OPTIMIZING SORTING WITH GENETIC ALGORITHMS

AUTHORS

Xiaoming Li, María Jesús Garzarán, and David Padua

University of Illinois

http://polaris.cc.uiuc.edu

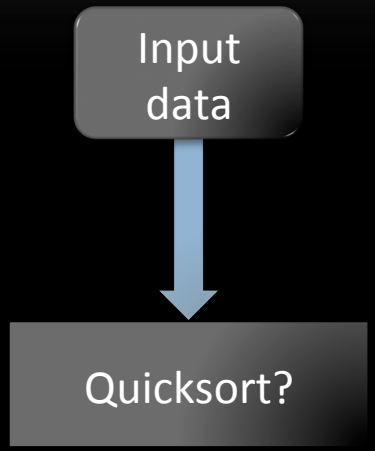
Marcos Sánchez 19-10-2011

INDEX

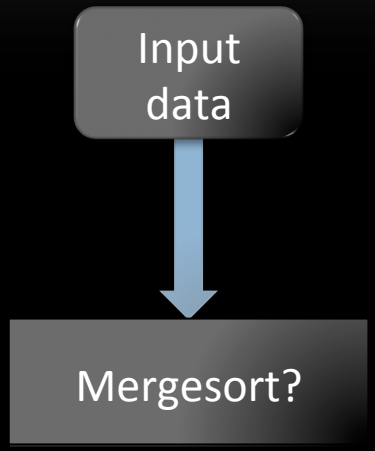
- INTRODUCTION
- SORTING PRIMITIVES
- GENE SORT
- EVALUATION OF GENE SORT
- CLASSIFIER SORTING
- CONCLUSION

INDEX

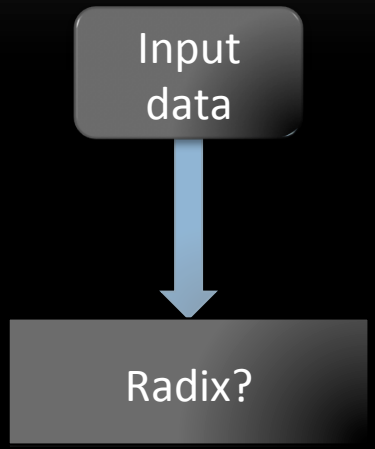
INTRODUCTION



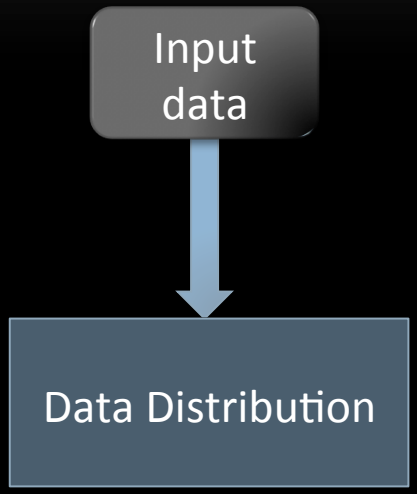
INTRODUCTION



INTRODUCTION



INTRODUCTION



INDEX

SORTING PRIMITIVES

Sorting

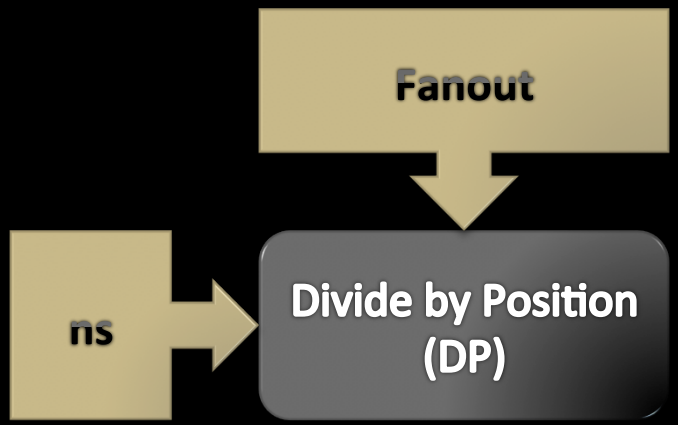
Selection

SORTING PRIMITIVES

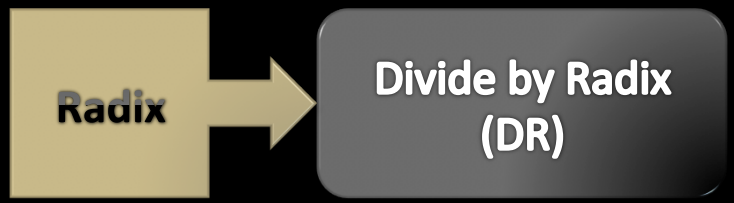
Sorting

SORTING PRIMITIVES

SORTING PRIMITIVES



SORTING PRIMITIVES



SORTING PRIMITIVES

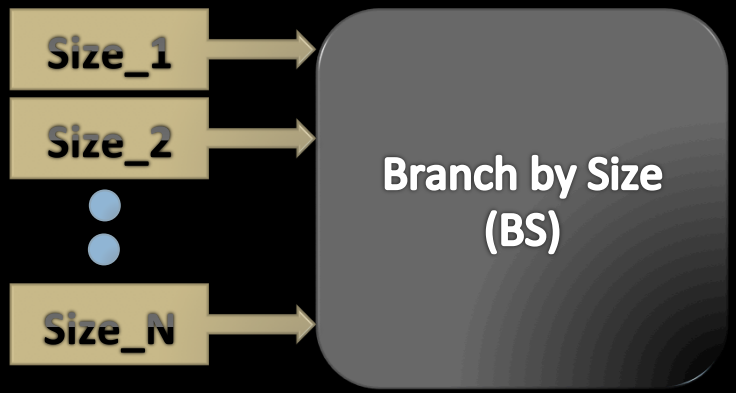
SORTING PRIMITIVES

SORTING PRIMITIVES

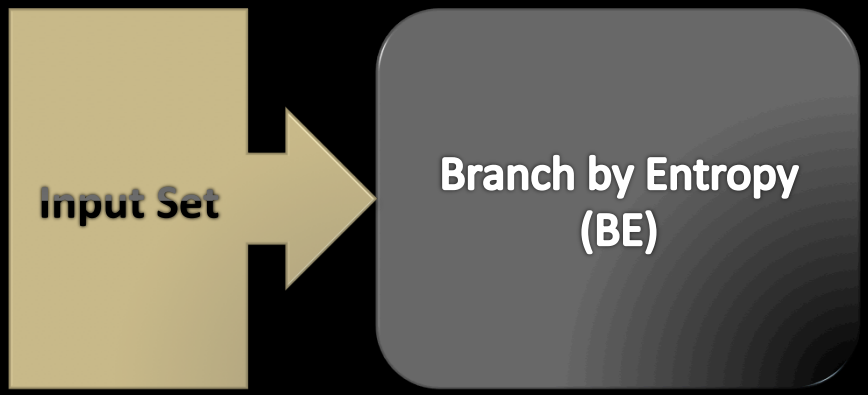
SORTING PRIMITIVES

Selection

SORTING PRIMITIVES

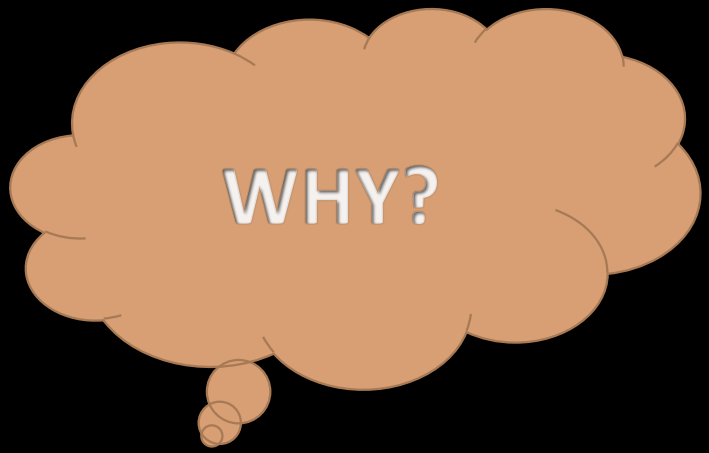


SORTING PRIMITIVES



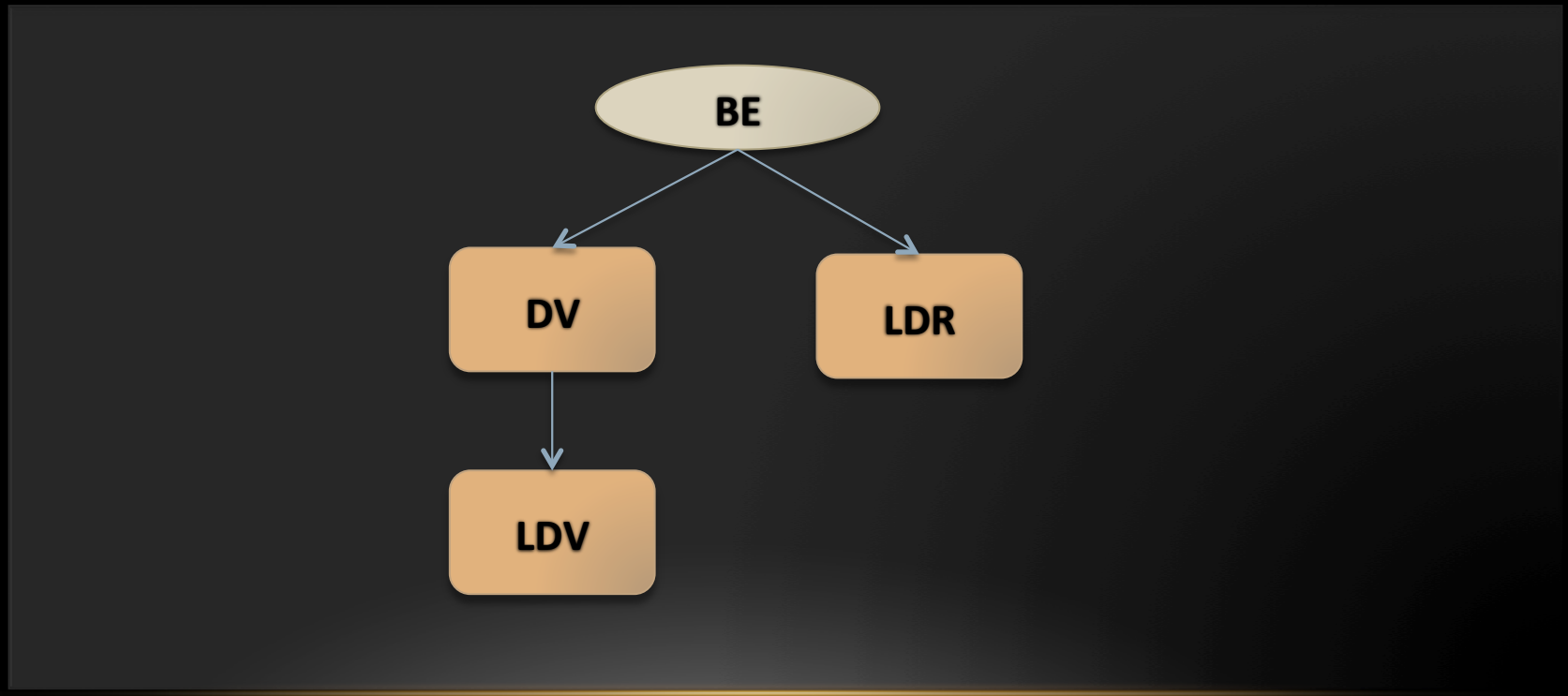
INDEX

GENE SORT



GENE SORT

Example of a Sorting Algorithm Tree



GENE SORT

Crossover

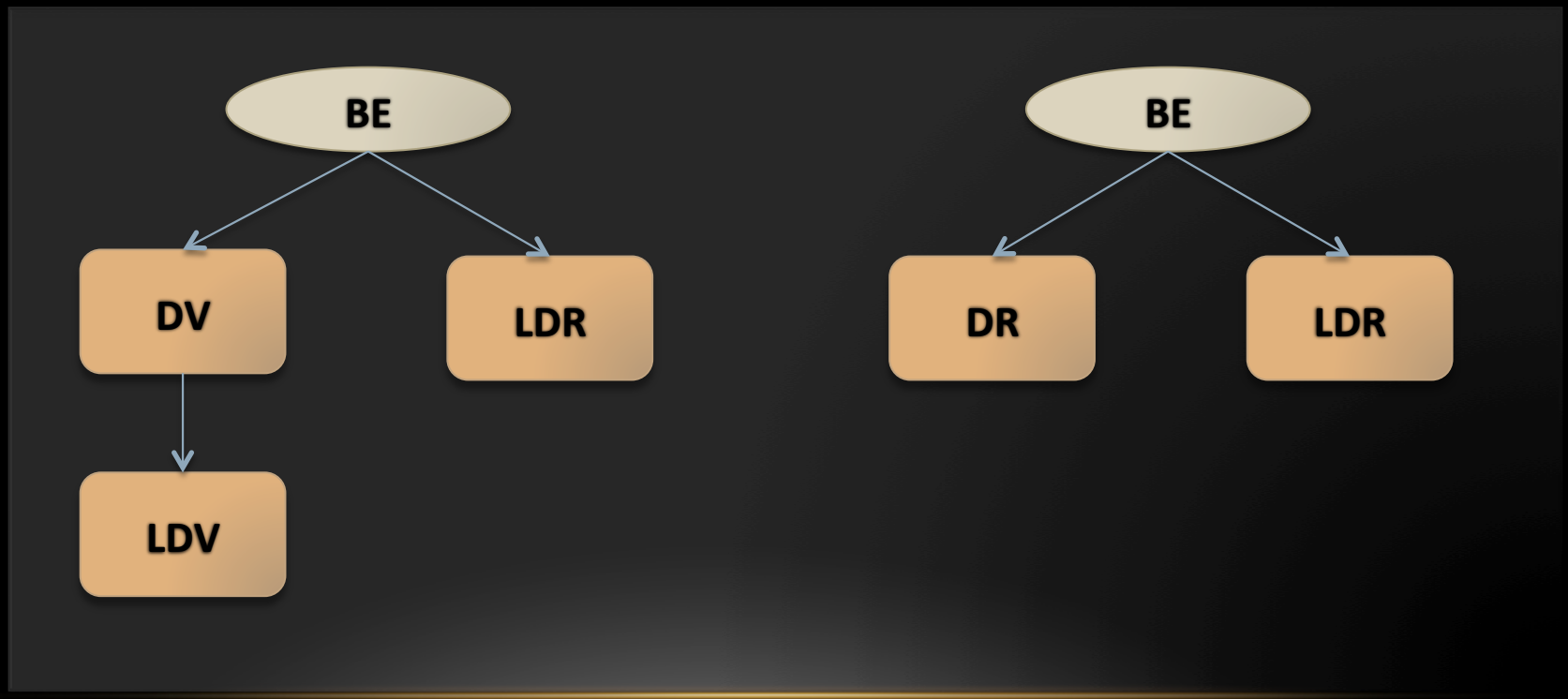
Mutation

GENE SORT

Crossover

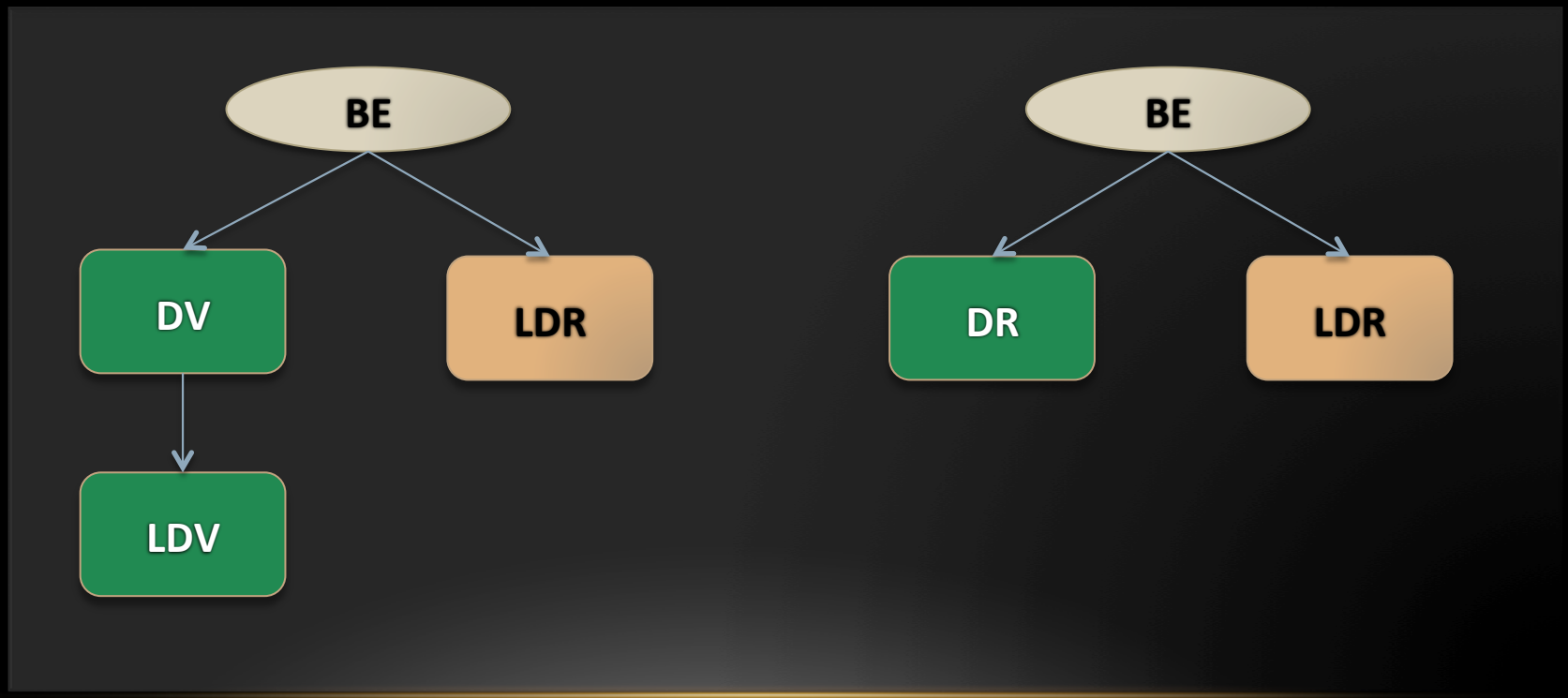
GENE SORT

Example of Crossover



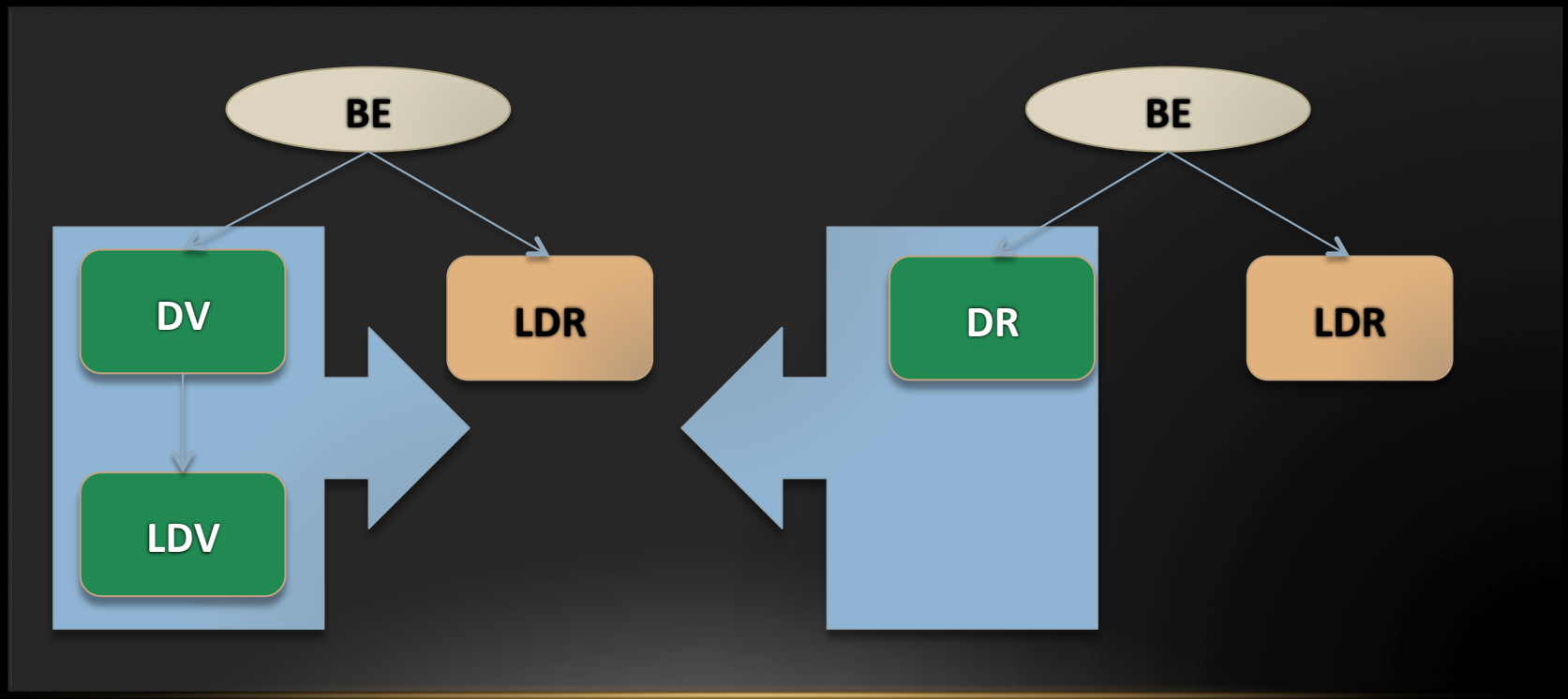
GENE SORT

Example of Crossover



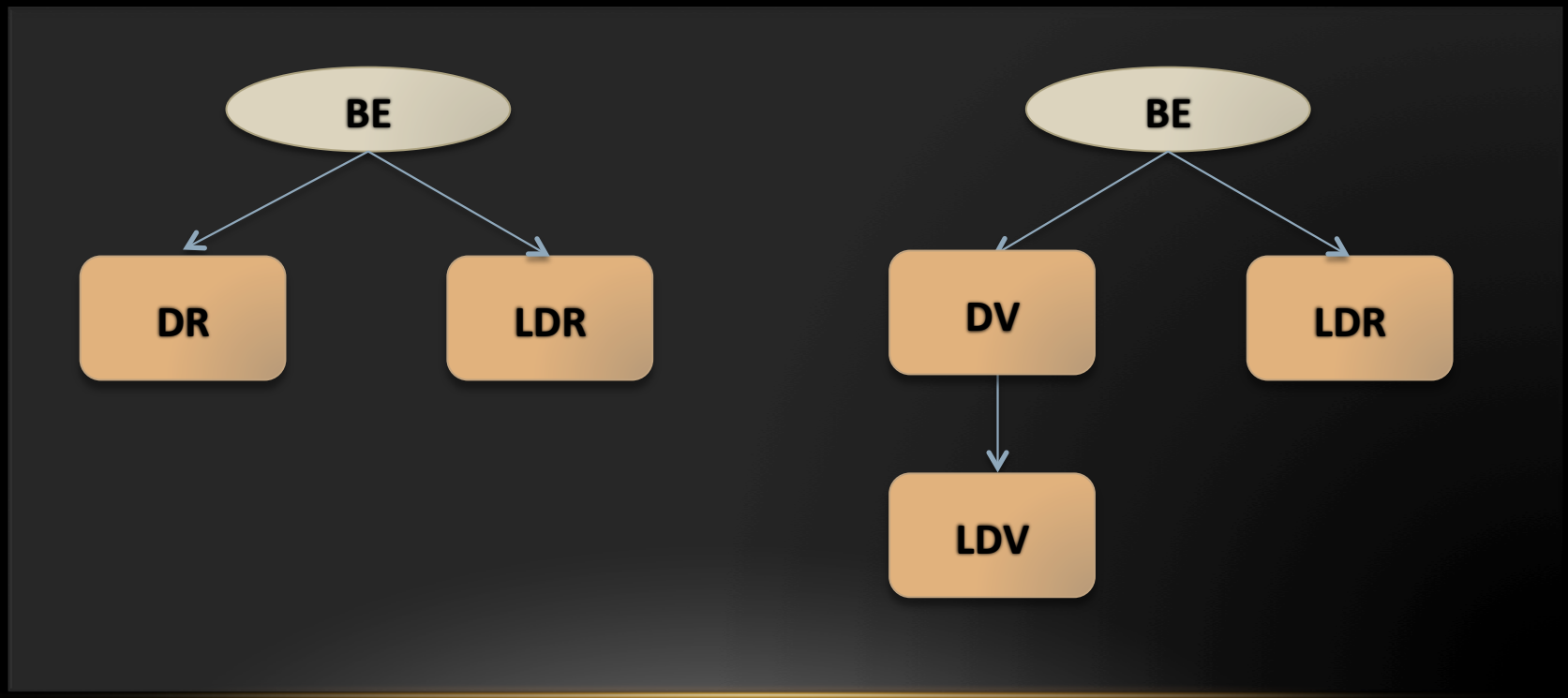
GENE SORT

Example of Crossover



GENE SORT

Example of Crossover



GENE SORT

Mutation

GENE SORT

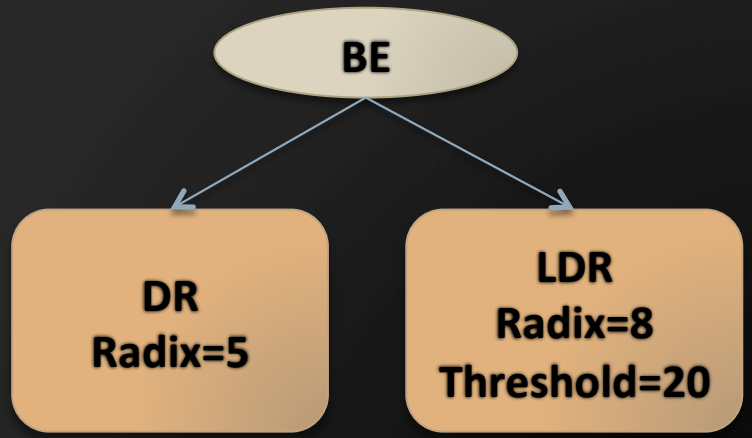
Mutation

- Change values of parameters (randomly)
- Exchange two subtrees
- Add a new subtree
- Remove a subtree

GENE SORT

Mutation

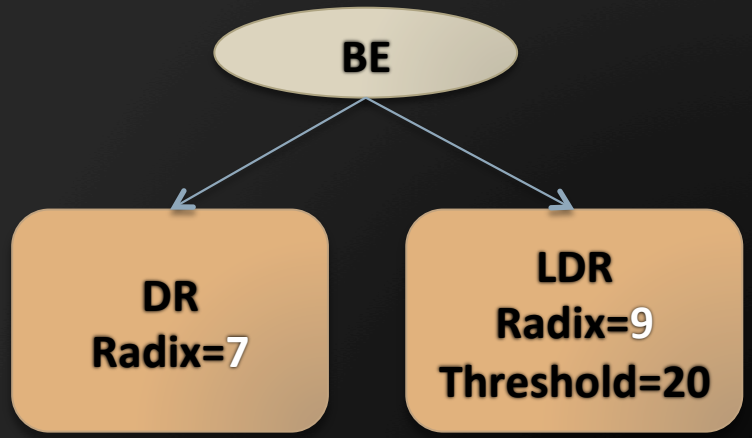
Change values of parameters (randomly)



GENE SORT

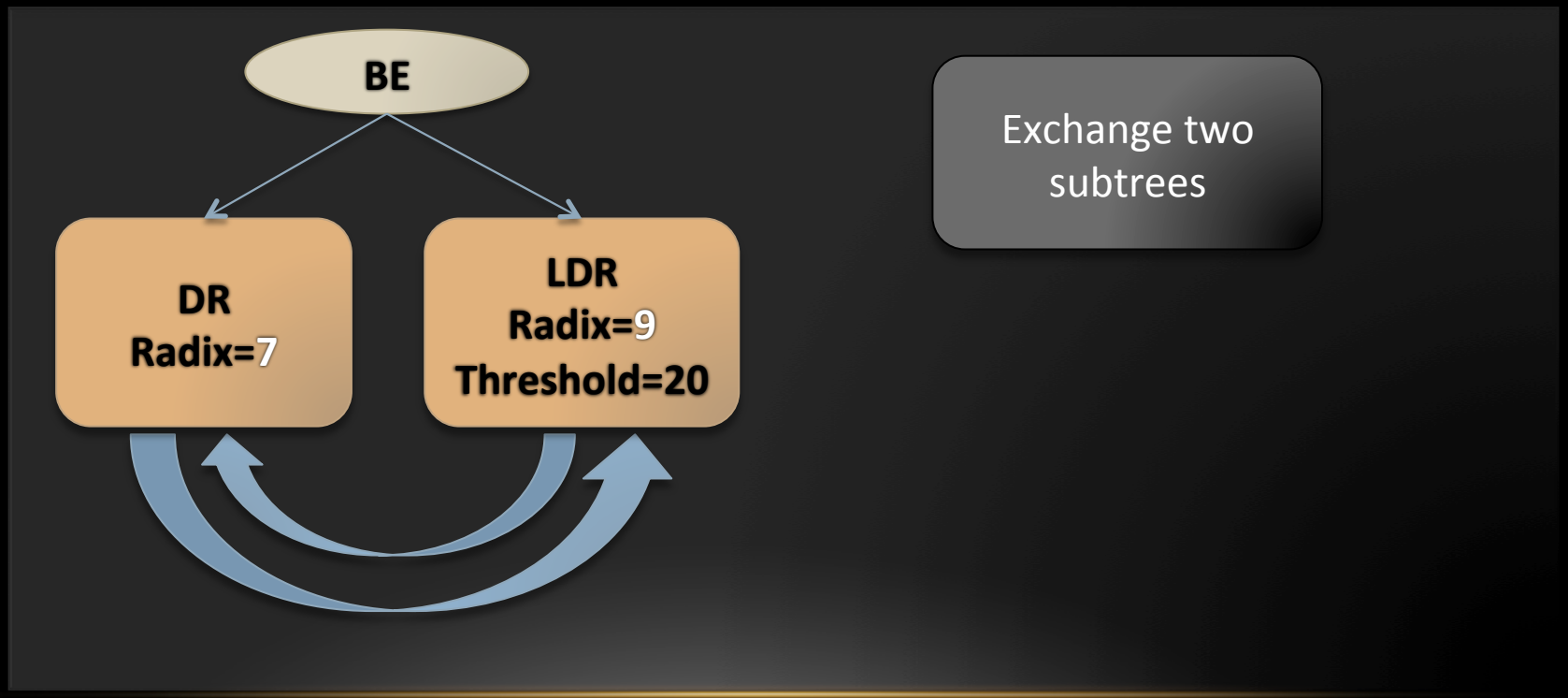
Mutation

Change values of parameters (randomly)



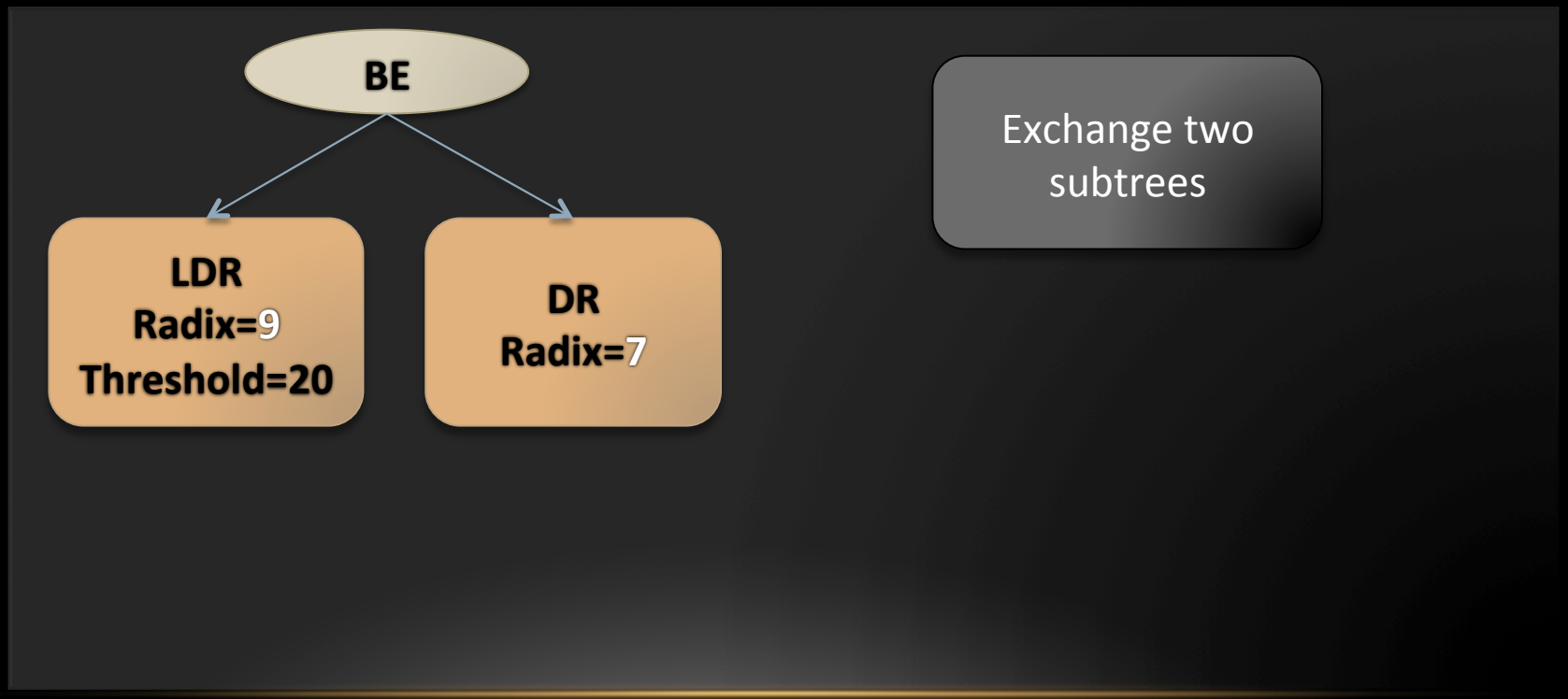
GENE SORT

Mutation



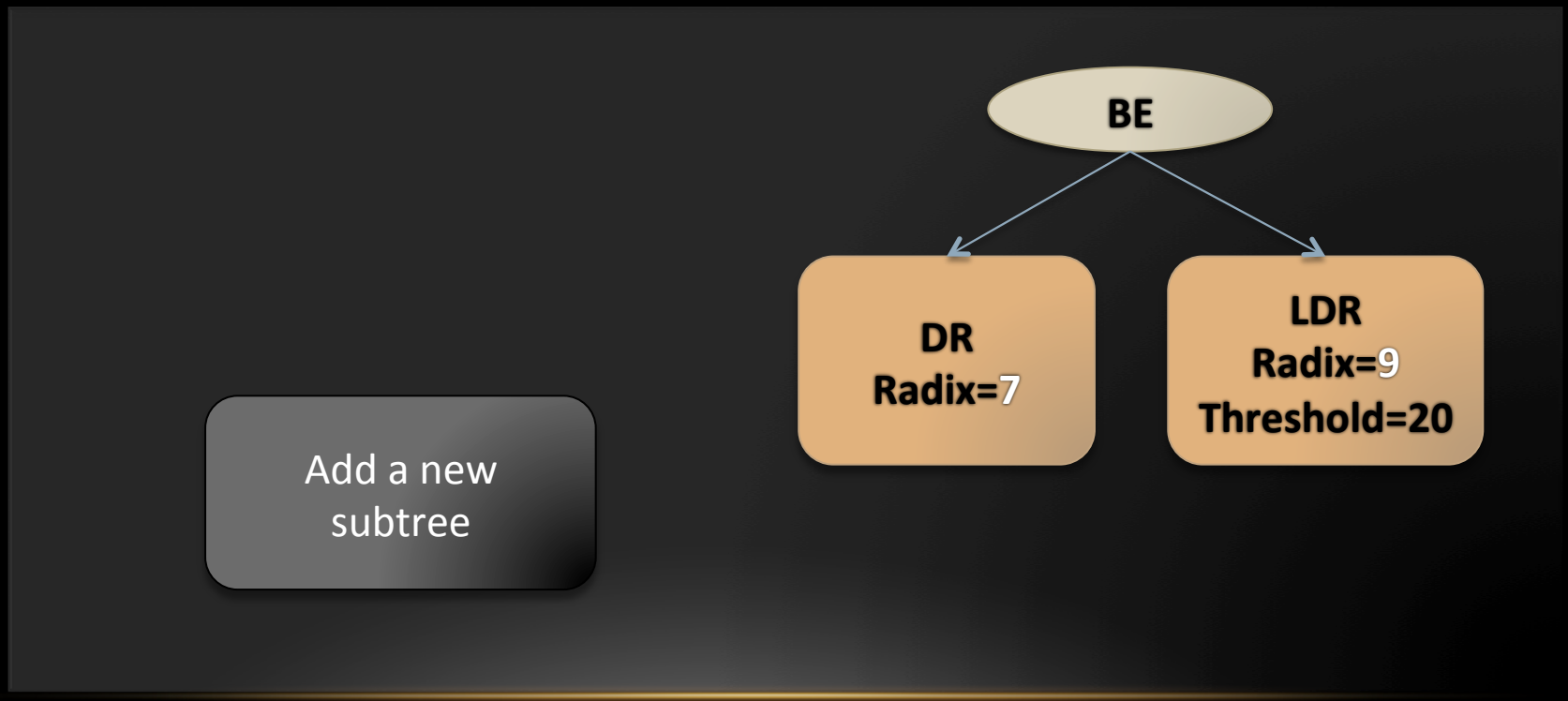
GENE SORT

Mutation



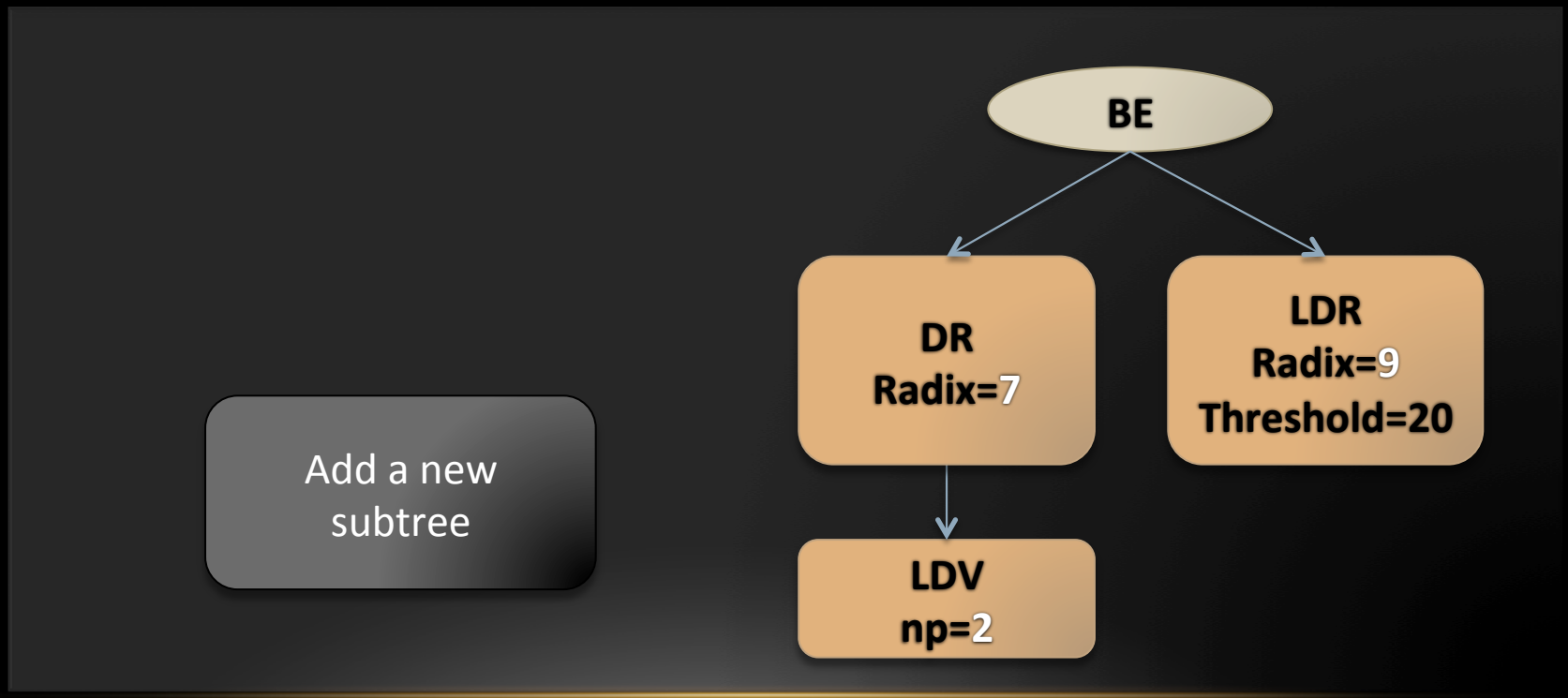
GENE SORT

Mutation



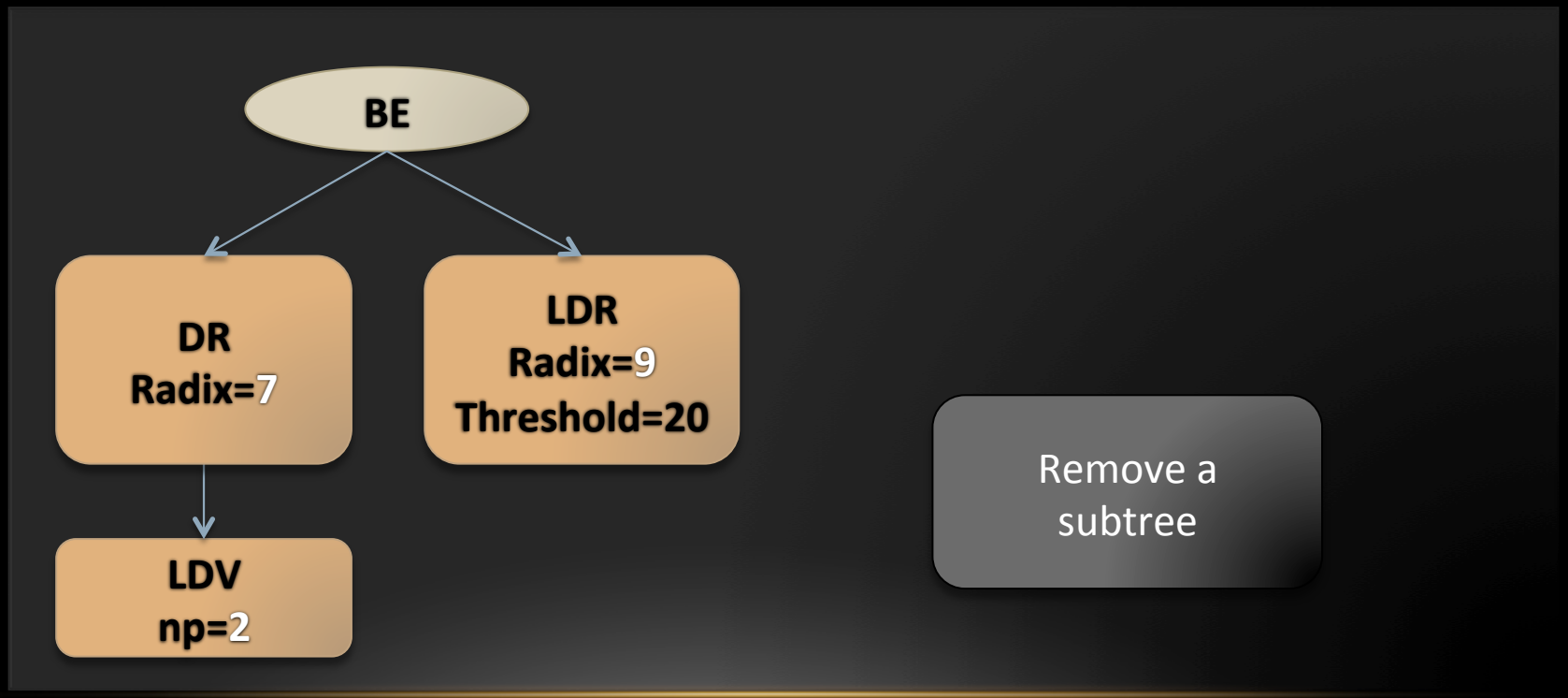
GENE SORT

Mutation



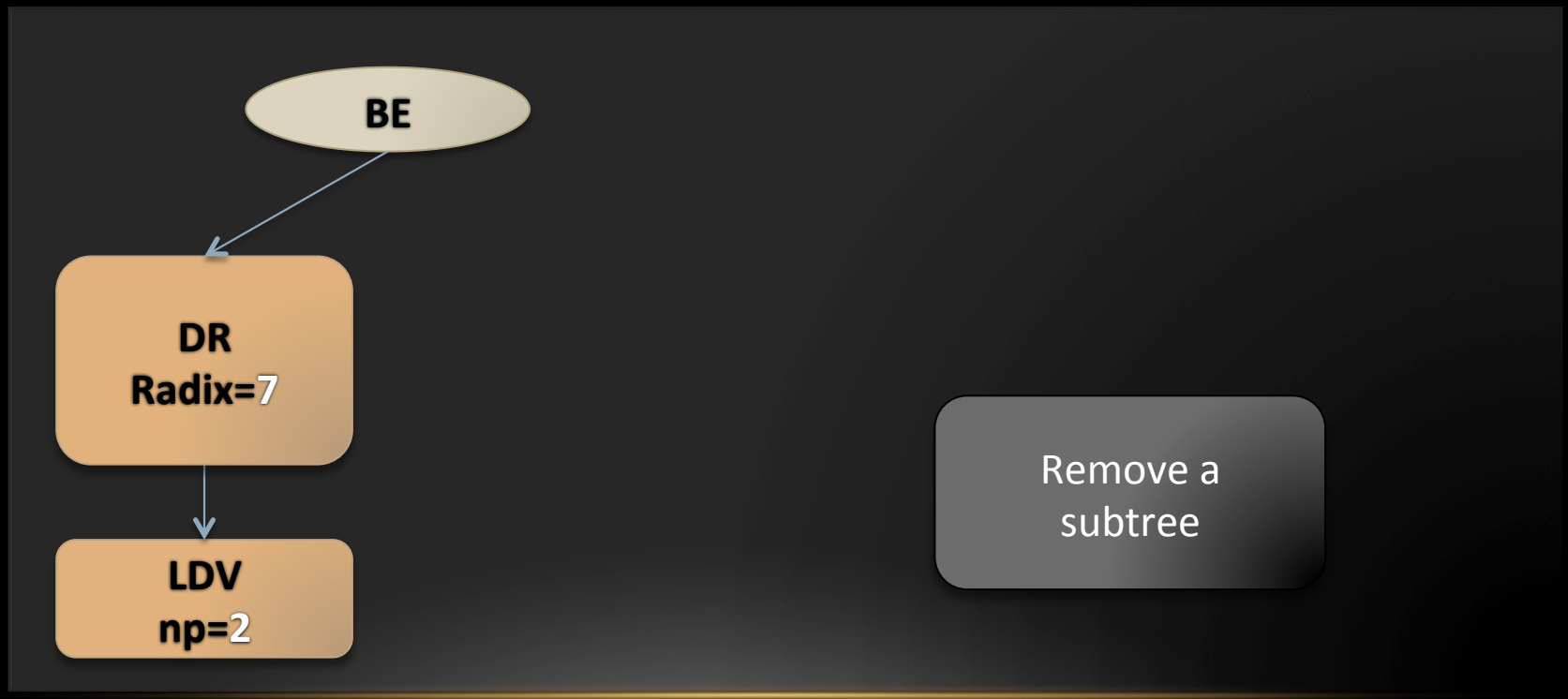
GENE SORT

Mutation



GENE SORT

Mutation

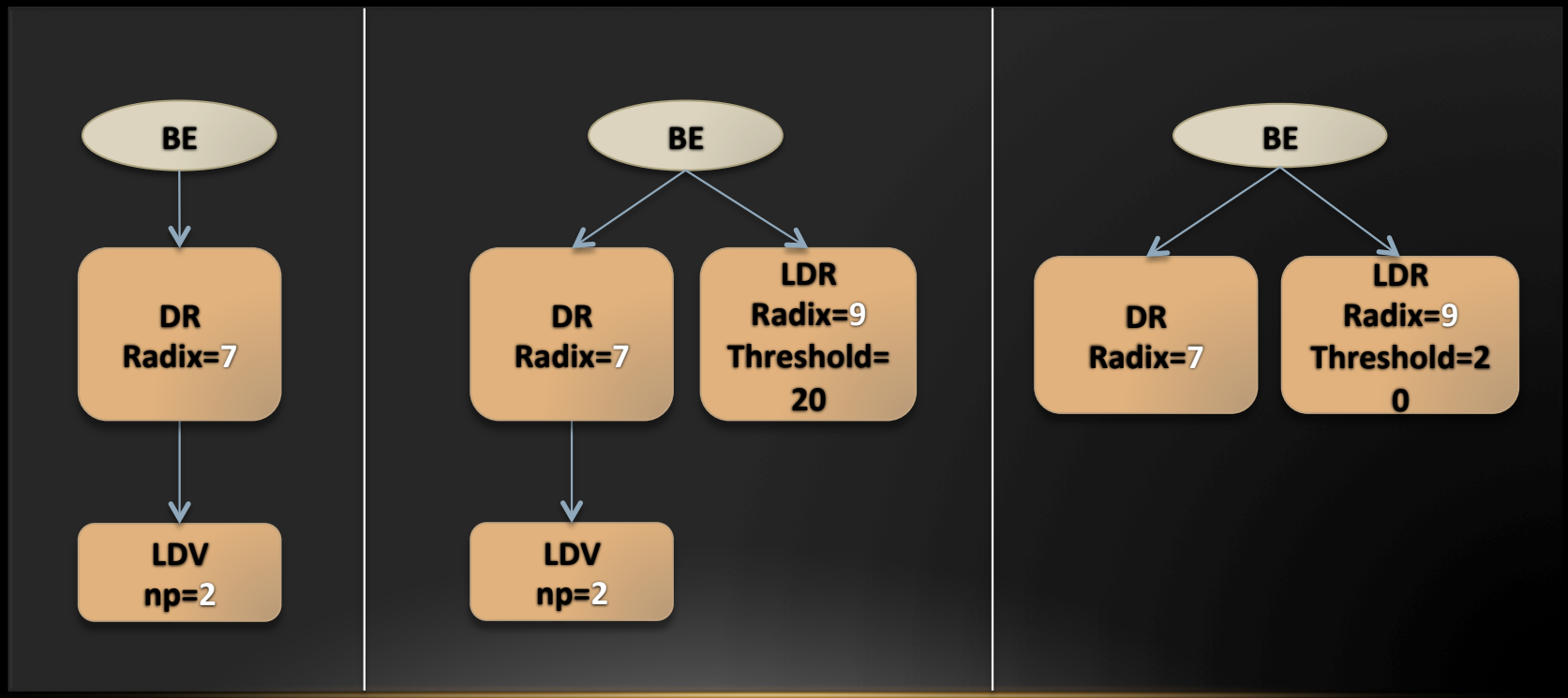


GENE SORT

Fitness Function

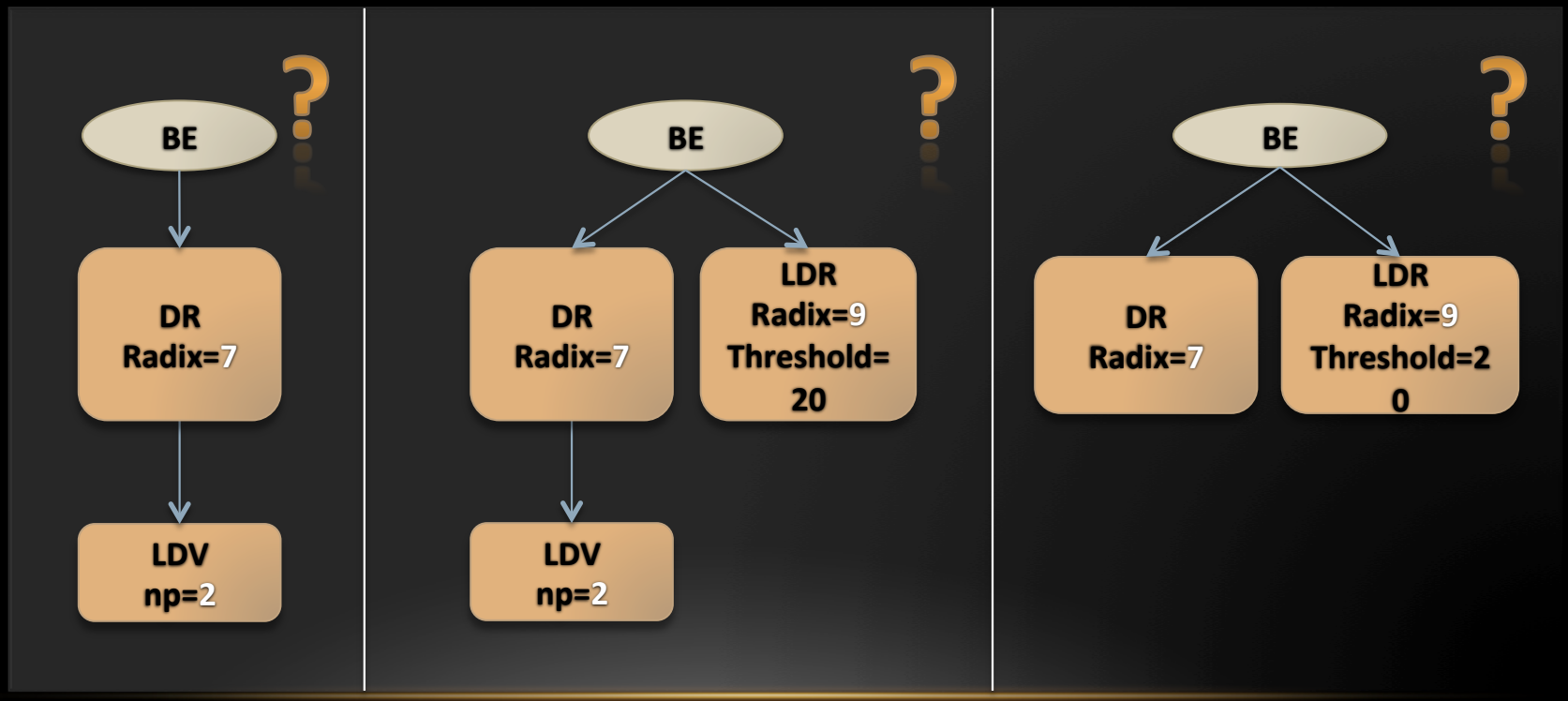
GENE SORT

Fitness Function



GENE SORT

Fitness Function



GENE SORT

Performance as Fitness Function

GENE SORT

Genetic Algorithm

GENE SORT

Genetic Algorithm {

P = Initial Population

While (stopping criteria is false) do {

Apply mutation and crossover and generate set M of k individuals
P = P ∪ M
S = Input sets with different sizes and different standard deviations
Use each genome of P to sort each element of S
Apply fitness function to remove the k least fit individuals from P.

}}

INDEX

EVALUATION OF GENE SORT

- Tested on seven different platforms

Settings	
Population Size	50
# Generations	100
# Generated offsprings	30
Mutation Rate Possibilty	6%
# Training input sets	12

EVALUATION OF GENE SORT

9h on the Intel Xeon
(3.0 GHz)

80h on the SGI R12000
(300 MHz)

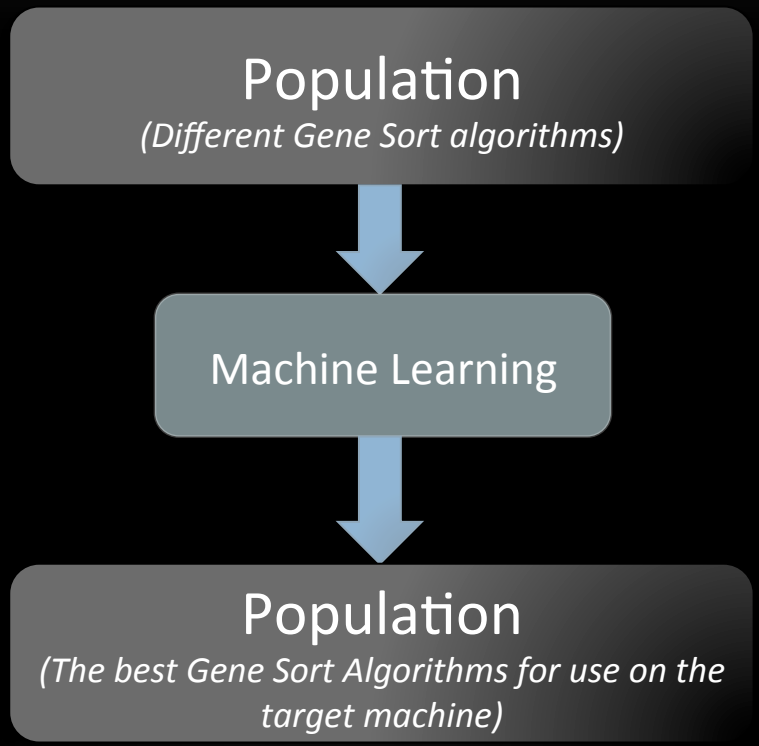
EVALUATION OF GENE SORT

Gene Sort : 7% worse than other algorithm ONLY in three platforms.

- AMD Athlon (having low values of std. Deviation)
- SGI R12000 and Intel Itanium (having high values of std. Deviaton)

INDEX

CLASSIFIER SORTING



INDEX

CONCLUSION

- Best sorting routines using Genetic Algorithms to generate Classifier Sorting (Xsort).
- 36% Faster than “Pure” sorting algorithms
- Faster than comercial libreries like IBM ESSL, INTEL MKL, STL (C++),etc.

THAT'S ALL

THANK YOU VERY MUCH FOR YOUR ATTENTION