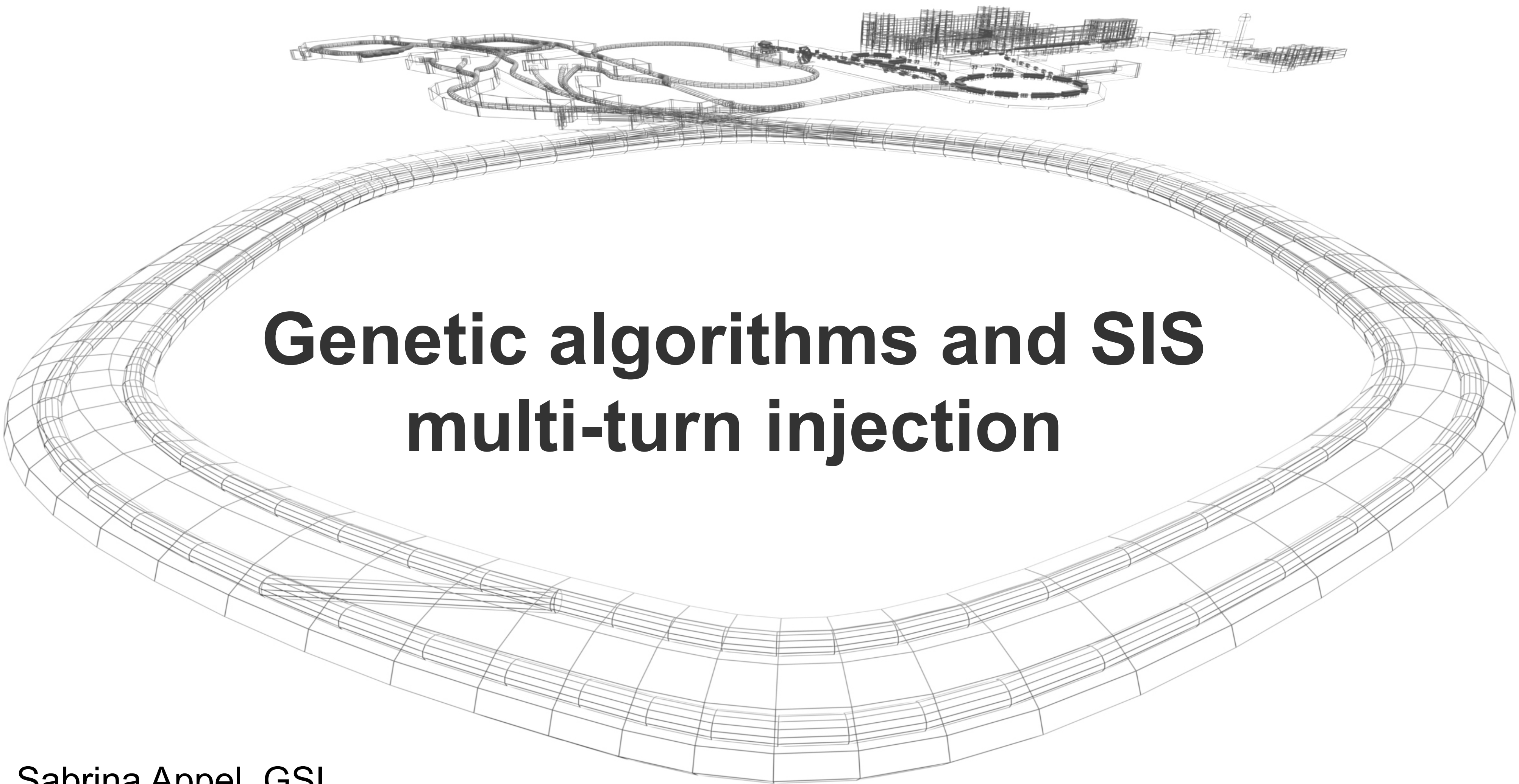
GSI Helmholtzzentrum für Schwerionenforschung GmbH



Genetic algorithms and SIS multi-turn injection

Sabrina Appel, GSI

Outline

- Numerical optimization
  - Single and multi-objective
- Genetic algorithms (GA)
  - Cycle and GA operators
  - GA implementation
- Multi-turn injection into SIS
- Improvement of MTI quality due to GA (first results)
- Summery and Outlook

Numerical optimization

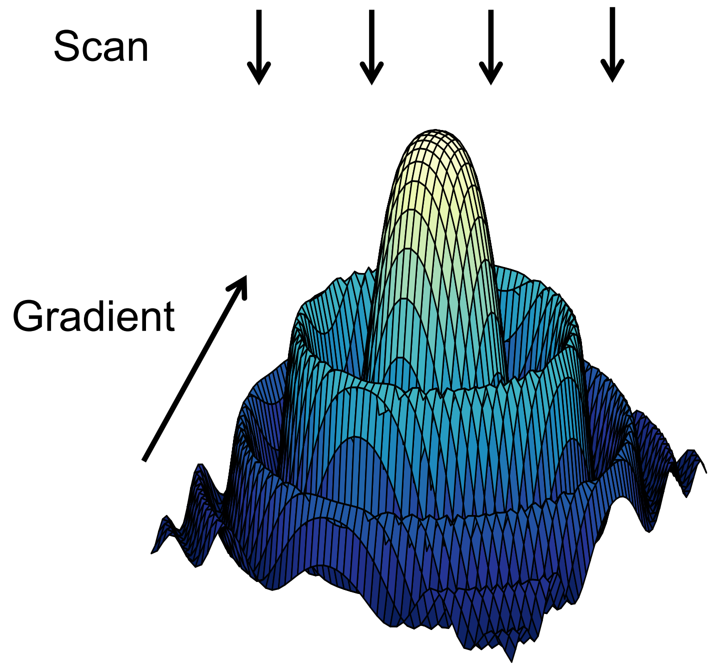
Optimization problem for single objective:

$$\max/\min(f(x, y, \ldots))$$

f(**x**) evaluated by simulation code (or measured in the machine) with only few iteration steps

Gradient-based methods (traditional):

- May get stuck in a local minimum/maximum (and never come out)
- Require local gradients
- Work if initial guess is already close to the optimum



Parameter scans (traditional):

- Only applicable for 1D or 2D parameter spaces

Accelerator problems:

- Multi-dimensional, nonlinear, multi-objective, contradicting optimums
- Several 'optimum' solutions (choice of the designer is required)

Numerical optimization

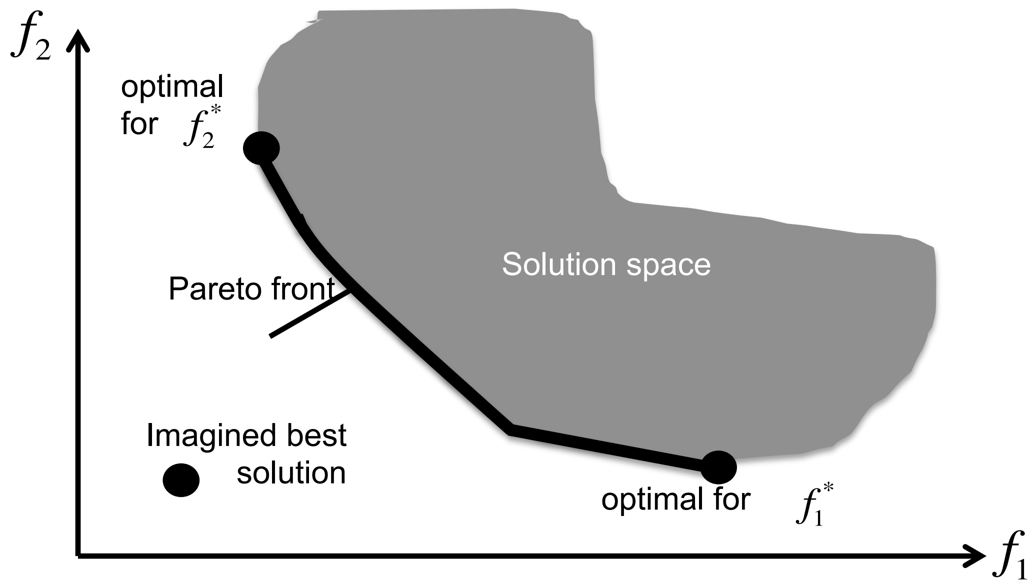
General:

- The optimization problem is multi-objective

$$\min(f_1(x), f_2(x), \ldots)$$

(several criterions to optimize)

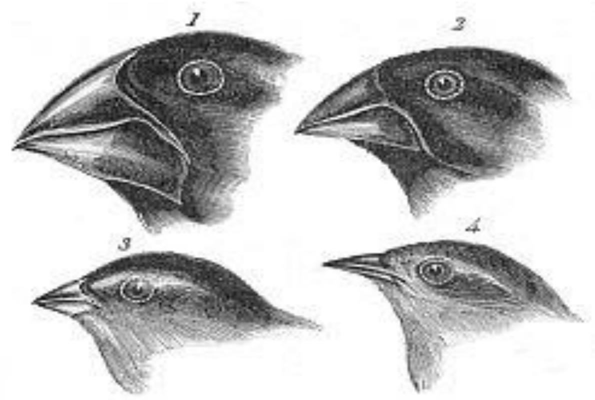
- The criterions can be contradicting
  - Improving one criterion means worsening others



- Find a set of optimal solutions instead of a single solution (Pareto front)

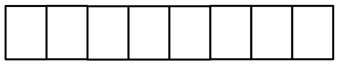
Inspired by natural evolution

- Search for solutions using techniques such as mutation, selection, and crossover
- Genetic algorithms are smart parameter scans
- They are very flexible and can solve multi-objective problems (wide range of different algorithms)
- Can be combined with gradient-based methods (for refinement)



Darwin Finches
J. Gould, Voyage of the Beagle

Individual in the population
One point in the search area



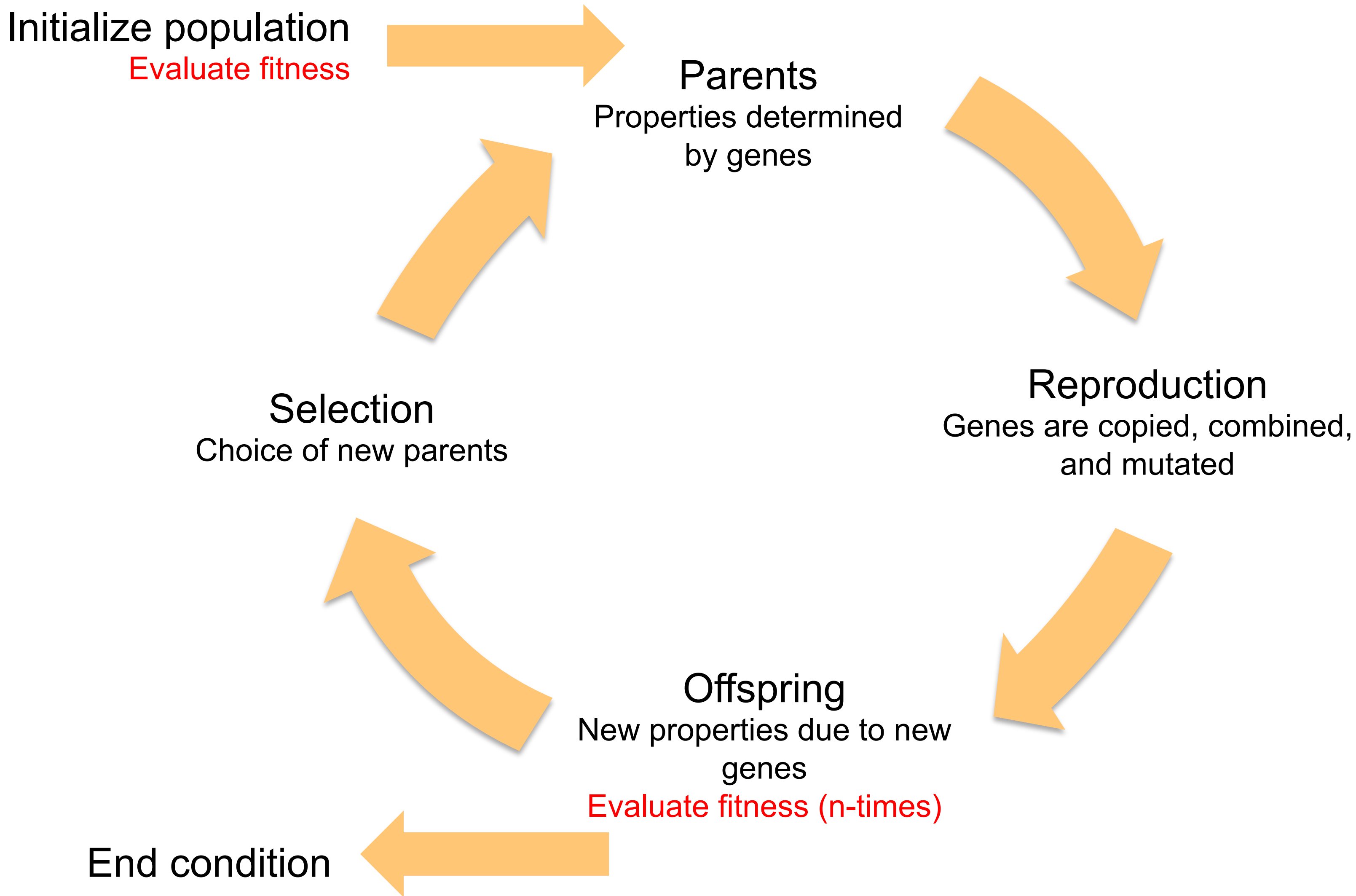
(n – parameters)

Fitness
Measure how good an individual is adapted to the optimization problem (fulfills constraints)

Selection
The survival of the fittest leads to an optimization of the properties

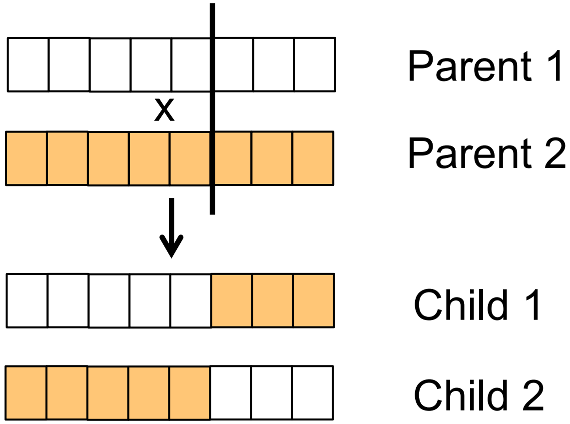
Variation
Recombination and mutation generated variety over the individuals

Genetic algorithms: Cycle

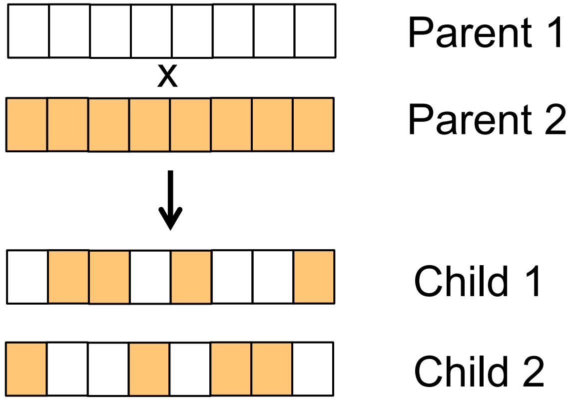


Genetic operators

Crossover
Discovering promising areas (Exploration)

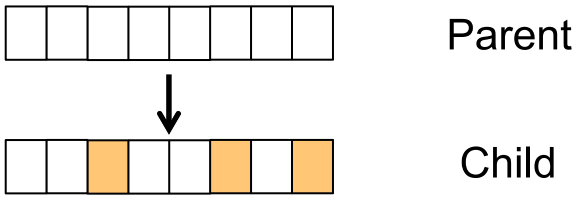


1 (n) – point crossover



uniform crossover

Mutation
Optimizing within a promising area (Exploitation)



Alter each gene with probability *p=1/l*
At least one bit on average should mutate

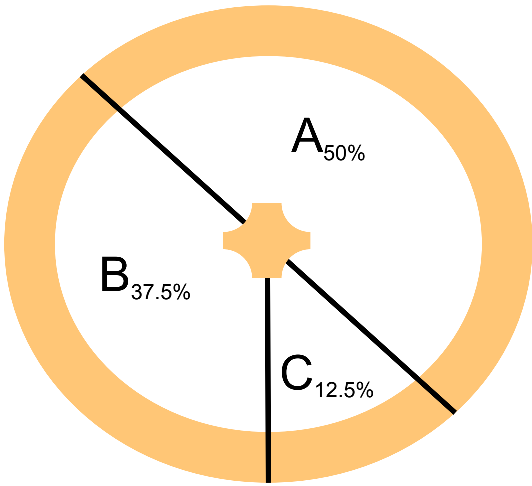
-> To find the optimum a combination of both is needed

Genetic operators

Selection (single-objective optimization)

- Choose the most promising individual to created the next generation
- Prevent the population to be dominated by a single individual (local optimum) by allowing individuals with poor fitness to take part at the creation process
- Techniques are fitness proportional selection, ranking selection, tournament selection, …

Roulette wheel technique



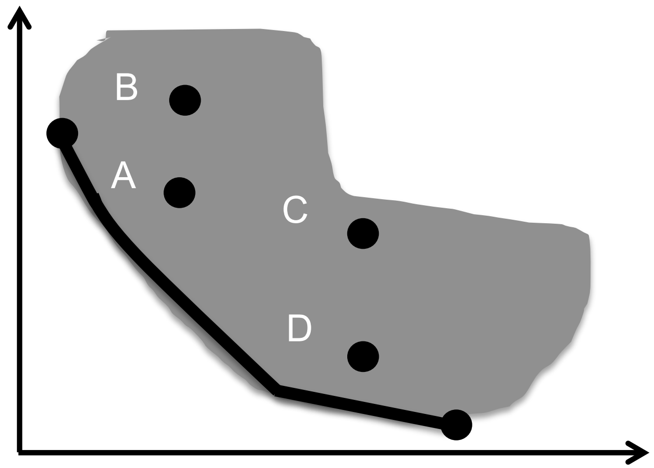
- Assign to each individual a part of the roulette wheel (The size is proportional to its fitness)
- Spin the wheel n times to select n individuals

Genetic operators

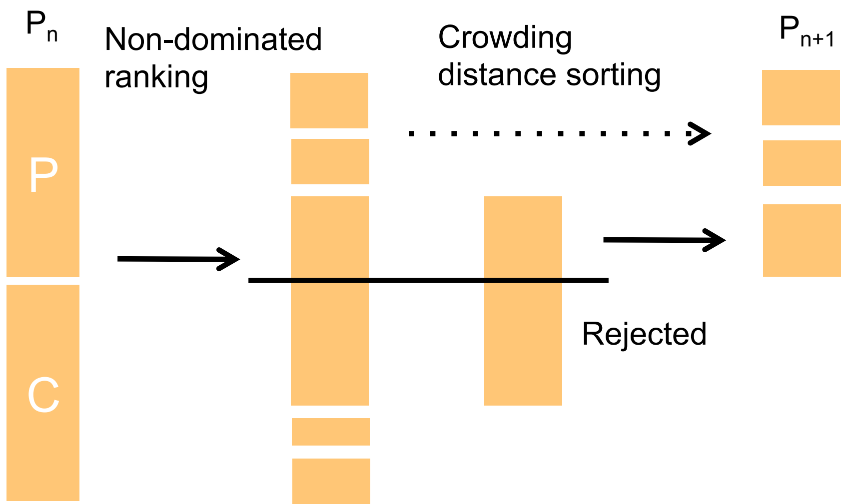
Selection (multi-objective optimization)

- Non-dominated selection (Selection of solution near to the Pareto front)

  - A dominates B+C but not D
  - D dominates C but not A+B
  - B + C do not dominate
  - A + D are non-dominated (near Pareto front)



NSGA-II (Non-dominated sorting genetic algorithm)



- The next generation is selected from parents and children
- The solutions are ranked according to their non-domination level and combined to sets
- The best non-dominated solutions are selected directly for the new generation
- Solutions which violate criteria or are of low rank are rejected

Genetic algorithm implementation

Parallel algorithms

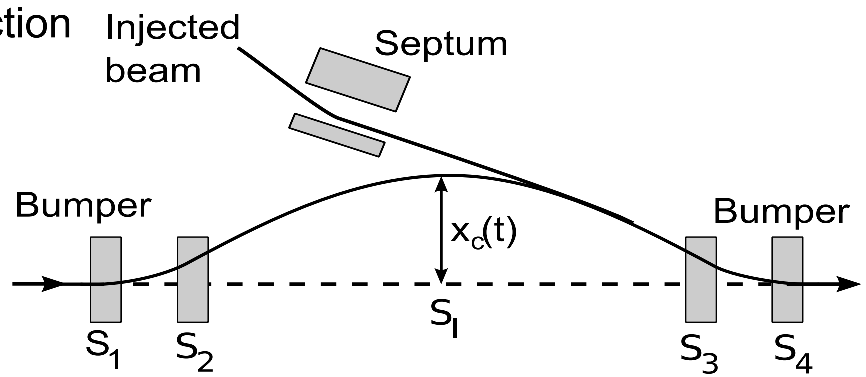
- Implementation of genetic operators and algorithms
- Use MPI to establish a master/slave model
  - The master performs genetic operations
    - Generate population, selection, crossover, mutation
  - The slaves evaluate fitness function for each individual
    - Accelerator simulation code
    - “Bottleneck” -> Will be called for each individual at each generation

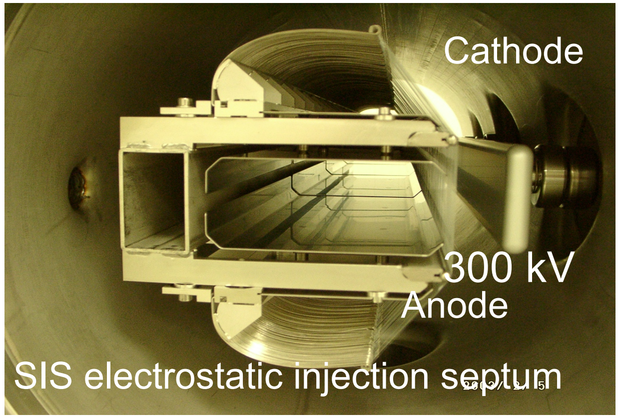
It sounds like a lot of work

- Not, if you use available genetic algorithm packages for Python, Java, Matlab, …
- And decouple the genetic algorithm from accelerator simulation code

Multi-turn injection into SIS

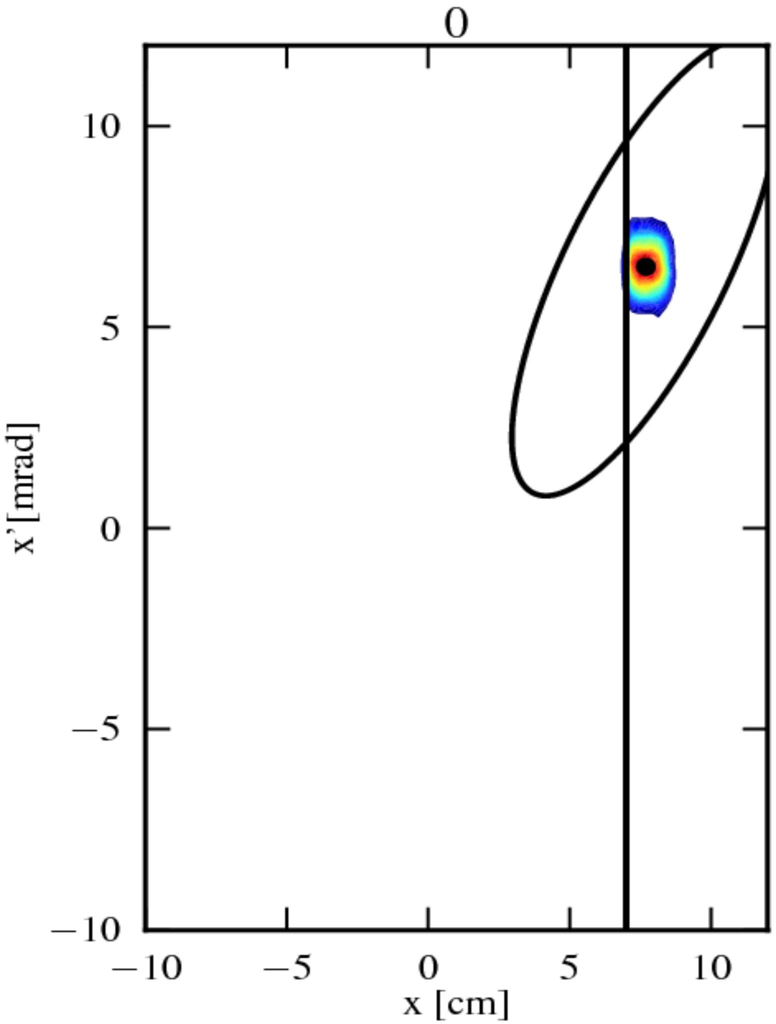
Layout of injection region





- The linac beam is injected in horizontal phase space until the machine acceptance is reached
- MTI has to respect Liouville's theorem: Injected beams only in free space
- Loss (at septum + acceptance) should be as low as possible due to activation, damage, vacuum
- Previous study indicated analytically description for the model variables, but the model is underrepresented
- For MTI simulations we use **pyORBIT** (A. Shishlo, S. Cousineau, J. Holmes et al.)
  - https://code.google.com/p/py-orbit/
  - Python + C^{++} + MPI
  - Teapot tracking
  - 2D/3D space charge models

Multi-turn injection into SIS: Movie



- Loss at septum is the major loss source
  - Loss of incoming beam
  - Later loss of stored beam

- Second loss source is the acceptance

The analytically description characterize:
- The position of incoming beam
- Input mismatch
- And the depending of one variable on other variables

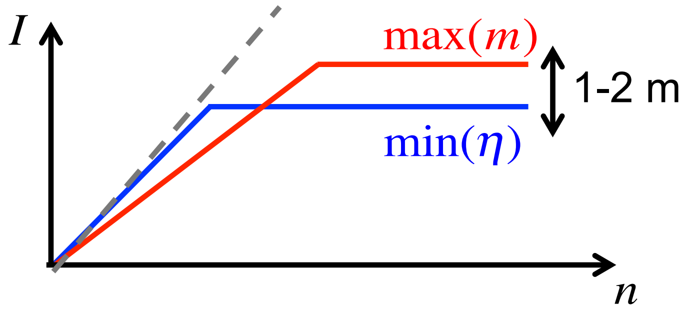
- But the model is underrepresented
  - A few variables can be choose freely from a value range

Multi-turn injection into SIS

- **Multi-objectives:**
  - stacked current (maximize) $I = mI_0$
  - beam loss (minimize) $\eta = \dfrac{I_{loss}}{nI_0}$

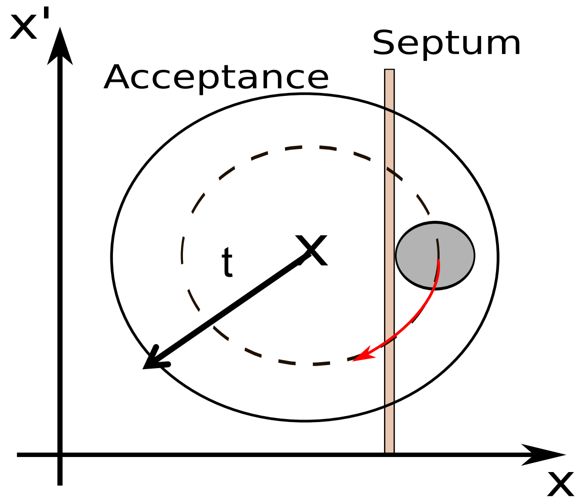
output

$m = (1-\eta)n$



- **Constraints:**
  - Position of septum x_s
  - Machine acceptance A
  - Closed orbit (bumper kick) $\phi_i(Q_x)$

Model in simulation code (fitness function)



- **Parameters:**
  - Position of incoming beam at septum x_c, x_c', M
  - Initial bump amplitude and its decreasing x_0, x_0', τ
  - Injected turns n
  - Horizontal tune Q_x
  - Horizontal emittance ε_x
  - Current from linac I_0

Improvement of the SIS MTI quality

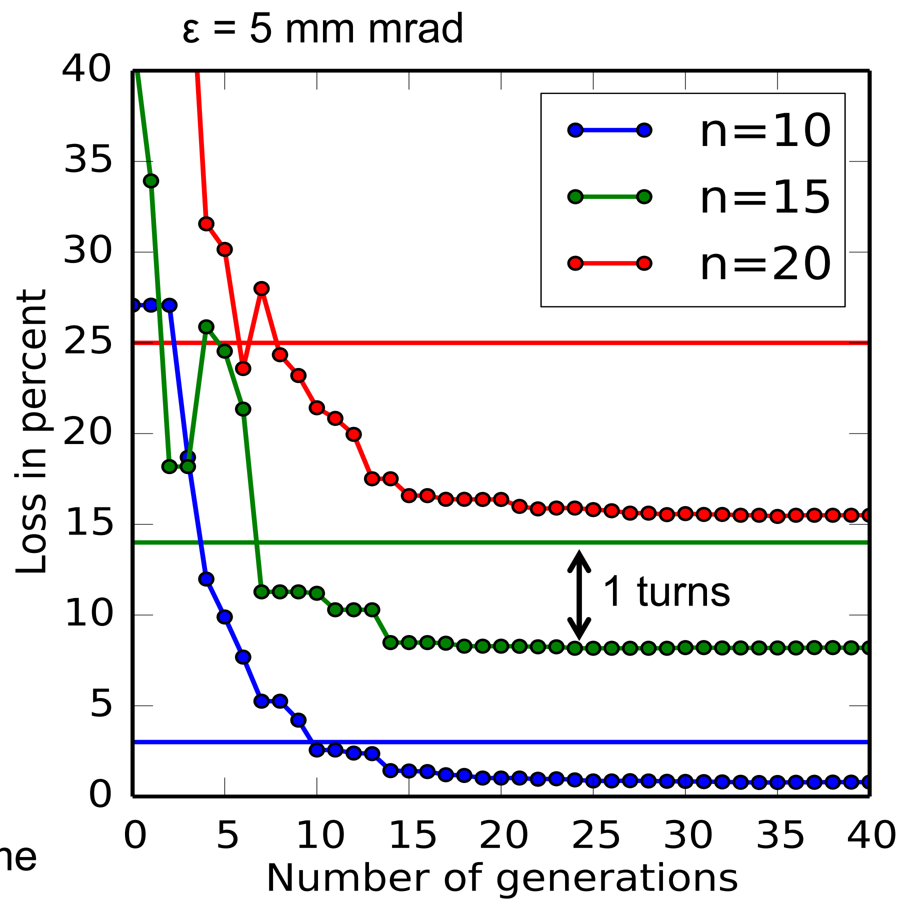
Single object: min (loss) $\eta = \frac{I_{loss}}{nI_0}$

Fixed parameters: $n \; Q_x \; \varepsilon_x \; I_0$

Variables: x_c, x_c', M (beam)

x_0, x_0', τ (orbit bump)

Genetic algorithms is better than the traditional approach: $\frac{\eta_{GA}}{\eta_{tra}} \approx \frac{2}{3}$



- The parameter set of the best individual are in the same area as the analytically description are suggested

- Due to an other combination of the freely selectable variables give the GA are more ideally solution than analytically description

Improvement of the SIS MTI quality

Multi-object: max(intensity) $I = mI_0$

and min (loss) $\eta = \dfrac{I_{loss}}{nI_0}$

$m = (1-\eta)n$

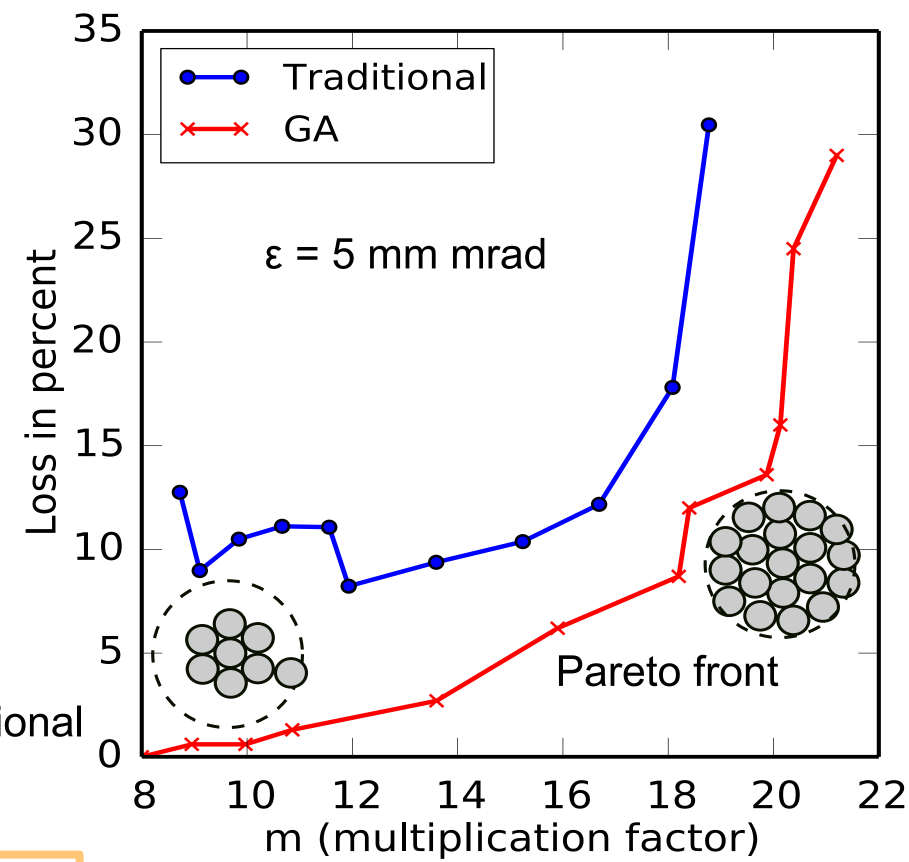
Fixed parameters: $Q_x \ \varepsilon_x \ I_0$

Variables: x_c, x_c', M (beam)

x_0, x_0', τ (orbit bump)

$+ \ n$ (inj. time)

- Genetic algorithms is better than the traditional approach



- A more better Pareto front have been found
  - Loss could minimized and intensity maximized

Other examples of applications of GA

- Dynamic aperture maximization

A. Hofler et al., Innovative applications of genetic algorithms to problems in accelerator physics Phys. Rev. ST AB, 16 (2013)

- Magnet design optimization

S. Ramberger, S. Russenschuck, *Genetic algorithms for the optimal design of superconducting accelerator magnets* EPAC (1998)

- Magnet sorting in a storage ring.

Chen, J., Wang, L., Li, W.-M., & Gao, W.-W. ,*Optimization of magnet sorting in a storage ring using genetic algorithms*, Chinese Physics C (2013)

- Linac settings for high intensity

Pang, X., & Rybarcyk, L. J., *Multi-objective particle swarm and genetic algorithm for the optimization of the LANSCE linac operation.* NIMA 741 (2013)

- Real machine based optimization in a storage ring

L.Yang, et al. , Global optimization of an accelerator lattice using multiobjective genetic algorithms, NIMA, 609 (2009)

Summary and Outlook

- Summary
  - Numerical optimization
  - Genetic algorithms (GA)
  - Improvement of MTI quality due to GA (first results)
  - Various applications for accelerators

- Outlook
  - Include in GA optimization more parameters like tune, current
  - Use of other algorithms like particle swarm optimization
  - Improvement of MTI simulation model due to measurements (analysis is in progress)

Thank you for your attention