Fractional Knapsack Problem
&
0-1 Knapsack Problem

Fractional Knapsack Problem

- Given weights and values of n items, we need to put these items in a knapsack of capacity W to get the maximum total value in the knapsack.

- In Fractional Knapsack, we can break items for maximizing the total value of knapsack.

Fractional Knapsack Problem

- Greedy approach.

Fractional Knapsack Problem

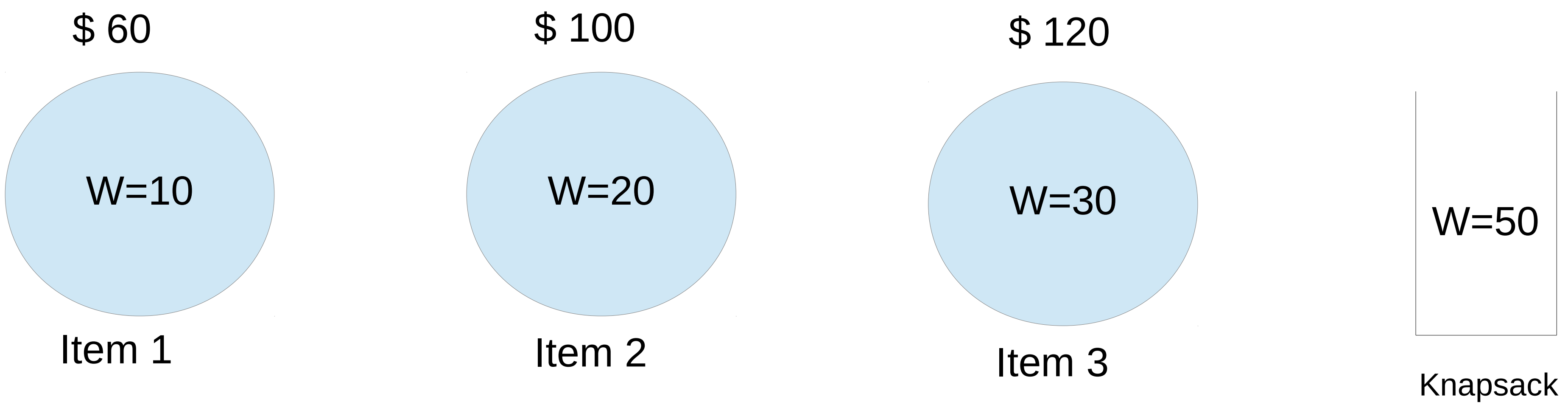
- Greedy approach.

robbery ....

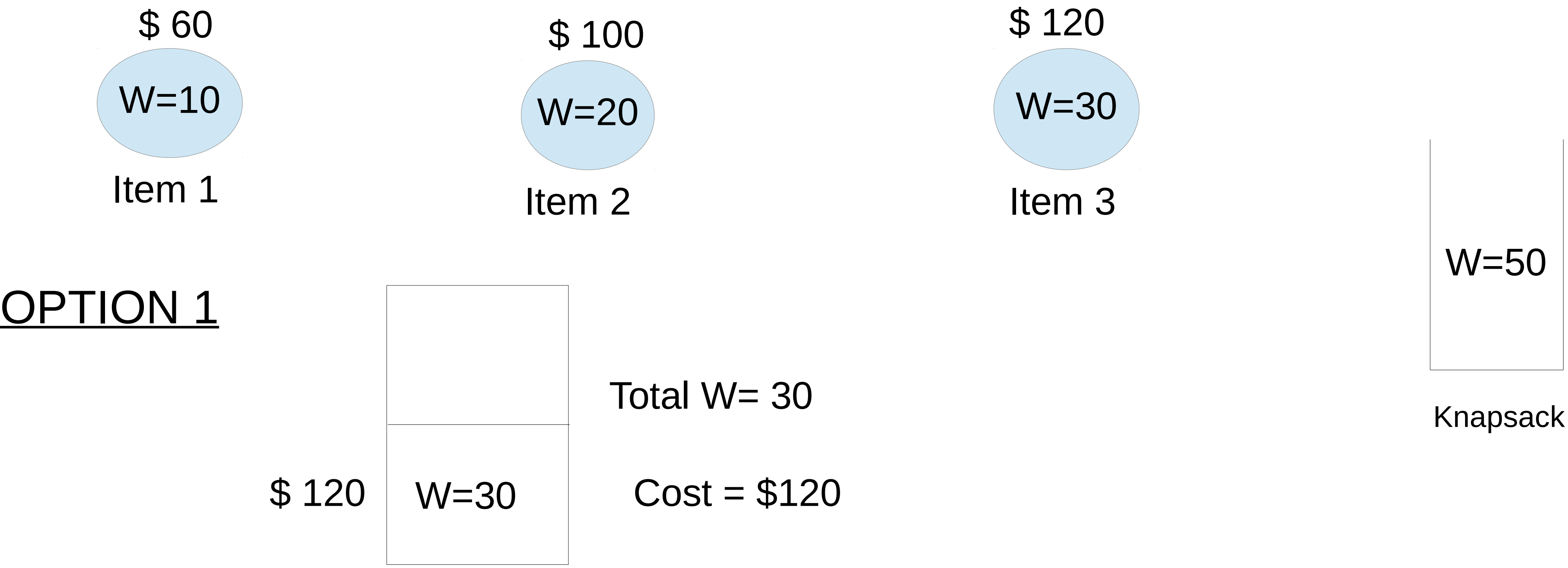
Fractional Knapsack Problem

- Greedy approach.
-  robbery ....

Fractional Knapsack Problem



Fractional Knapsack Problem



Fractional Knapsack Problem

$ 60

W=10

Item 1

$ 100

W=20

Item 2

$ 120

W=30

Item 3

W=50

Knapsack

<u>OPTION 1</u>

$ 100 W=20

$ 120 W=30

Total W= 50

Cost = $120 + $100
= $220

Fractional Knapsack Problem

$ 60

W=10

Item 1

$ 100

W=20

Item 2

$ 120

W=30

Item 3

W=50

Knapsack

<u>OPTION 2</u>

$ 60 W=10

Total W= 10

Cost = $60

Fractional Knapsack Problem

$ 60
W=10
Item 1

$ 100
W=20
Item 2

$ 120
W=30
Item 3

W=50
Knapsack

<u>OPTION 2</u>

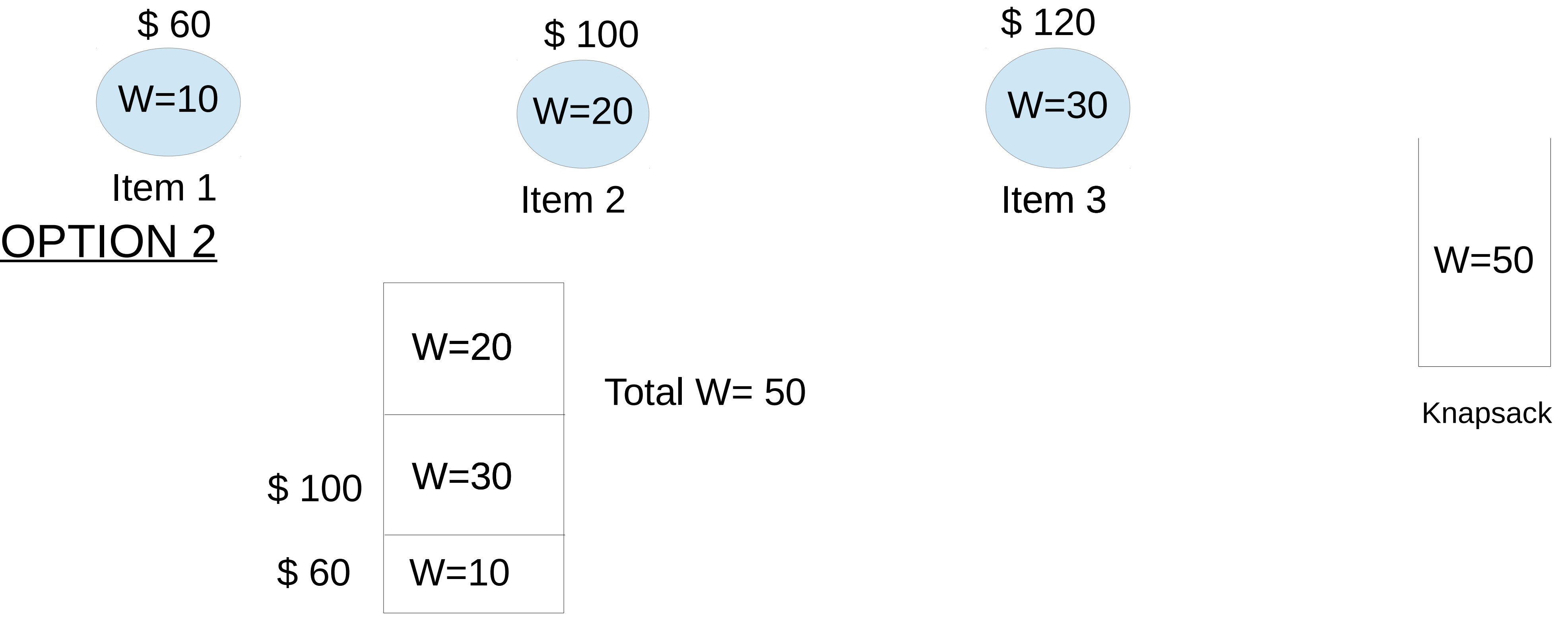
$ 100 W=20

$ 60 W=10

Total W= 30

Cost = $60 + $100

Fractional Knapsack Problem



Fractional Knapsack Problem

$ 60

W=10

Item 1

$ 100

W=20

Item 2

$ 120

W=30

Item 3

W=50

Knapsack

<u>OPTION 2</u>

$ 120/30 * 20 = $80	W=20 out of 30
$ 100	W=30
$ 60	W=10

Total W= 50

Cost = $60 + $100 + $80
= $240

Fractional Knapsack Problem

- An efficient solution is to use Greedy approach.

- The basic idea of the greedy approach is to calculate the ratio value/weight for each item and sort the item on basis of this ratio. Then take the item with the highest ratio and add them until we can't add the next item as a whole and at the end add the next item as much as we can.

- Which will always be the optimal solution to this problem.

Fractional Knapsack Problem

- In this case, items can be broken into smaller pieces, hence we can select fractions of items. According to the problem statement,

There are n items in the store

- Weight of i^{th} item $w_i>0$
- Profit for i^{th} item $p_i>0$ and
- Capacity of the Knapsack is W

- So, we may take only a fraction x_i of i^{th} item. $0 \leqslant xi \leqslant 1$

The i^{th} item contributes the weight xi.wi to the total weight in the knapsack and profit xi.pi to the total profit.

- Hence, the objective of this algorithm is to

Maximize $\sum(xi.pi)$

subject to constraint,

$\sum(xi.wi) \leqslant W$

Thus, an optimal solution can be obtained by

$\sum(xi.wi)=W$

0-1 Knapsack Problem

0-1 Knapsack problem

0-1 Knapsack Problem

- In the 0-1 Knapsack problem, we are not allowed to break items. We either take the whole item or don’t take it.

0-1 Knapsack Problem

Item 1	$ 60	W=10
Item 2	$ 100	W=20
Item 3	$ 120	W=30

W=50

Knapsack

<u>OPTION 1</u>

$ 100	W=20
$ 120	W=30

Total W= 50

Cost = $120 + $100
= $220

Thank You