Genetic Algorithm

- Optimization
- Algorithm
- Dominance and Nondominated Sorting
- Crowding Distance

Application on Lattice Optimization

- $\epsilon, \beta_x$
- $\epsilon, \text{Low-high } \beta_x$

Reference
Motivation: Magnetic Lattice Optimization on ALS

We want to optimize brightness:

1. Emittance $\epsilon_x$.
2. Match $\beta_x$ to ID(insertion device).

by tuning

- Quad strength: $k_{QF}, k_{QD}$.

Constraints:

- $|\text{Tr}(M)| \leq 2$.
- $J_x > 0, J_E > 0$. 

$\nu_x = 1.19$, $\nu_y = 0.77$, $\epsilon = 3.10$ [nm]
There are many ways ...
There are many ways ...

- Deterministic, $\epsilon(k_{QF}, k_{QD}), \epsilon'_k$
- Brute force, $(k_{QF}, k_{QD})$
There are many ways ...

- Deterministic, $\epsilon(k_{QF}, k_{QD}), \epsilon'_k$
- Brute force, $(k_{QF}, k_{QD})$
- Stochastic, Evolutionary
GA has been used on DC gun photoinjector [Bazarov and Sinclair, 2005].

- Population based.
- Iterative (generation).
- Red: violate the constraints.
- Green: meet the constraints.

$\left(k_{QF}, k_{QD}\right)$. 
Multi-Objective Optimization

Airline tickets for 2012 London Olympics

Cost $ \uparrow$

Time $h \downarrow$

Best
Not the best
So far the best

- Time ?
- Cost ?
- Weighted sum ?
- Whole picture.
Multi-Objective Optimization

Airline tickets for 2012 London Olympics

- AA1
- UA1
- NW1
- BA1
- UA2

- Time?
- Cost?
- Weighted sum?
- Whole picture.
Objective Space

- **Generation**: $f: 0, 10, 76$; $x: 100$.
- **Red**: violate the constraints
- **Green**: meet the constraints
- **Blue**: Pareto optimal set
The general form of an optimization problem is:

\[
\begin{align*}
\text{Minimize/Maximize} & \quad f_m(x), \quad m = 1, 2, \ldots, M; \\
\text{subject to} & \quad g_j \geq 0, \quad j = 1, 2, \ldots, J; \\
& \quad h_k(x) = 0, \quad k = 1, 2, \ldots, K; \\
& \quad x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, \ldots, N;
\end{align*}
\] (1)

MOGA (Multi-Objective Genetic Algorithm):

- Multiobjective, instead of single objective optimization of a weighted sum.
- Constraint.
Genetic Algorithm

Structure of MOGA/GA (Genetic Algorithm)

GA mimics the evolution of nature:

1. **Crossover**: generate children from parents.
2. **Mutation**: change the children.
3. **Nature select**: keep only certain number of population.

MOGA (Multi-Objective Genetic Algorithm)

1. Initialize population (first generation, random)
2. repeat
3. crossover: 2 parents → 2 children.
4. mutation: change children.
5. calculate \( f_m \)
6. nature select: “sorting”
7. until stop (reach maximum generation, find solution, ...)

Global Optimization of a Magnetic Lattice using Genetic Algorithms  
Lingyun Yang  
September 3, 2008  
9 / 21
Our optimization problem [Yang et al., 2008, Robin et al., 2008]:

- **Optimize:**
  1. Emittance $\epsilon$.
  2. $\min(\lvert\beta_x - 1.0\rvert)$.

- **Constraint:**
  - $|\text{Tr}(M_x)| \leq 2$, $|\text{Tr}(M_y)| \leq 2$
  - $\max(\beta_x) \leq 30$, $\max(\beta_y) \leq 30$
  - $\max(\eta_x) \leq 0.4$

- **Parameters:**
  1. QF, QD, QFA in one cell.

1. Evolution of objective functions. $\epsilon$, $|\beta_x - 1|$

2. $(k_{QF}, k_{QD})$, $(k_{QF}, k_{QFA})$, $(k_{QD}, k_{QFA})$
GA: Initialization

You have a lot of freedom to create the first generation.

- No filter: keep everyone, no need to calculate $f_m$ or $g_i$.
- Apply filter:
  - Check $f_m$.
  - Check $g_i$.

![Graph showing distribution of points in a genetic algorithm context]
**GA: Crossover**

Generate children from parents: $x^{(*,t)} \rightarrow x^{(*,t+1)}$. Parents are randomly chosen and used only once. ($t$ – generation)

There are simple ones:

- Middle point. e.g. $0.5(x^{(1,t)} + x^{(2,t)})$
- Blend(BLX), $(1 - \gamma)x^{(1,t)} + \gamma x^{(2,t)}$. $\gamma$ has random property, and extend certain range beyond $[0, 1]$.

More complicate ones:

1. Upper/Lower limit of variables. $[x^{(L)}, x^{(U)}]$
2. Continuous probability distribution. $P(x)$. 
GA: Crossover

Children are generated around two parents in certain probability.

- $x \in [-3, 5]$
- We choose polynomial PDF.
- Boundary is automatic considered.
- 2 parents to 2 children for every dimension of parameter space.
- $\eta_c$ to control the shape of PDF.
Genetic Algorithm

GA: Mutation

Purpose: keep diversity. For each individual:

1. Random, e.g. \( x^{(1,t+1)} = x^{(1,t+1)} + (r - 0.5)\Delta. \)

2. Non-Uniform, e.g.

\[
x^{(1,t+1)} = x^{(1,t+1)} + \tau(x(U) - x(L))(1 - r^{(1-t/t_{max})^b})
\]

3. Normally Distributed, \( x^{(1,t+1)} = x^{(1,t+1)} + N(0, \sigma). \)

More complicate ones will consider:

1. Boundary
2. Probability
GA: Mutation

The new value due to the mutation also follows certain distribution.

- Polynomial PDF.
- Equal probability go left or right.
- Boundaries are considered.
- $\eta_m$ can control the shape of PDF.
Calculate $f_m$, merge parents and children

- Calculate objective function, here lattice properties.
  - Stable/unstable: betatron resonance ($|\text{Tr}(M)|, J_E, J_x$).
  - Constraint: $\beta_x, \beta_y, \eta_x$.
- **Elite-preserving**: merge parents and children, no difference.
  - The population number are fixed.
  - Good parents are kept.
  - Never went worse from generation to generation.

In order to pick the better ones, in multiobjective case, we use **nondominated sorting** to sort the whole population, and keep only the top half. (Airline ticket example).
**Dominance**

- **Domination** [Deb, 2001]. \( f^{(1)} \prec f^{(2)} \) (dominate, precede)
  1. The solution \( f_i^{(1)} \) is no worse than \( f_i^{(2)} \) in all \( m \)-objectives.
  2. The solution \( f^{(1)} \) is strictly better than \( f^{(2)} \) in at least one objective.

Given \( f, a_0, a_1, a_2, b_1, c_1 \) in a 2-dimensional objective space:

- \( a_1 \prec a_2 \), \( a_1 \) is "better" than \( a_2 \).
- \( a_0 \prec a_1 \)
- \( a_1, b_1 \) and \( c_1 \) are not dominated by each other.
**Dominance**

- **Domination** [Deb, 2001]. $f^{(1)} \prec f^{(2)}$ (dominate, precede)
  1. The solution $f_i^{(1)}$ is no worse than $f_i^{(2)}$ in all $m$-objectives.
  2. The solution $f^{(1)}$ is strictly better than $f^{(2)}$ in at least one objective.

- $a_1 \prec a_2$, $a_1$ is “better” than $a_2$.
- $a_0 \prec a_1$
- $a_1, b_1$ and $c_1$ are not dominated by each other.
3 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m$

- **Red**: violate the constraints, or no physical solution.
- **Green**: meet the constraints.
- **Blue**: Pareto optimal set, the best solutions so far.

**Generation 19**
3 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m$

- **Red**: violate the constraints, or no physical solution.
- **Green**: meet the constraints.
- **Blue**: Pareto optimal set, the best solutions so far.

1. Generation 19
2. Generation 46

![Graph showing generation 46 and 19 with points in red, green, and blue colors.]
3 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m$

- **Red**: violate the constraints, or no physical solution.
- **Green**: meet the constraints.
- **Blue**: Pareto optimal set, the best solutions so far.

1. **Generation 19**
2. **Generation 46**
3. **Generation 66**
3 Parameters, optimize \( \epsilon \) and \( \beta_x \rightarrow 1m \)

- **Red**: violate the constraints, or no physical solution.
- **Green**: meet the constraints.
- **Blue**: Pareto optimal set, the best solutions so far.

1. Generation 19
2. Generation 46
3. Generation 66
4. Generation 130
Application on Lattice Optimization

1. up left, small $\epsilon$. 

$\nu_x = 1.38, \ \nu_y = 0.23, \ \epsilon = 1.46 \ [nm]$
1. up left, small $\epsilon$.
2. down left, $\beta_x$ and $\epsilon$. 

$$\nu_x = 1.76, \quad \nu_y = 0.62, \quad \epsilon = 1.67 \text{ [nm]}$$
1. up left, small $\epsilon$.
2. down left, $\beta_x$ and $\epsilon$.
3. down right, small $|\beta_x - 1|$.
6 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m/10m$

- Pareto optimal set.
6 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m/10m$

1. Pareto optimal set.
2. Twiss

$\nu_x=2.90, \quad \nu_y=1.38, \quad \epsilon=3.15$ [nm]
6 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m/10m$

1. Pareto optimal set.
2. Twiss
3. Twiss
6 Parameters, optimize $\epsilon$ and $\beta_x \rightarrow 1m/10m$

- Pareto optimal set.
- Twiss
- Twiss
- Movies

