Random-Key Optimization:
Problem Independent Solvers for Combinatorial Optimization

Mauricio G. C. Resende

Amazon (Bellevue, Washington)

Joint work with M. Schuetz, K. Brubaker, and H. Katzgraber

(all three with AWS)

INFORMS Annual Meeting – Indianapolis, October 16, 2022

Random Key

- A random real number $x_i \in [0,1)$

Vector of Random Keys

- A vector of n random keys: $X = (x_1, x_2, \ldots, x_n)$
- A vector of random keys can encode a solution of a combinatorial optimization problem
  - Permutation: Sort $X = (.92, .15, .28, .05) \longrightarrow \pi = (4, 2, 3, 1)$
  - Indicator: Ind(.25) $X = (.92, .15, .28, .05) \longrightarrow Y = (1, 0, 1, 0)$
  - Hybrid: $X = (.92, .75, .98, .19 \mid .05, .45, .22, .77) \longrightarrow (4,2,1,3 \mid 1, 5, 3, 8)$

Sort keys and select three smallest

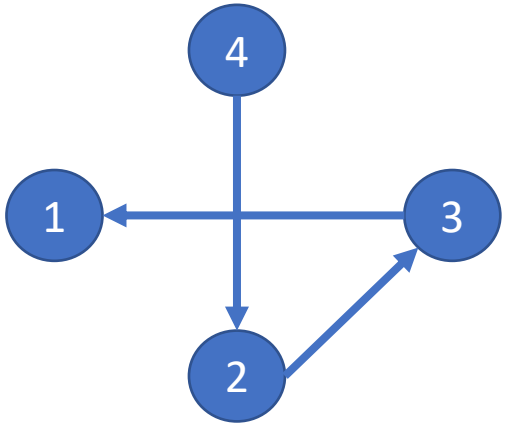
Assign weights $w_i = \lfloor 10 \times x_i \rfloor + 1$

(1,5,3)

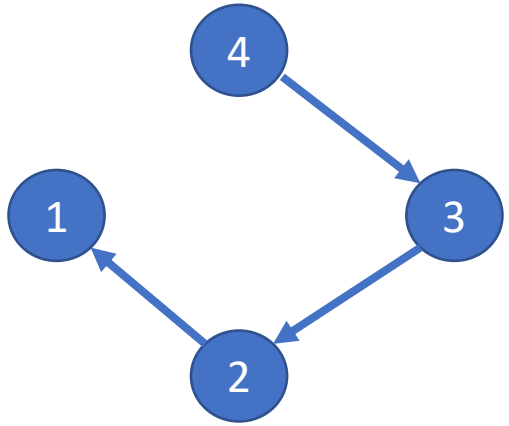
Decoders

A decoder is a deterministic algorithm that takes as input a vector of N random keys and outputs a solution to the combinatorial optimization problem

$X = (.92, .15, .28, .05)$
$\pi = \text{sort}(X) = (4, 2, 3, 1)$



$X = (.82, .75, .29, .09)$
$\pi = \text{sort}(X) = (4, 3, 2, 1)$



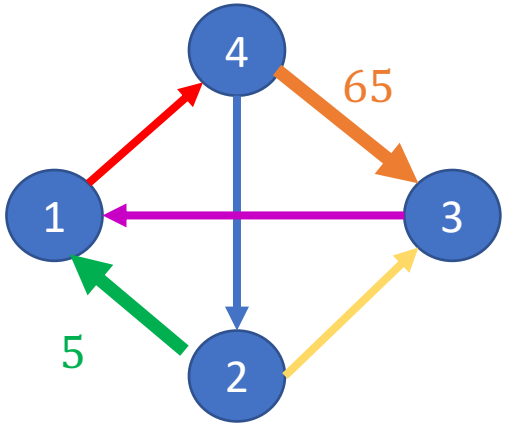
TSP (Bean, 1994)

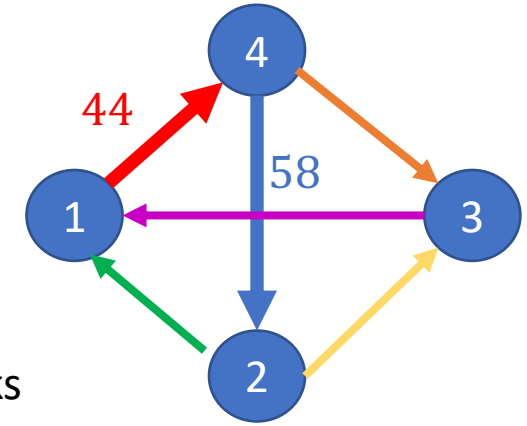
Decoders

A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$

$X = (.12, .35, .88, .15, .13, .99 \mid .44, .99, .43, .14, .58, .02)$





Toll setting in road networks (Stefanello et al., 2017)

Decoders

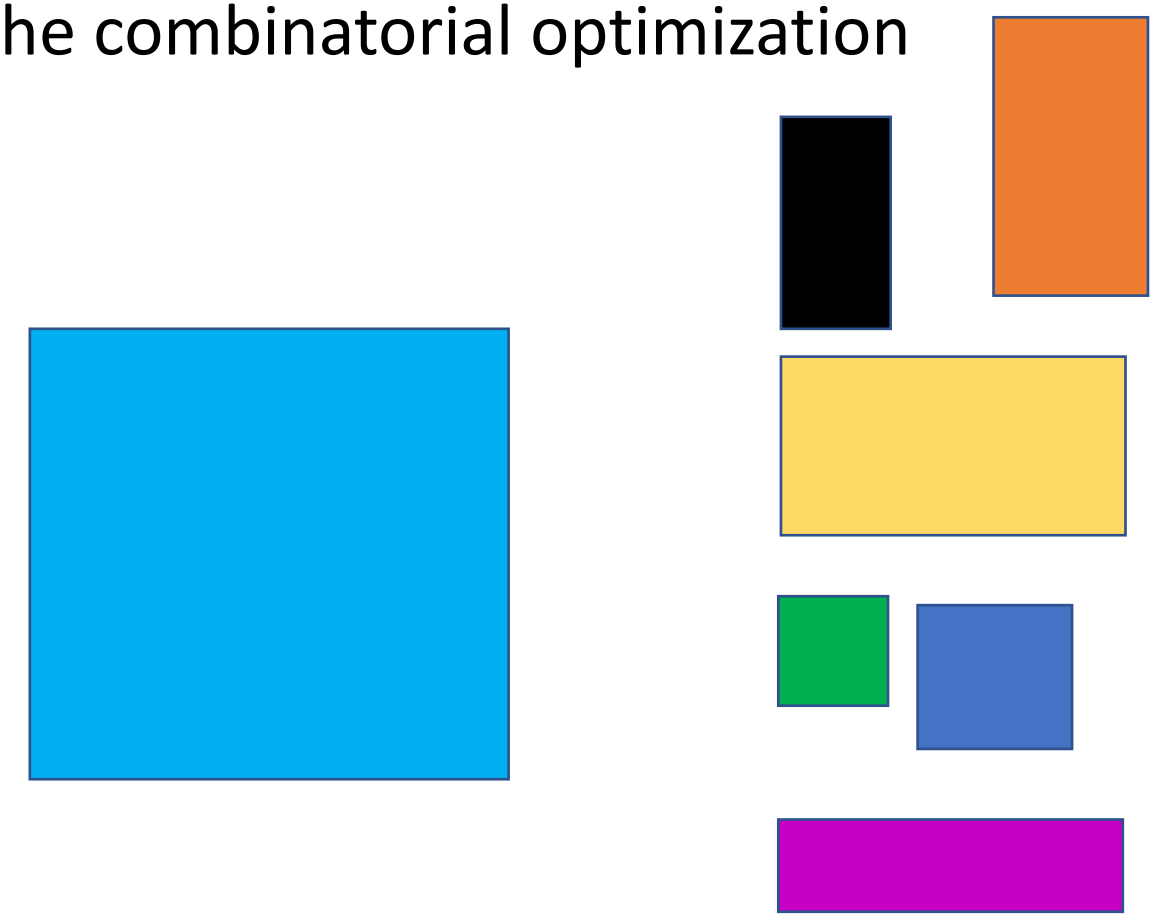
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

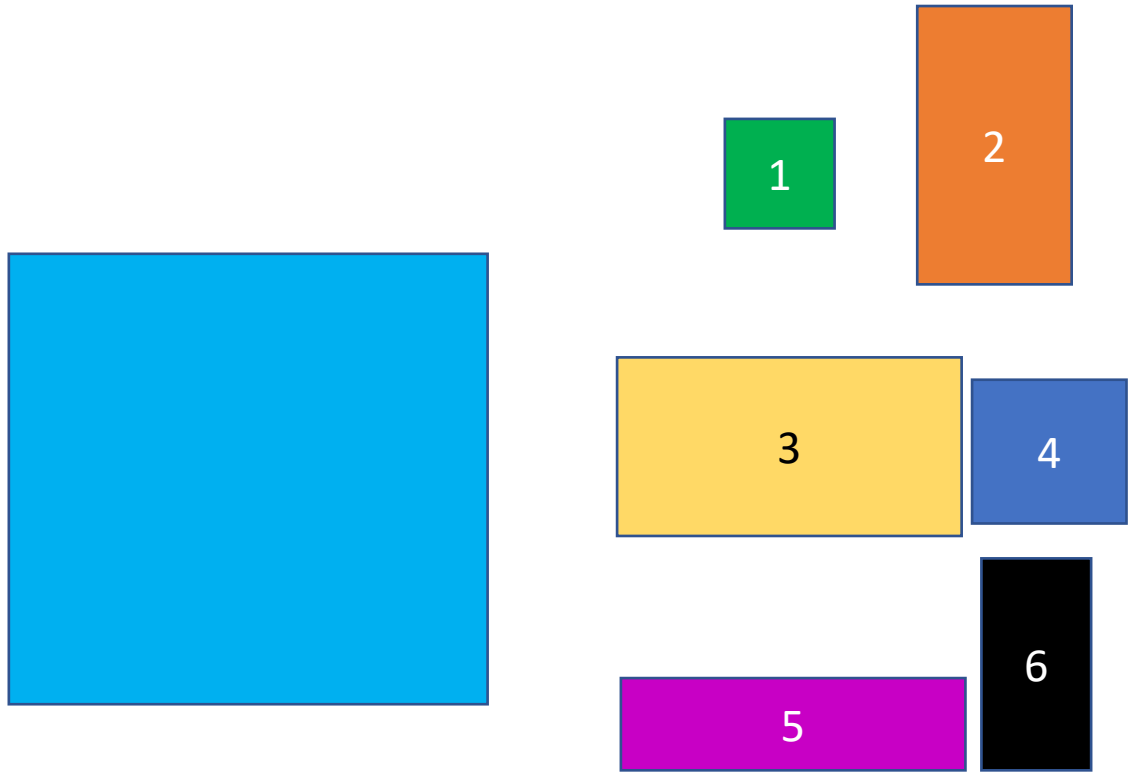
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

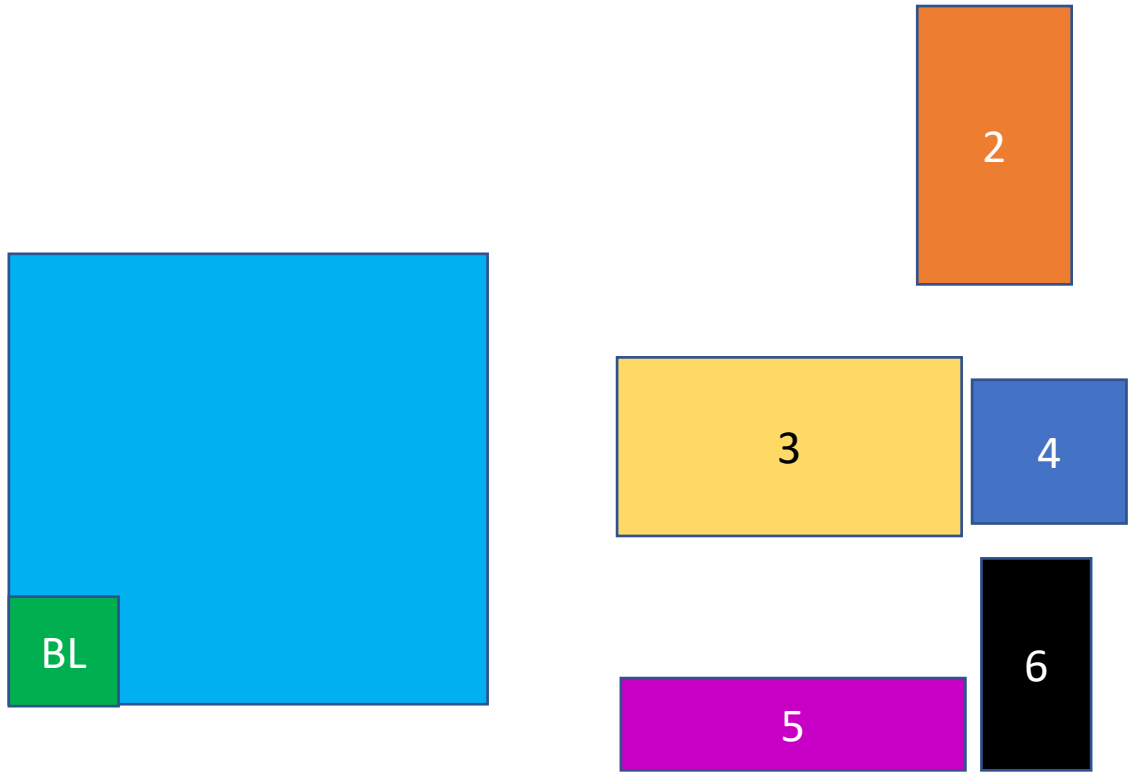
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

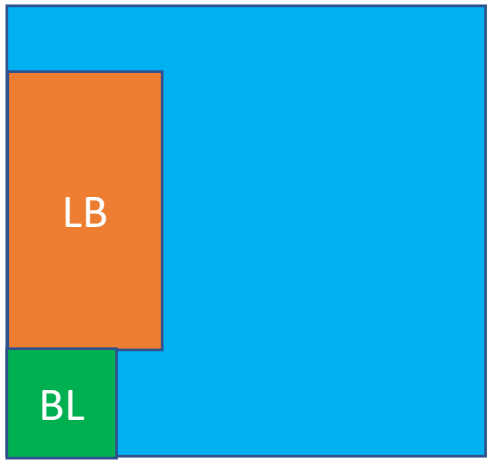
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

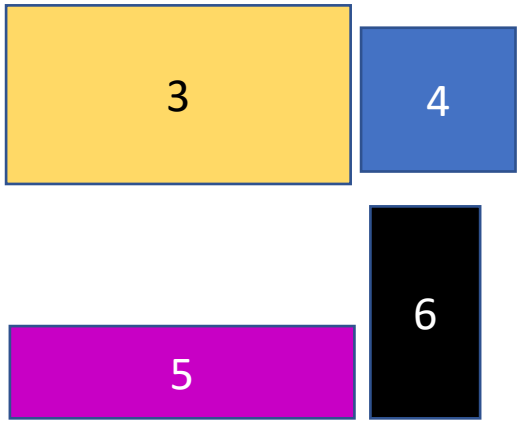
$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle





Decoders

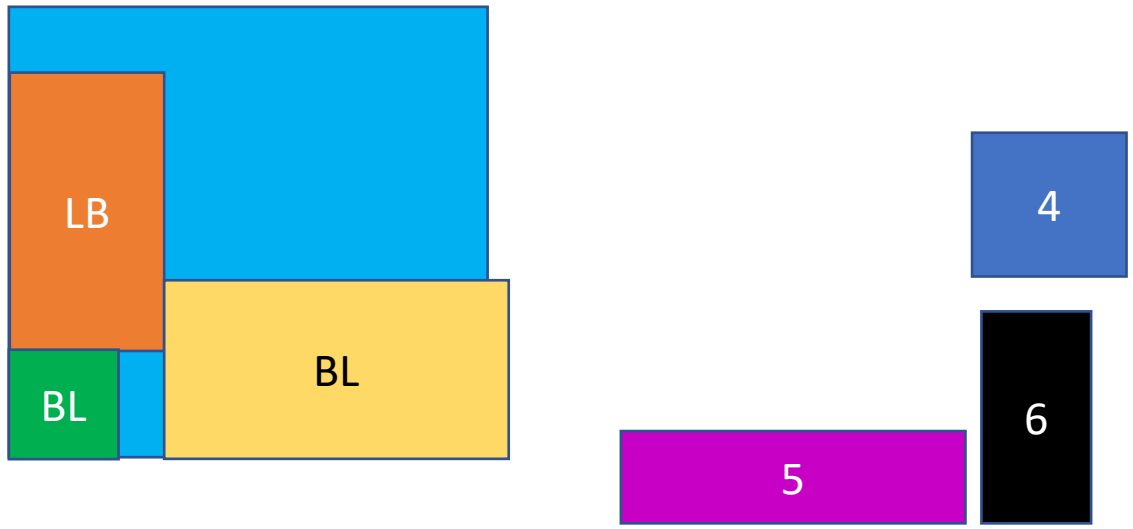
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

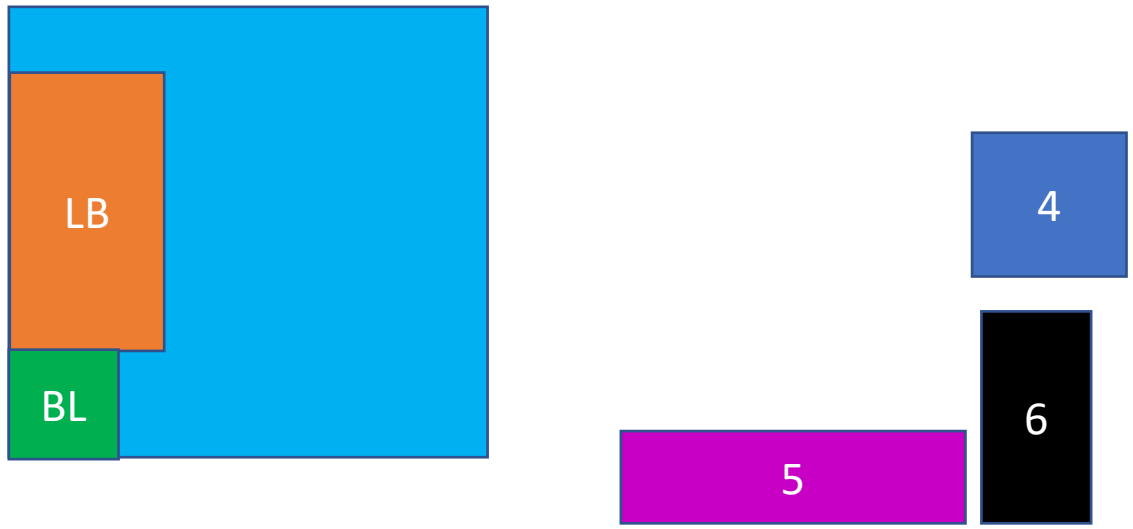
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

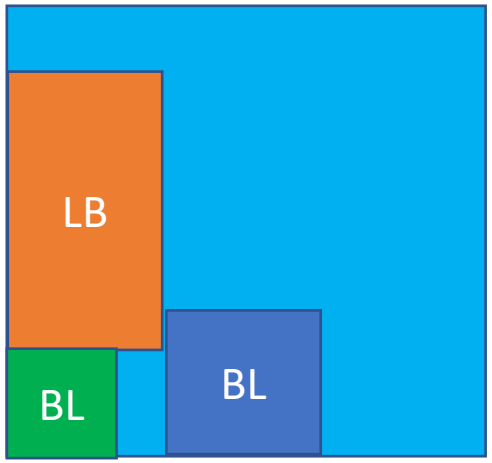
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

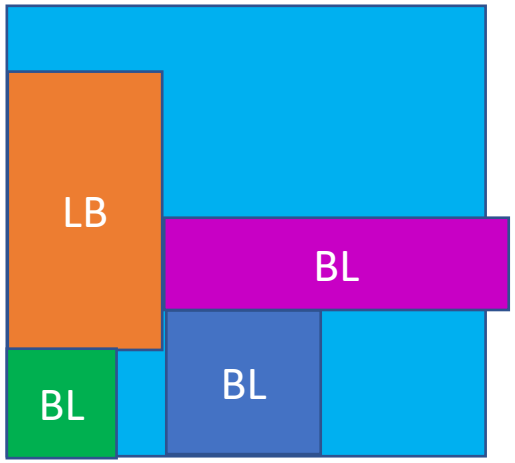
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

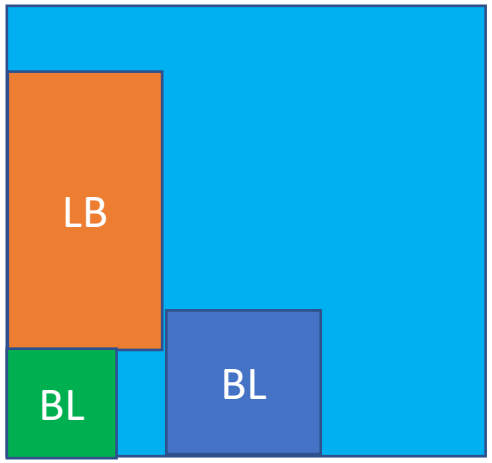
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle



Decoders

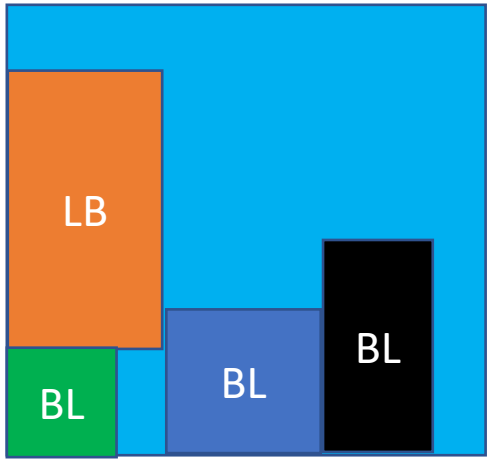
A decoder is a deterministic algorithm that takes as input a vector of N random keys and output a solution to the combinatorial optimization problem

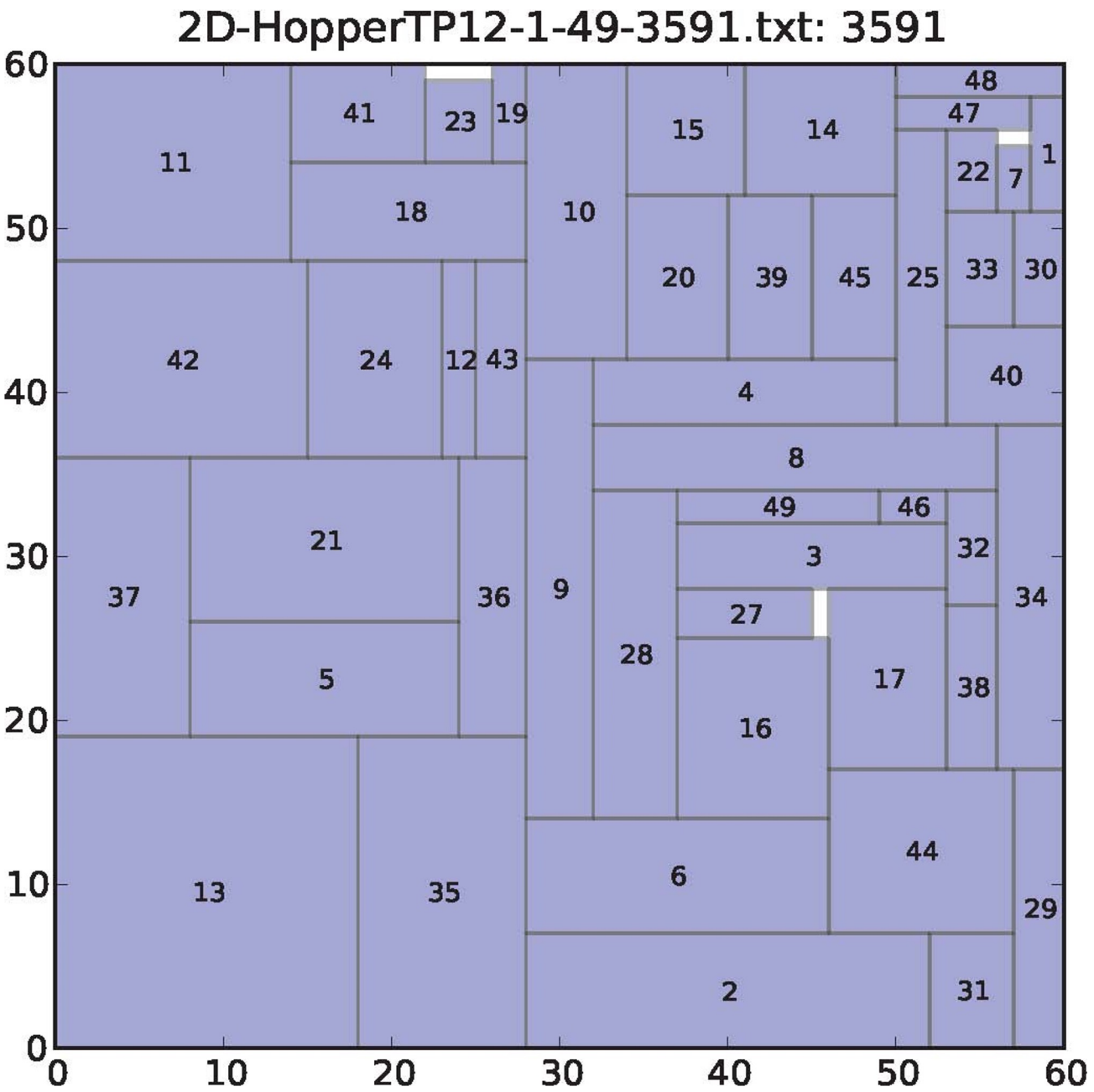
$$X = (.92, .15, .28, .05, .33, .59 \mid .22, .65, .08, .05, .38, .19)$$

Constrained two-dimensional orthogonal packing (Gonçalves & Resende, 2011)

Phase 1: Use first half of RKs to order rectangles

Phase 2: Use second half of RKs to determine if Bottom-Left or Left-Bottom rule is used to place rectangle





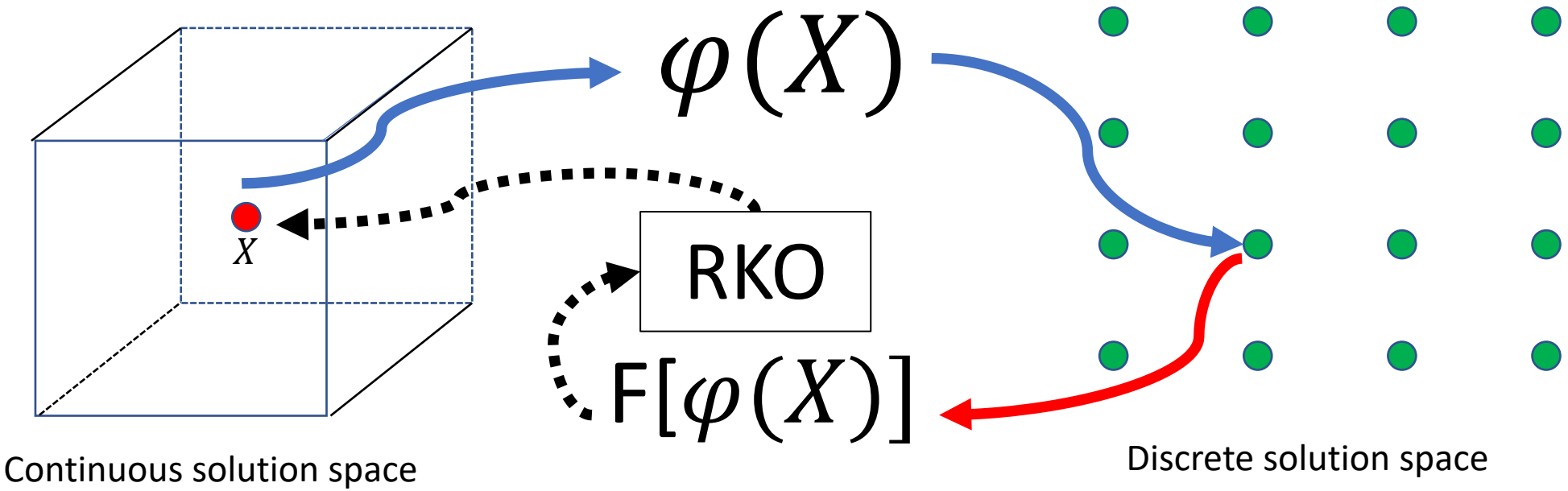
Hopper & Turton, 2001
Instance 4-2 60 x 60
Value: 3591

New best known solution found with a RKO (BRKGA, Gonçalves & Resende (2011))

Previous best: 3580 by a Tabu Search heuristic (Alvarez-Valdes et al., 2007)

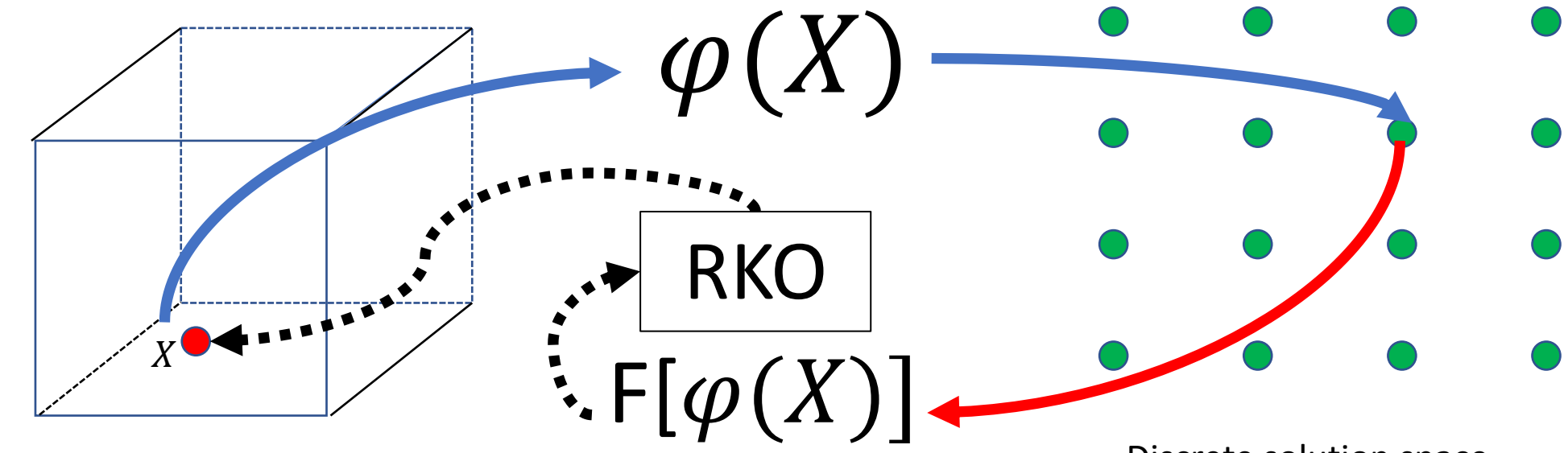
Random-Key Optimizers

- A random-key optimizer is a
  - Derivative-free optimization algorithm
  - Domain is N-dimensional real (continuous) hypercube
  - Function evaluation accomplished by decoder



Random-Key Optimizers

- A random-key optimizer is a
  - Derivative-free optimization algorithm
  - Domain is N-dimensional real (continuous) hypercube
  - Function evaluation accomplished by decoder



Random-Key Optimizers

- One algorithm, many decoders
  - Problem-independent algorithm for combinatorial optimization
  - Problem dependency is solely confined to decoder
- Easy implementation: solver is developed once (API, library)
- User only needs to develop decoder and set parameters of solver

Some Examples of Random-Key Optimizers

- Random-Key GA (Bean, 1994)
- Biased Random-Key GA (Gonçalves & Resende, 2011)
- Dual annealing (Schuetz et al., 2022)
- Continuous GRASP (Hirsch et al., 2007)
- Differential Evolution (Storm and Price, 1997)

Thanks

resendem@amazon.com
mgcr@berkeley.edu