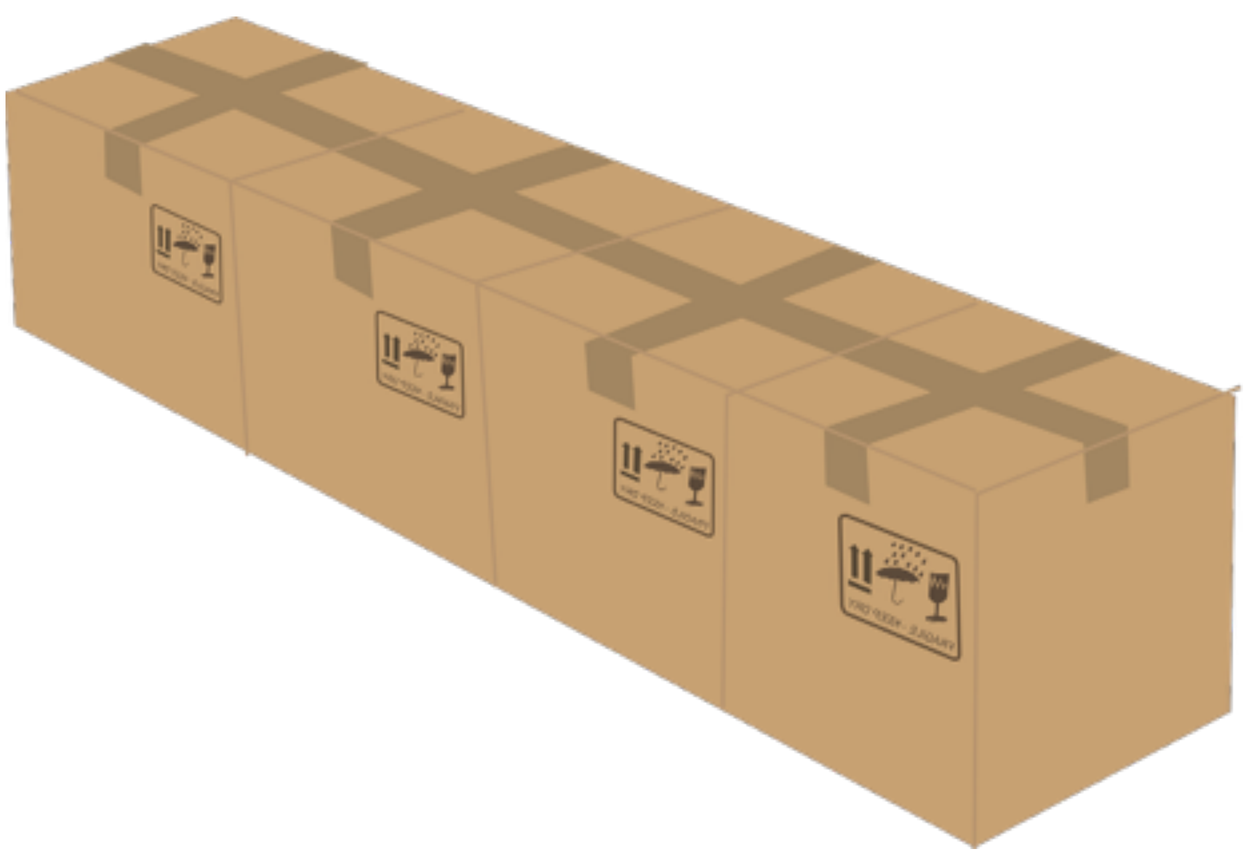
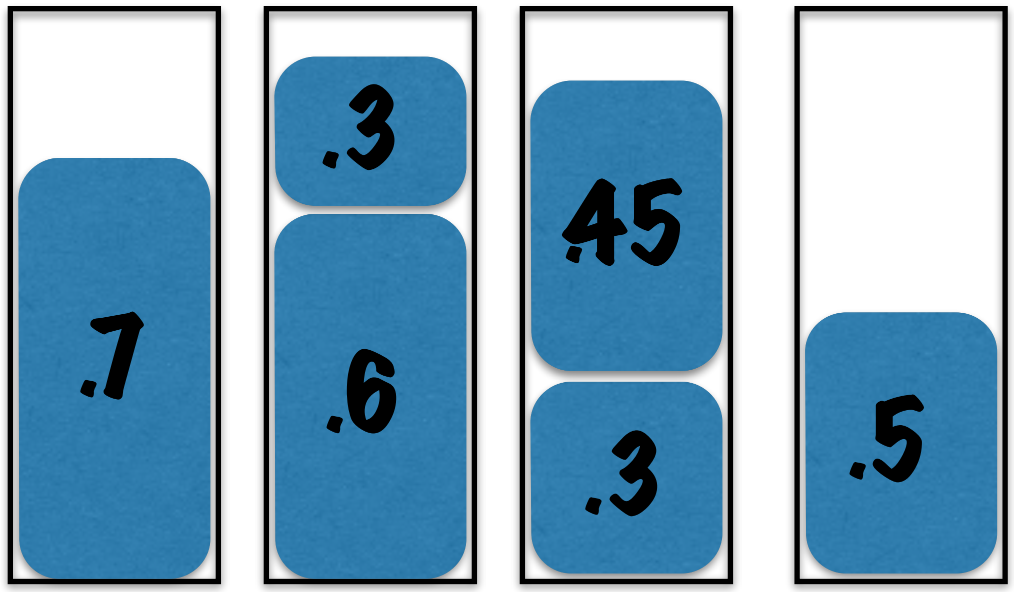
Bin packing, linear programming and rounding





Algorithm - large items

Assume: sizes > capacity * ϵ

Sort sizes

Make groups of cardinality $n \times \epsilon^2$

Round up to max size in group

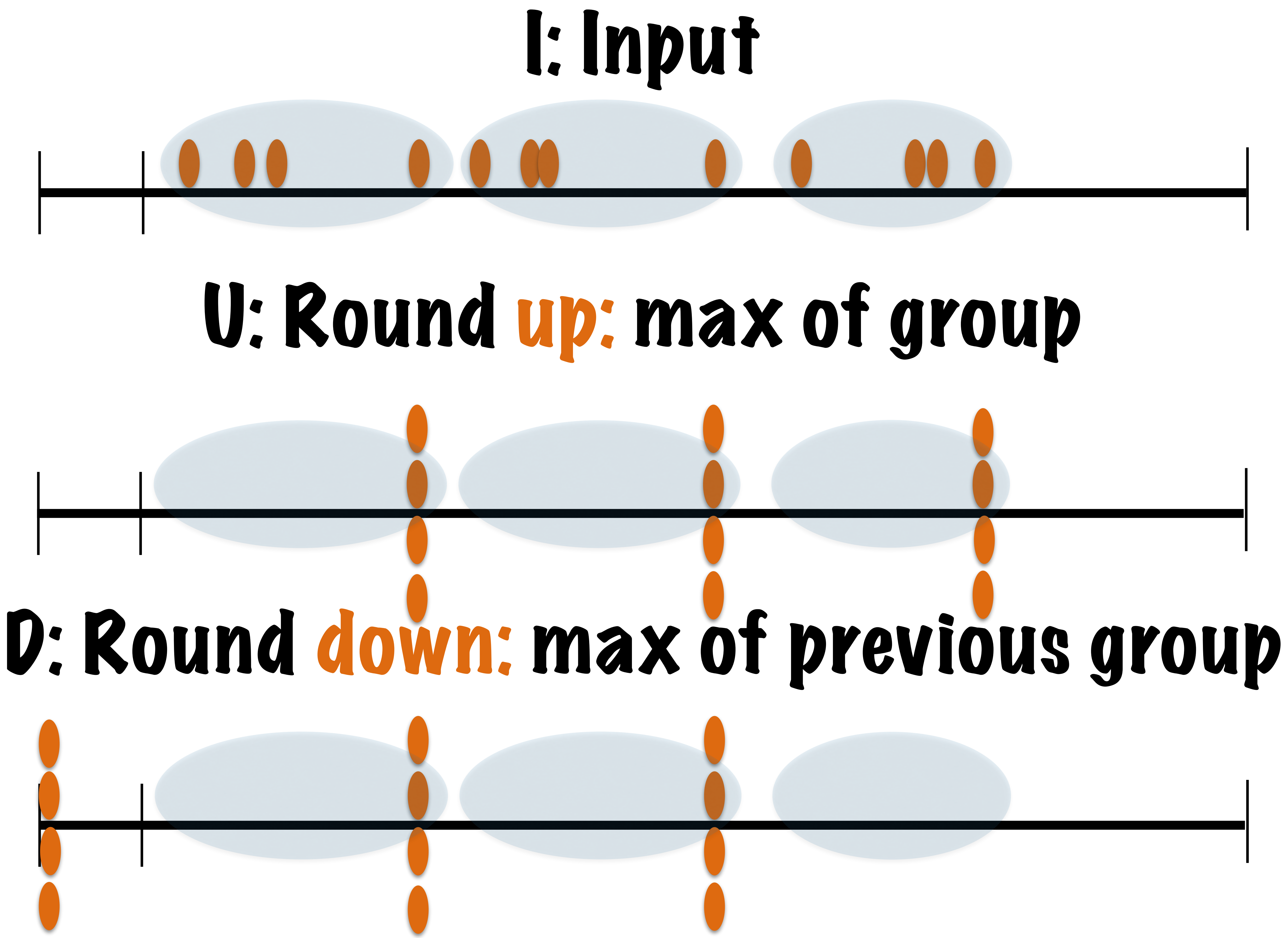
Solve rounded problem U

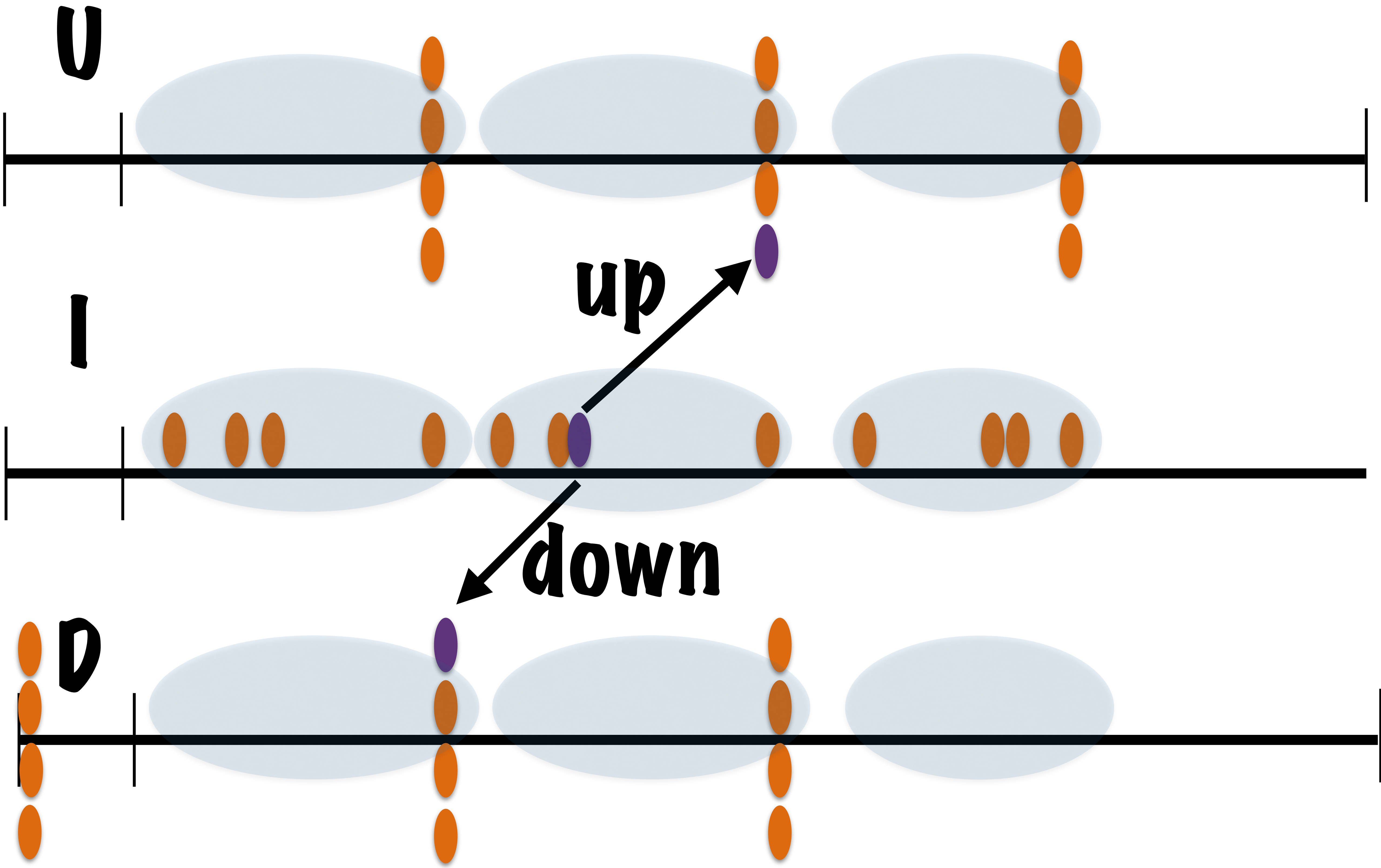
Output corresponding packing

But how good is it?

Value(Output) = OPT(U)

Must relate OPT(U) to OPT(I)


I: Input
U: Round up: max of group
D: Round down: max of previous group

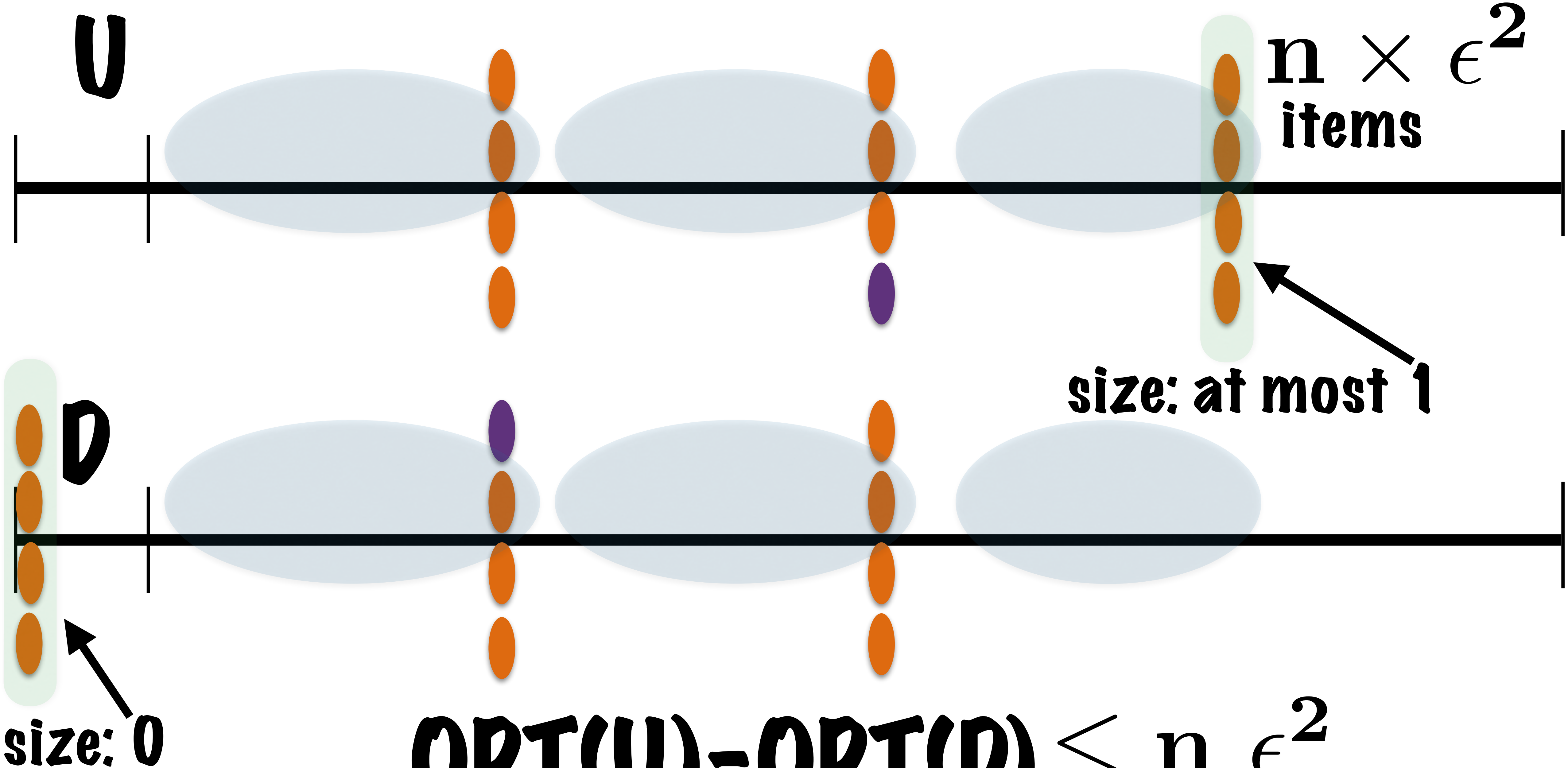

U
up
I
down
D

Relating input to rounded input

Observe:
Increasing sizes
can only increase OPT

$$OPT(D) \leq OPT(I) \leq OPT(U)$$

U and D are similar!



OPT(U)-OPT(D) $\leq n\ \epsilon^2$

Combine:

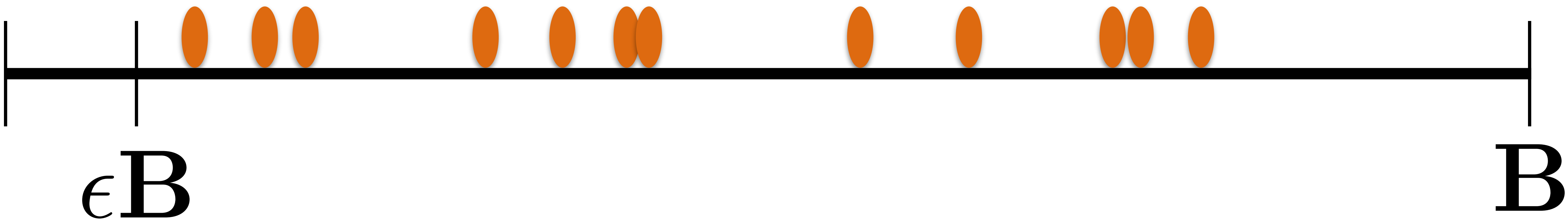
$$\mathrm{OPT(D)} \leq \mathrm{OPT(I)} \leq \mathrm{OPT(U)}$$

$$\mathrm{OPT(U)}-\mathrm{OPT(D)} \leq \mathbf{n}\ \epsilon^{\mathbf{2}}$$

$$\mathrm{OPT(U)} \leq \mathrm{OPT(I)} + \mathbf{n}\ \epsilon^{\mathbf{2}}$$

Additive error $\mathbf{n}\ \epsilon^{\mathbf{2}}$

Lower bound OPT



n items

max #items per bin: $1/\epsilon$

---

min #bins: ϵn

$$n\epsilon^2 \leq \epsilon \times (n\epsilon) \leq \epsilon \, \mathrm{OPT}$$

Theorem

When all sizes are > $\epsilon \mathbf{B}$
algorithm, in polynomial time
gives packing s.t.
Value(Output) < OPT * $(\mathbf{1} + \epsilon)$

Bin packing, linear programming and rounding