle swarm optimization (PSO)
- Stud genetic algorithm (SGA)

	ACO	*BBO*	*DE*	*ES*	*GA*	*PBIL*	*PSO*	*SGA*
Ackley	182	**100**	146	197	197	232	192	103
Fletcher	1013	**100**	385	494	415	917	799	114
Griewank	162	117	272	696	516	2831	1023	**100**
Penalty 1	2.2E7	1.2E4	9.7E4	1.3E6	2.5E5	2.8E7	2.1E6	**100**
Penalty 2	5.0E5	715	5862	4.2E4	1.1E4	5.4E5	6.4E4	**100**
Quartic	3213	262	1176	7008	2850	4.8E4	8570	**100**
Rastrigin	454	**100**	397	536	421	634	470	134
Rosenbrock	1711	102	253	716	428	1861	516	**100**
Schwefel 1.2	202	**100**	391	425	166	606	592	110
Schwefel 2.21	161	**100**	227	162	184	265	179	146
Schwefel 2.22	688	**100**	290	1094	500	861	665	142
Schwefel 2.26	108	118	137	140	142	177	142	**100**
Sphere	1347	**100**	250	910	906	2785	1000	109
Step	248	112	302	813	551	3271	1161	**100**

Average performance of 100 simulations (n = 50)

Aircraft Engine Sensor Selection



Health estimation

- Better maintenance
- Better control performance

Aircraft Engine Sensor Selection

What sensors should we use?

- Measure pressures, temperatures, speeds
- A total of 11 sensors; some can be duplicated
- Estimate efficiencies and airflow capacities
- Optimize a combination of estimation accuracy and financial cost
- Use a Kalman filter for health estimation

Aircraft Engine Sensor Selection

Suppose we want to pick *N* objects out of *K* classes while choosing from each class no more than *M* times.

Example: We have red balls, blue balls, and green balls *(K=3)*. We want to pick 4 balls *(N=4)* with each color chosen no more than twice *(M=2)*.

6 Possibilities: {B, B, G, G}, {R, B, G, G}, {R, B, B, G}, {R, R, G, G}, {R, R, B, G}, {R, R, B, B}

Aircraft Engine Sensor Selection

Pick N objects out of K classes while choosing from each class no more than M times.

$$q(x) = (1 + x + x^2 + \ldots + x^M)^K$$
$$= 1 + q_1 x + q_2 x^2 + \ldots + x^{MK}$$

Multinomial theorem: The number of unique combinations is equal to q_N (order independent)

Aircraft Engine Sensor Selection

Pick *20* objects out of *11* classes while choosing from each class no more than *4* times.

$$q(x) = (1 + x + x^2 + x^3 + x^4)^{11}$$

$$= 1 + \ldots + 3{,}755{,}070\, x^{20} + \ldots$$

21 hours of CPU time for an exhaustive search.

So we need a quick suboptimal search strategy.

Aircraft Engine Sensor Selection

	ACO	BBO	DE	ES	GA	PBIL	PSO	SGA
Mean	8.22	**8.01**	8.06	8.15	8.04	8.18	8.14	8.02
Best	8.12	**7.19**	7.60	8.05	8.02	8.80	8.06	8.02

Average and best performance over 100 Monte Carlo simulations. Computational savings = 99.99% (21 hours $\rightarrow$ 8 seconds).

Conclusion

- A new biologically-motivated optimizer
- Paper and Matlab code is at http://academic.csuohio.edu/simond/bbo

Future Work

- Applications
- Convergence
- Dynamic/noisy fitness functions
- Extinction is not the same as emigration
- Take island proximity into account
- Model species populations (demographics)
- Species age affects extinction and emigration

Future Work

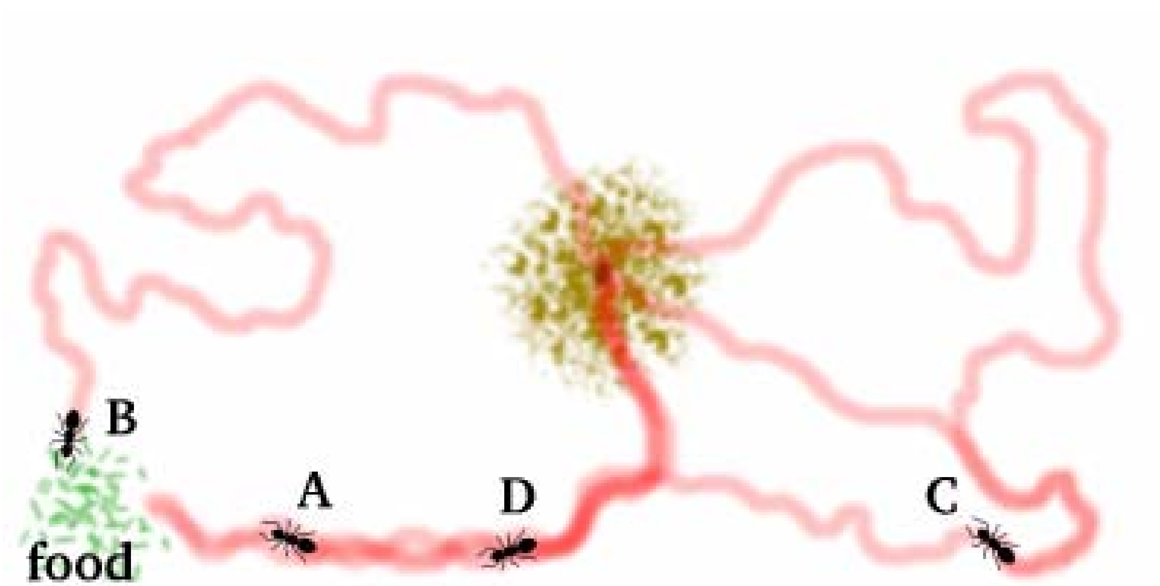
- Migration curves are probably convex, and vary between islands
- Continuous BBO
- Connections between BBO and other Eas
- Migration varies with species mobility
- Predator/prey relationships
- Population sizes
- Constrained BBO
- Multiobjective BBO

Future Work

- Number of islands
- Combination with other EAs
  - Tabu search, particle swarm optimization, use of global information, etc.
  - Local memory of past performance for each island
  - Fuzzy fitness functions

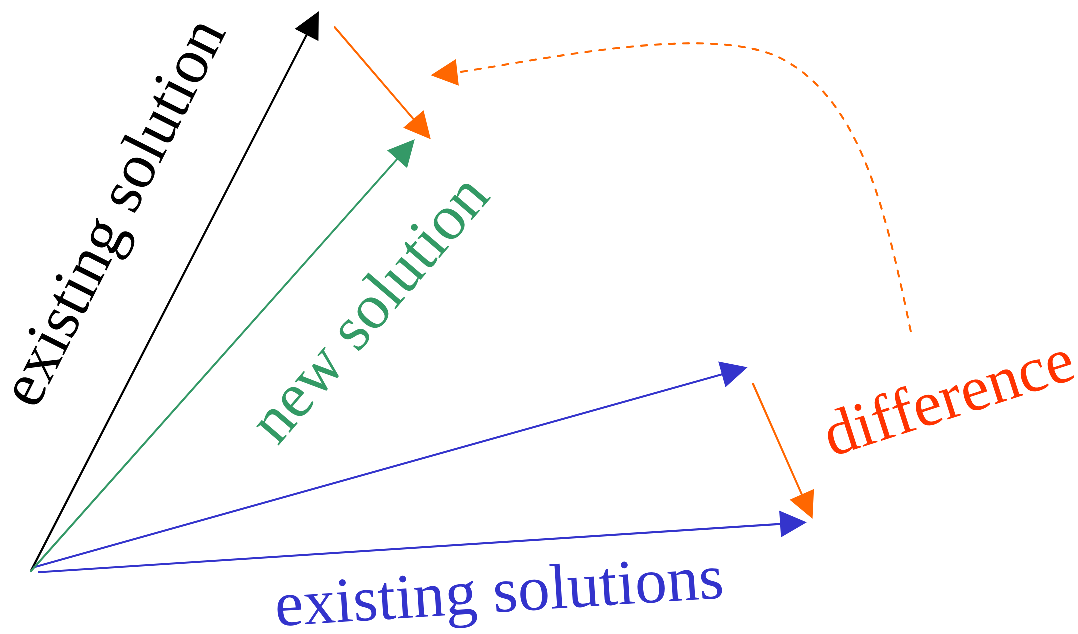
Other optimization methods

Ant colony optimization: Based on the pheromone deposition of ants



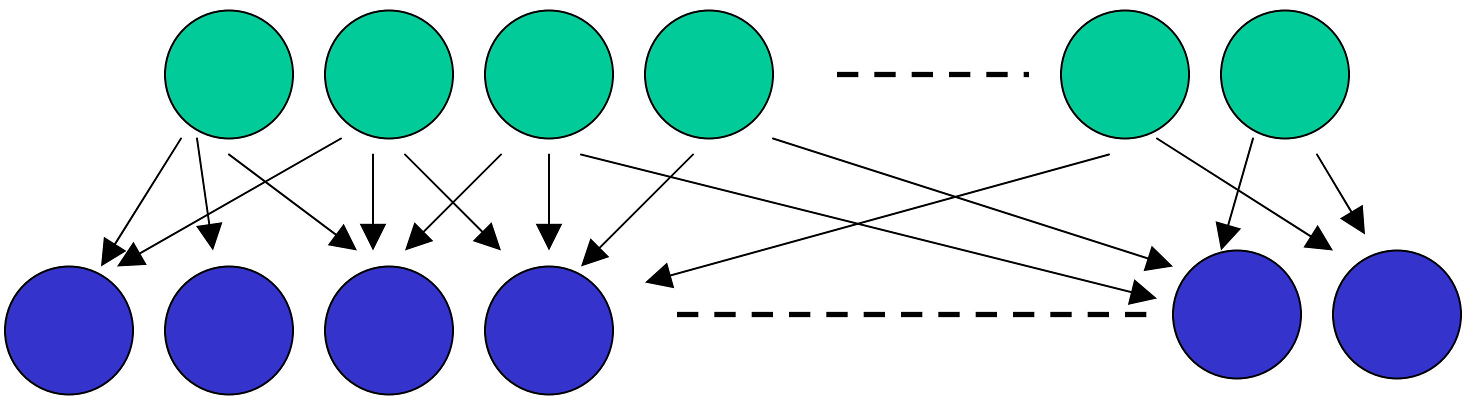
Other optimization methods

Differential evolution: Based on simple difference-based modification of solutions



Other optimization methods

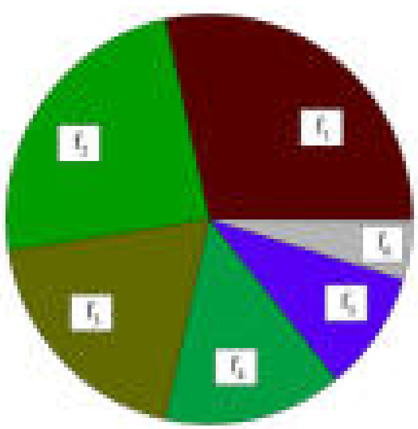
Evolutionary Strategy: Based on multiple parents contributing to offspring

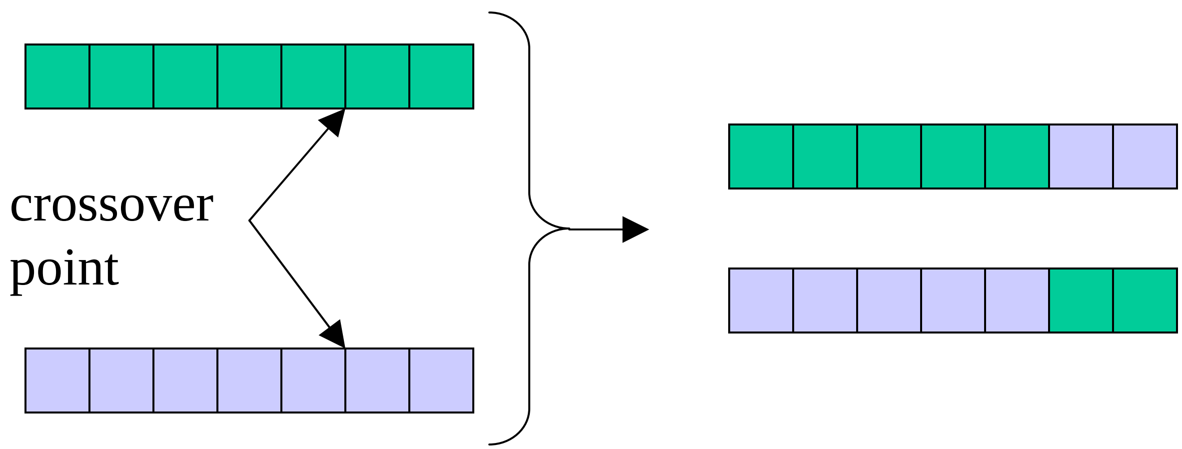


The most fit offspring become the next generation of parents

Other optimization methods

Genetic algorithms: Based on natural selection in biological evolution





selection **parents**

children

Other optimization methods

Particle swarm optimization: Based on the swarming behavior of birds, fish, etc.

Each particle (solution) learns from its neighbors and adjusts itself accordingly