Online k-Median with Consistent Clusters

Heather Newman (Carnegie Mellon)

APPROX 2024

Joint work with: Benjamin Moseley (Carnegie Mellon) and Kirk Pruhs (U. of Pittsburgh)

Offline k-Median

Offline k-Median

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Offline k-Median

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Offline k-Median

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Offline k-Median

Input:

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

Offline k-Median

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$

Offline k-Median

Input:

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Offline k-Median

Input:

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Offline k-Median

Input:

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Offline k-Median

Input:

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

Offline k-Median

Input:

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$

Offline k-Median

Input:

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Offline k-Median

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Offline k-Median

Input:

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Constant-factor approximations exist

Offline k-Median

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

(offline) =

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Online Offline k-Median

Input:

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Online ~~Offline~~ k-Median

Input:

$$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$$

$x_1, \ldots, x_n$ lying in metric space

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Online ~~Offline~~ k-Median

Input:

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

$x_1, \ldots, x_n$ lying in metric space ***arrive over time***

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Online ~~Offline~~ k-Median

Input:

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

$x_1, \ldots, x_n$ lying in metric space ***arrive over time***

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i center on arrival

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Online ~~Offline~~ k-Median

$\min \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)$

Input:

$x_1, \ldots, x_n$ lying in metric space ***arrive over time***

(small) k = #clusters = #labels

Two perspectives on output

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i center on arrival

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Give x_i label in $[k]$ on arrival

Online ~~Offline~~ k-Median

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Give x_i a label in $[k]$ on arrival

Online ~~Offline~~ k-Median

Maximizing Quality

$O(1)$ competitive ratio

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Give x_i a label in $[k]$ on arrival

Online ~~Offline~~ k-Median

Maximizing Quality
$O(1)$ competitive ratio

Maximizing Consistency
no changes to centers (center-based) or labels (cluster-based)

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Cluster-based clustering

- Output: clusters $C_1, \ldots, C_k$
- Centers c_i implicit

Give x_i a label in $[k]$ on arrival

Online ~~Offline~~ k-Median

Maximizing Quality
$O(1)$ competitive ratio

Maximizing Consistency
no changes to centers (center-based) or labels (cluster-based)

Center-based clustering

- Output: centers $c_1, \dots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Cluster-based clustering

- Output: clusters $C_1, \dots, C_k$
- Centers c_i implicit

Give x_i a label in $[k]$ on arrival

Online ~~Offline~~ k-Median

Maximizing Quality

$O(1)$ competitive ratio

Maximizing Consistency

no changes to centers (center-based) or labels (cluster-based)

Online ~~Offline~~ k-Median

Maximizing Quality

$O(1)$ competitive ratio

Maximizing Consistency

no changes to centers (center-based) or labels (cluster-based)

$k = 2$

Online ~~Offline~~ k-Median

Maximizing Quality

$O(1)$ competitive ratio

Maximizing Consistency

no changes to centers (center-based) or labels (cluster-based)

$k = 2$

Online ~~Offline~~ k-Median

Maximizing Quality
$O(1)$ competitive ratio

$\longleftrightarrow$

Maximizing Consistency
no changes to centers (center-based) or labels (cluster-based)

$k = 2$

Online ~~Offline~~ k-Median

Maximizing Quality

$O(1)$ competitive ratio

Maximizing Consistency

no changes to centers (center-based) or labels (cluster-based)

$k = 2$

Online ~~Offline~~ k-Median

Maximizing Quality

$O(1)$ competitive ratio

Maximizing Consistency

no changes to centers (center-based) or labels (cluster-based)

$k = 2$

Upshot: competitive ratio must depend on aspect ratio $\Delta \gg n$ **if choices are irrevocable**

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17;

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \dots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17; Fichtenberger et. al., '21)

Prior Work: Online k-Median

Maximizing Quality ⟷ Maximizing Consistency

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17; Fichtenberger et. al., '21)

- Change centers small number of times

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

- Output: centers $c_1, \ldots, c_k$
- Clusters C_i (points labelled i) implicit

Decide if x_i a center on arrival

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17; Fichtenberger et. al., '21)

- Change centers small number of times
- $O(1)$-competitive, $O(k \operatorname{poly} \log(n\Delta))$ center **changes**

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17; Fichtenberger et. al., '21)

- Change centers small number of times
- $O(1)$-competitive, $O(k \text{ poly} \log(n\Delta))$ center **changes**

Prior Work: Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

Both use a randomized subroutine for online facility location (Meyerson '01)

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17; Fichtenberger et. al., '21)

- Change centers small number of times
- $O(1)$-competitive, $O(k \operatorname{poly} \log(n\Delta))$ center **changes**

Maximizing Quality ⟷ Maximizing Consistency

Beyond worst-case approaches?

Center-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17)

- Change centers small number of times
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency**

Beyond worst-case approaches?

Center-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17)

- Change centers small number of times
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ Maximizing Consistency

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17)

- Change centers small number of times
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Our Work: *Consistent* Online k-Median

Maximizing Quality $\longleftrightarrow$ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ centers, i.e., bi-criteria approx.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17)

- Change centers small number of times
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen … x.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Use at most k labels

Recourse (Lattanzi & Vassilvitskii, '17)

- Change centers small number of times
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen… x.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Use at most k labels

Recourse (Lattanzi & Vassilvitskii, '17)

- Change c… imes
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Never relabel points

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen… x.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Use at most k labels

Recourse (Lattanzi & Vassilvitskii, '17)

- Change c… imes
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center changes

Never relabel points

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen... x.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Use at most k labels

Recourse (Lattanzi & Vassilvitskii, '17)

- Change c... imes
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Never relabel points

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen… x.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Use at most k labels

Recourse (Lattanzi & Vassilvitskii, '17)

- Change c… imes
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Never relabel points

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

BUT! Lower bd $\Longrightarrow$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen… x.
- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Use at most k labels

Recourse (Lattanzi & Vassilvitskii, '17)

- Change c… imes
- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center changes

Never relabel points

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

BUT! Lower bd $\Longrightarrow$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Resource Augmentation (Liberty et. al., '16)

- $> k$ cen... x.

Use at most k labels

- $O(\log n)$-competitive, $O(k \log n \log n\Delta)$ centers

Recourse (Lattanzi & Vassilvitskii, '17)

- Change c... imes

Never relabel points

- $O(1)$-competitive, $O(k^2 \log^4 n\Delta)$ center **changes**

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~**Center-based clustering**~~

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: $\text{ALG} \leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~**Center-based clustering**~~

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

Our Work: *Consistent* Online k-Median

Maximizing Quality $\longleftrightarrow$ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~**Center-based clustering**~~

Cluster-based clustering

BUT! Lower bd $\Longrightarrow$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs

Our Work: *Consistent* Online k-Median

Maximizing Quality $\longleftrightarrow$ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~**Center-based clustering**~~

Cluster-based clustering

BUT! Lower bd $\Longrightarrow$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~Center-based clustering~~

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance
- **Natural** information about instance

Our Work: <u>*Consistent*</u> Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~**Center-based clustering**~~

Cluster-based clustering

BUT! Lower bd $\Longrightarrow$ need some info *a priori*

<u>Given:</u> "budget" B where $B \geq$ (final) OPT

<u>Objective:</u> $\mathrm{ALG} \leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance
- **Natural** information about instance

Prior techniques (Meyerson) help?

Our Work: *Consistent* Online k-Median

Maximizing Quality ⟷ **Maximizing Consistency** ✓

Beyond worst-case approaches?

~~**Center-based clustering**~~

Cluster-based clustering

BUT! Lower bd $\Longrightarrow$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance
- **Natural** information about instance

Prior techniques (Meyerson) help?

Seemingly no

Our Work: *Consistent* Online k-Median

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: $\mathrm{ALG} \leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance
- **Natural** information about instance

Our Work: *Consistent* Online k-Median

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance
- **Natural** information about instance

Our Work: *Consistent* Online k-Median

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

Lower Bound: Dependence on k is necessary: cost = $\Omega(k \cdot B)$.

Cluster-based clustering

BUT! Lower bd $\implies$ need some info *a priori*

Given: "budget" B where $B \geq$ (final) OPT

Objective: ALG $\leq f(k) \cdot B$

➡ **No dependence on n or Δ!**

Why B?

- Learn **scale** of costs
- **Minimal** information about instance
- **Natural** information about instance

Attempt 1: How to use B?

Attempt 1: How to use B?

Natural candidate greedy algo (using B):

Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels “justified” (w.r.t. B)

Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels "justified" (w.r.t. B)

Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels "justified" (w.r.t. B)

$k = 2, B = 2$

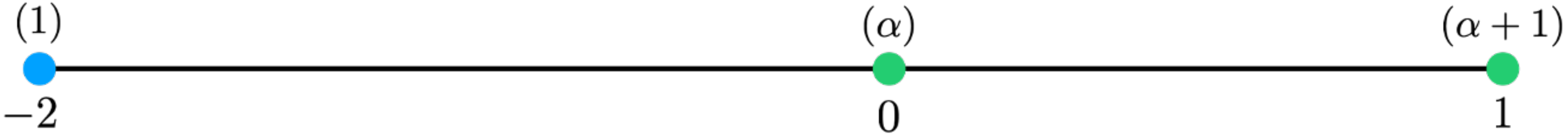
Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels “justified” (w.r.t. B)

$k = 2, B = 2$



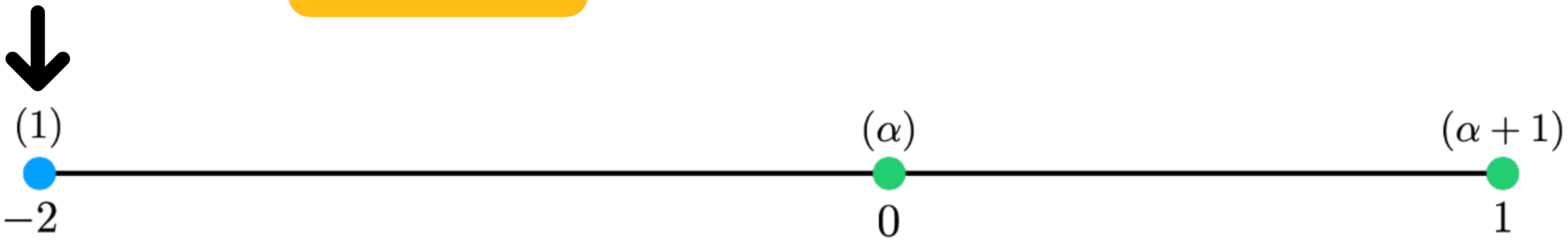
Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels “justified” (w.r.t. B)

$k = 2, B = 2$



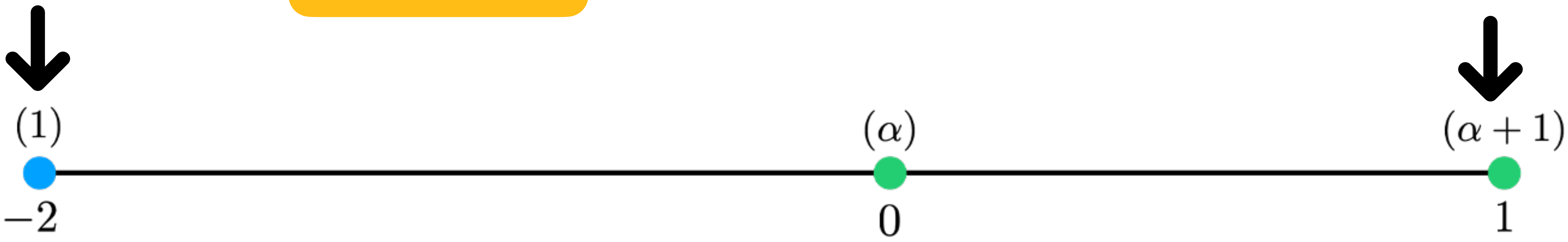
Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels “justified” (w.r.t. B)

$k = 2, B = 2$



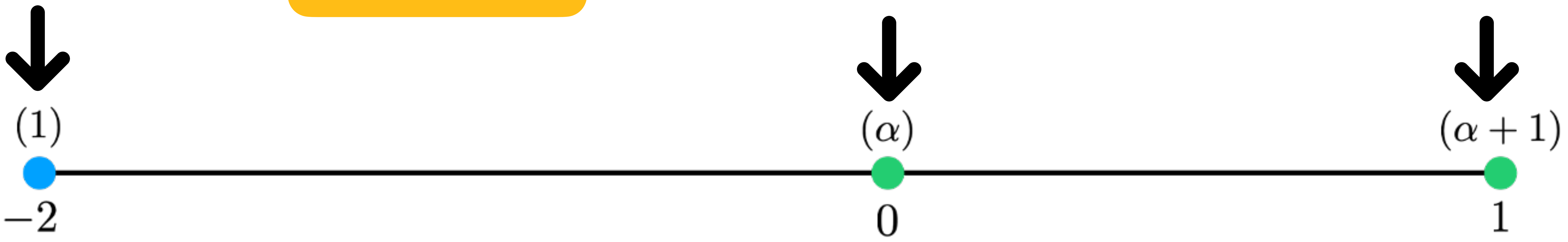
Attempt 1: How to use B?

Natural candidate greedy algo (using B):

give each data point the label minimizing increase in cost

S.T. only use up to number of labels “justified” (w.r.t. B)

$k = 2, B = 2$



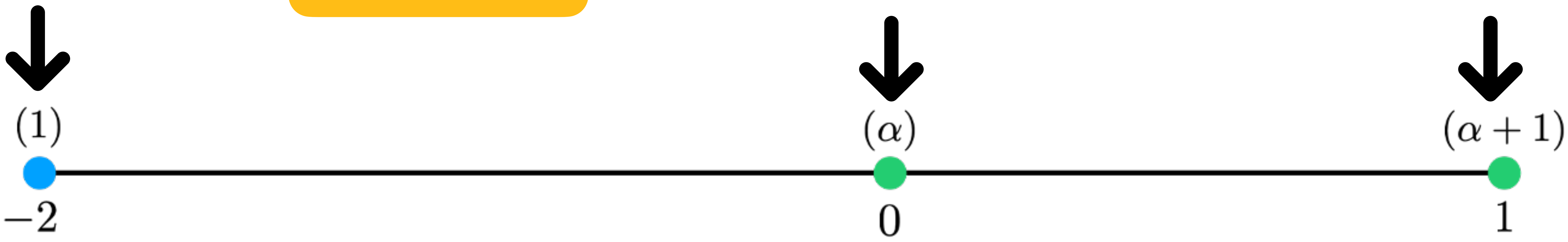
Attempt 1: How to use B?

Natural candidate greedy algo (using B):

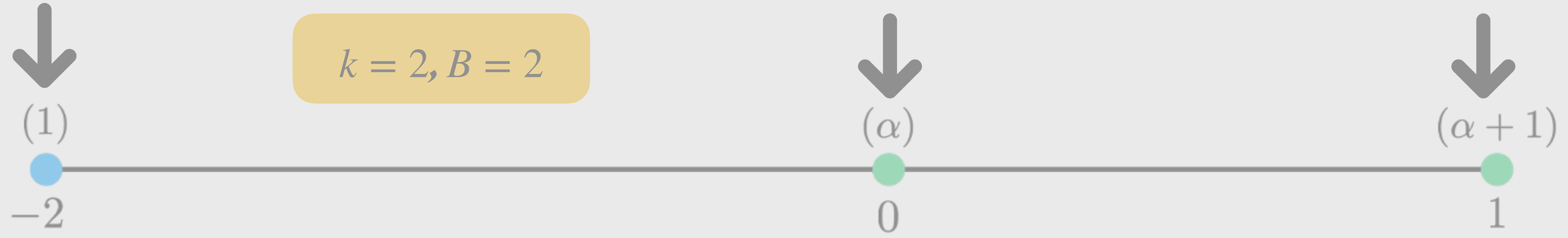
give each data point the label minimizing increase in cost

S.T. only use up to number of labels “justified” (w.r.t. B)

$k = 2, B = 2$

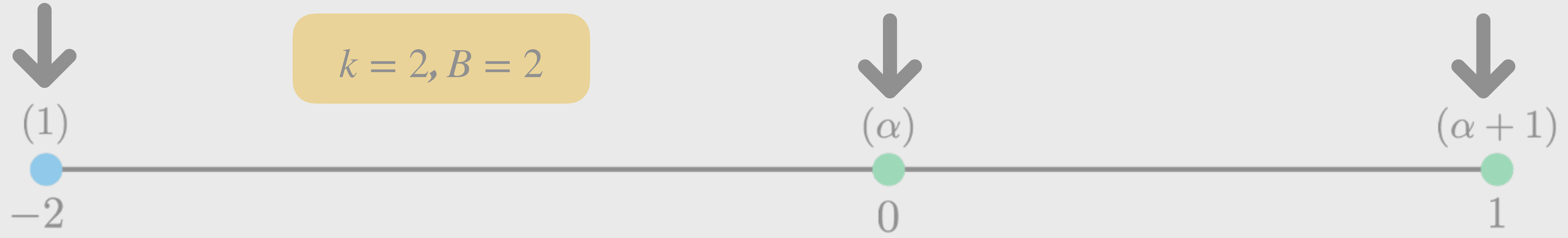


Upshot: this algo can have unbounded cost!



Upshot: this algo can have unbounded cost!

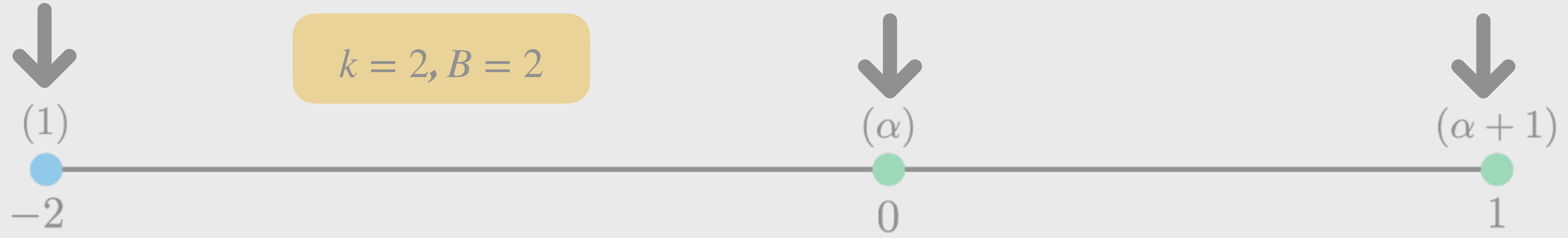
Can we still be greedy?



Upshot: this algo can have unbounded cost!

Can we still be greedy?

Q1: When to increase # labels?

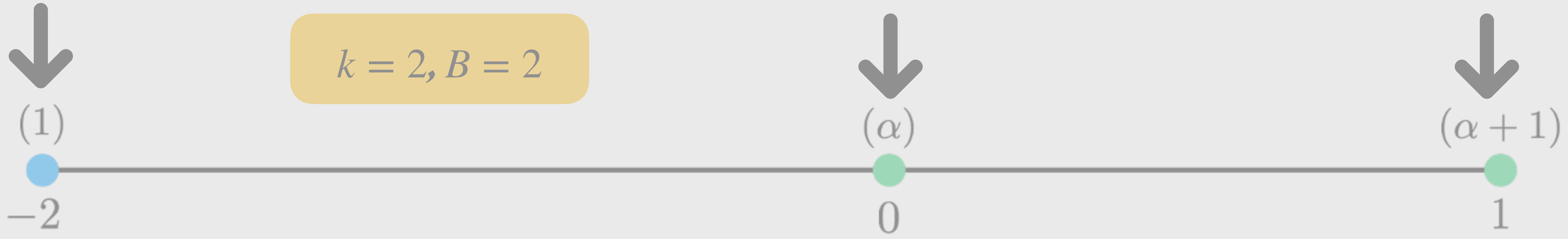


Upshot: this algo can have unbounded cost!

Can we still be greedy?

Q1: When to increase # labels?

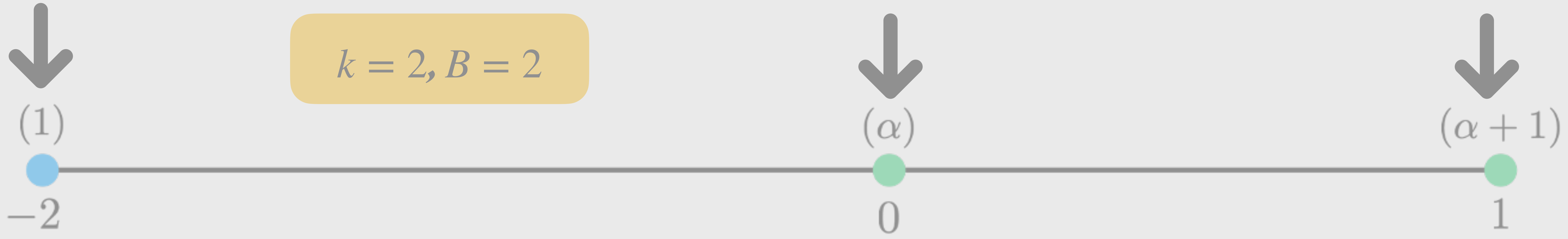
➡ Wait for more evidence of where **dense regions** are?



Upshot: this algo can have unbounded cost!

Can we still be greedy?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

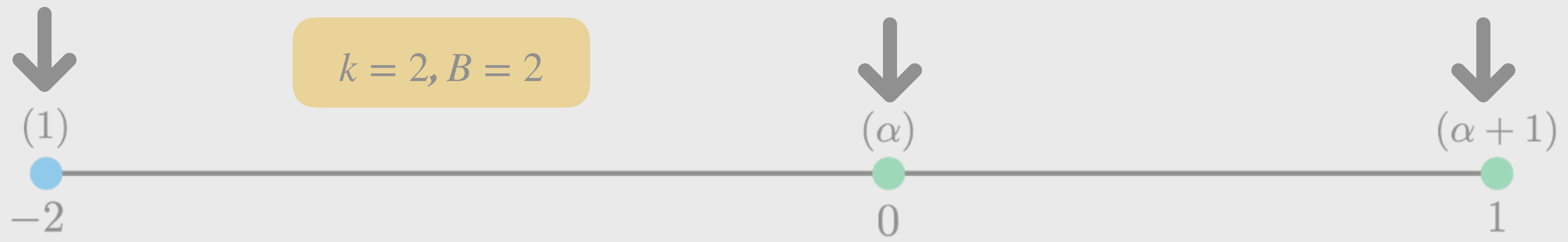


Upshot: this algo can have unbounded cost!

Can we still be greedy?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

Q2: Once we add label t, how to partition space from $t-1 \rightarrow t$ parts?

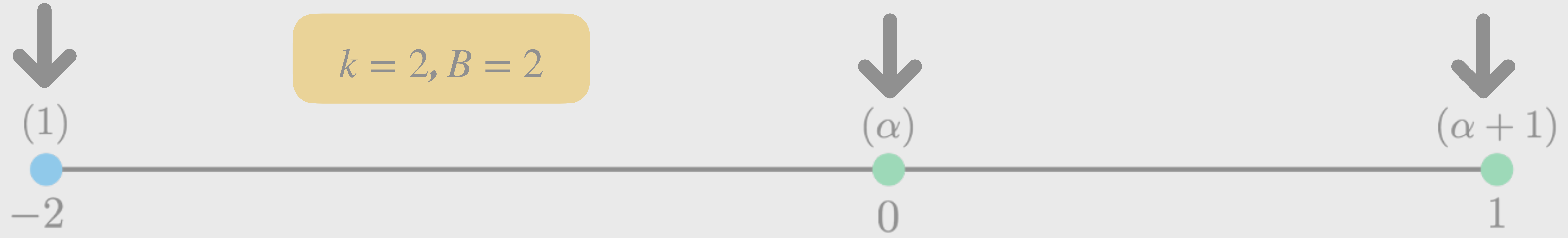


Upshot: this algo can have unbounded cost!

Can we still be greedy?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?
➡ Greedy = assign to "closest" part

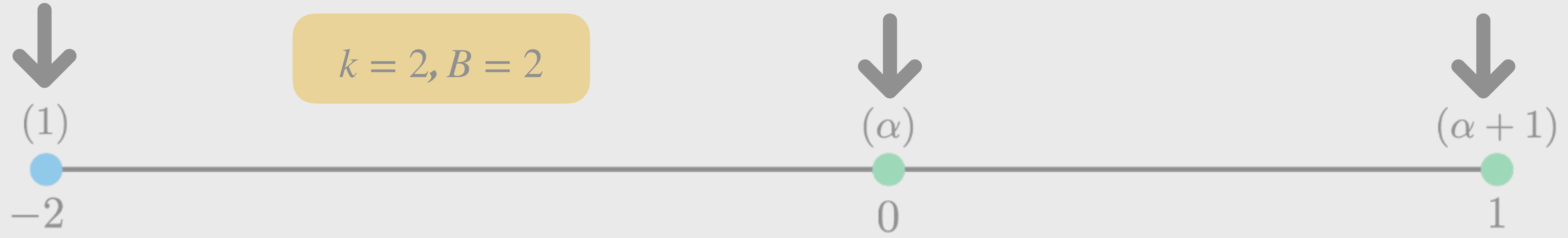


Upshot: this algo can have unbounded cost!

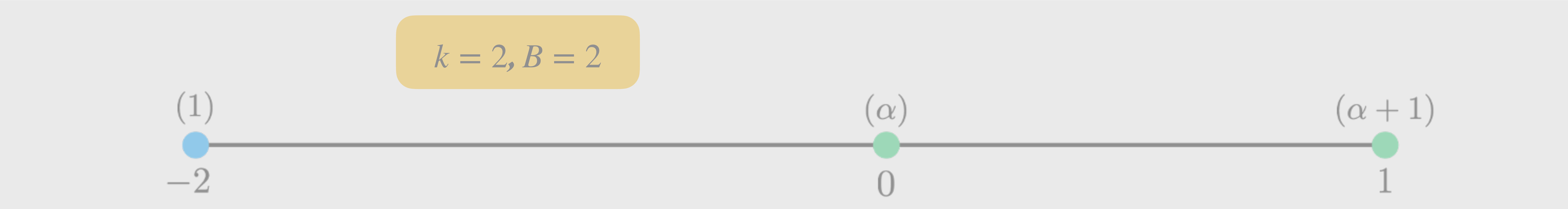
Can we still be greedy?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?
➡ Greedy = assign to "closest" part

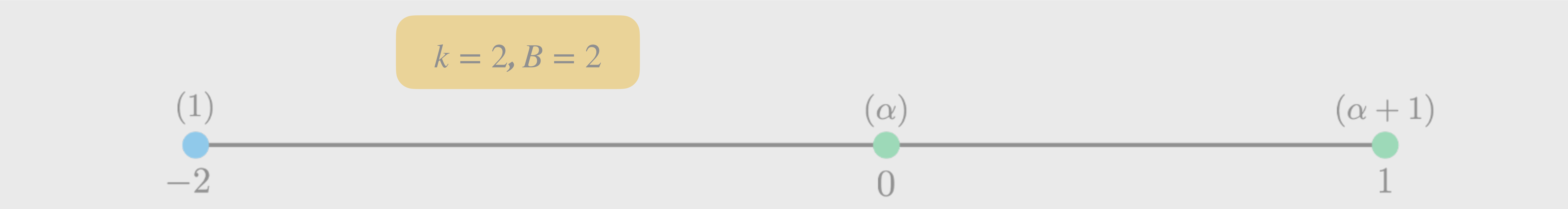


Upshot: this algo can have unbounded cost!



$k = 2, B = 2$
(1)
(α)
$(\alpha + 1)$
-2
0
1

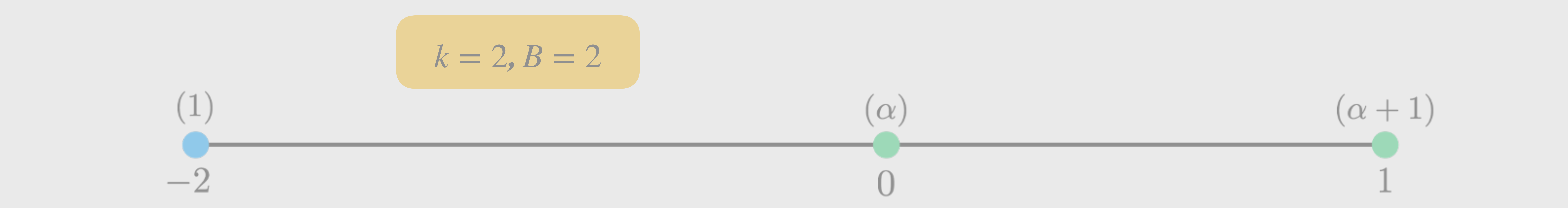
Attempt 2: How to use B?



Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?



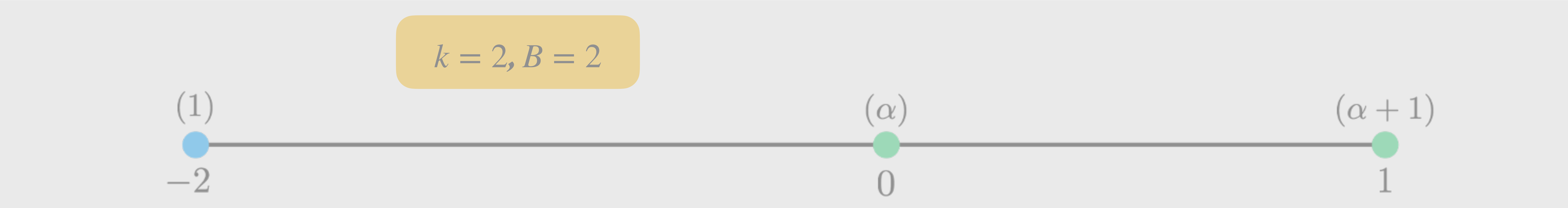
Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

natural weight of p:

$w(p) :=$ max # pts whose total distance to p is $\leq 2B$



Attempt 2: How to use B?

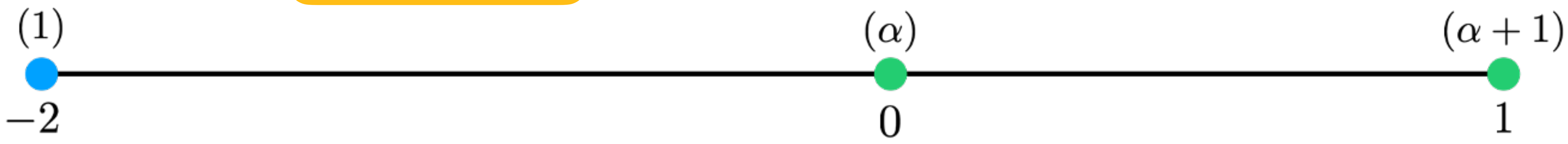
Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

natural weight of p:

$w(p)$:= max # pts whose total distance to p is $\leq 2B$

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase #labels?

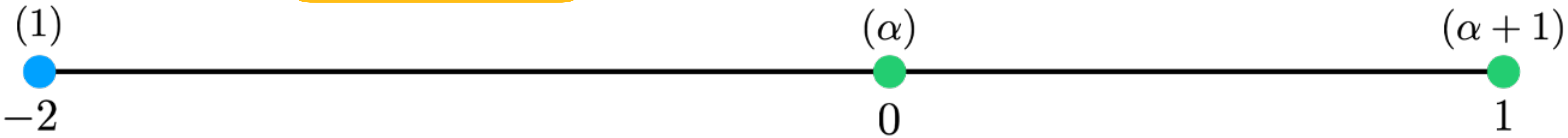
➡ Wait for more evidence of where **dense regions** are?

natural weight of p:

$w(p)$:= max # pts whose total distance to p is $\leq 2B$

$k = 2, B = 2$

$w(p)$ small



Attempt 2: How to use B?

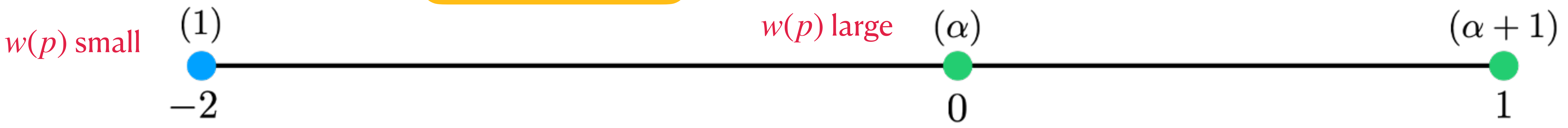
Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

natural weight of p:

$w(p)$:= max # pts whose total distance to p is $\leq 2B$

$k = 2, B = 2$



Attempt 2: How to use B?

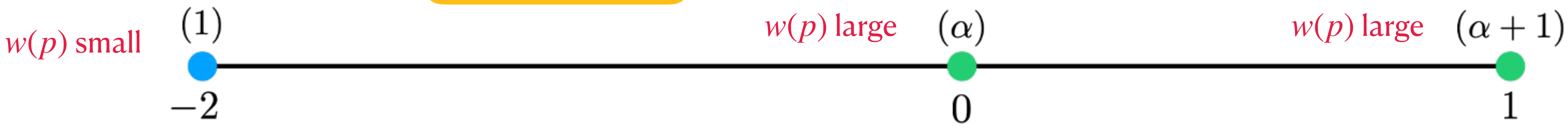
Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

natural weight of p:

$w(p)$:= max # pts whose total distance to p is $\leq 2B$

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

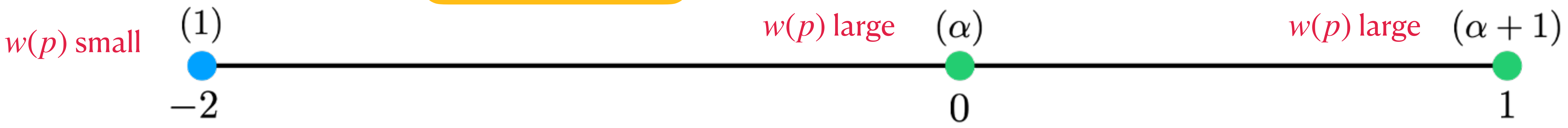
natural weight of p:

$w(p) :=$ max # pts whose total distance to p is $\leq 2B$

x, y are **β-well-separated** if far in weighted sense :

$$\min\{w(x), w(y)\} \cdot d(x, y) \geq \beta \cdot B$$

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

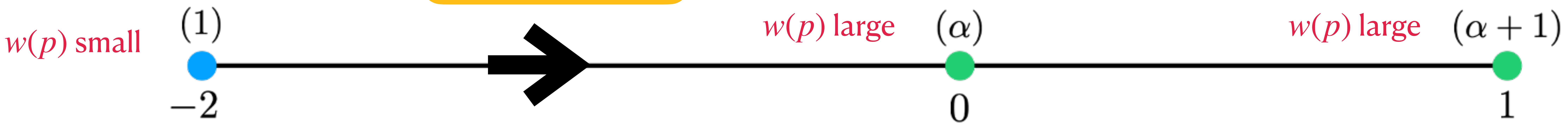
natural weight of p:

$w(p) :=$ max # pts whose total distance to p is $\leq 2B$

x, y are β**-well-separated** if far in weighted sense :

$$\min\{w(x), w(y)\} \cdot d(x, y) \geq \beta \cdot B$$

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

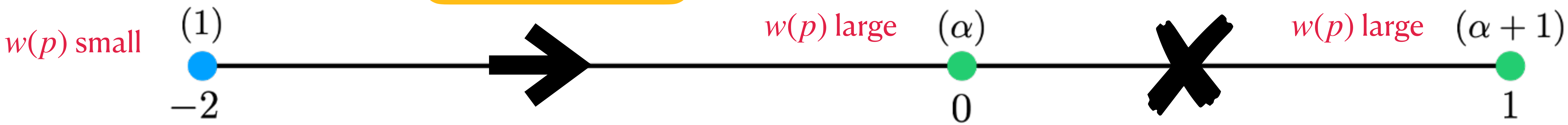
natural weight of p:

$w(p) :=$ max # pts whose total distance to p is $\leq 2B$

x, y are β**-well-separated** if far in weighted sense :

$$\min\{w(x), w(y)\} \cdot d(x, y) \geq \beta \cdot B$$

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

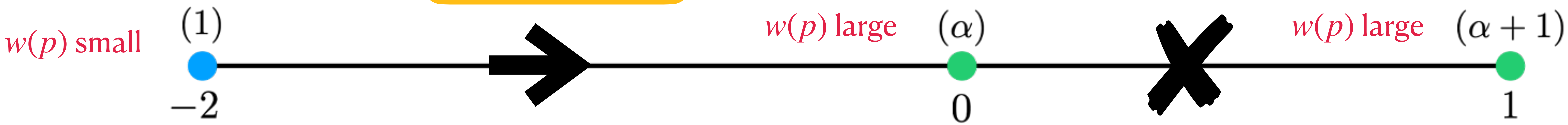
natural weight of p**:**

$w(p) :=$ max # pts whose total distance to p is $\leq 2B$

x, y are β-**well-separated** if far in weighted sense :

$$\min\{w(x), w(y)\} \cdot d(x, y) \geq \beta \cdot B$$

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase #labels?

➡ Wait for more evidence of where **dense regions** are?

natural weight of p:

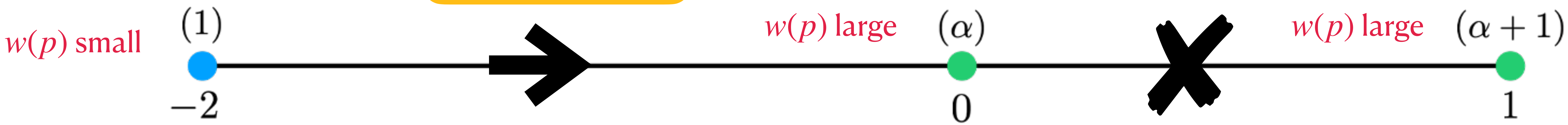
$w(p) :=$ max # pts whose total distance to p is $\leq 2B$

x, y are **β-well-separated** if far in weighted sense **:**

$$\min\{w(x), w(y)\} \cdot d(x, y) \geq \beta \cdot B$$

A1: if t well-separated points → t labels justified → use t labels

$k = 2, B = 2$



Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \rightarrow t$ parts?
➡ Greedy = assign to "closest" part

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \rightarrow t$ parts?
➡ Greedy = assign to "closest" part

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?
➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?
➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**
➡ assign a point to the cluster of its **closest** pivot **(do greedy)**

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \rightarrow t$ parts?
➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**
➡ assign a point to the cluster of its **closest** pivot **(do greedy)**
➡ pivot does not change while #labels in use stays the same

Attempt 2: How to use B?

Q1: When to increase # labels?

➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?

➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**

➡ assign a point to the cluster of its **closest** pivot **(do greedy)**

➡ pivot does not change while #labels in use stays the same

pivot $\neq$ center

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?
➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**
➡ assign a point to the cluster of its **closest** pivot **(do greedy)**
➡ pivot does not change while #labels in use stays the same
➡ **Invariant I:** Pivots are always well-separated

pivot $\neq$ center

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \rightarrow t$ parts?
➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**
➡ assign a point to the cluster of its **closest** pivot **(do greedy)**
➡ pivot does not change while #labels in use stays the same

pivot $\neq$ center

➡ **Invariant I:** Pivots are always well-separated
➡ **Invariant II:** No already arrived point is well-separated from existing pivots

Attempt 2: How to use B?

Q1: When to increase # labels?
➡ Wait for more evidence of where **dense regions** are?

A1: if t well-separated points → t labels justified → use t labels

Q2: Once we add label t, how to partition space from $t-1 \to t$ parts?
➡ Greedy = assign to "closest" part

A2: each cluster (= pts w/ same label) has a representative called a **pivot**
➡ assign a point to the cluster of its **closest** pivot **(do greedy)**
➡ pivot does not change while #labels in use stays the same

pivot $\neq$ center

➡ **Invariant I:** Pivots are always well-separated
➡ **Invariant II:** No already arrived point is well-separated from existing pivots

Pivots: Some Subtleties…

Pivots: Some Subtleties…

good representative for cluster (pivot)

Pivots: Some Subtleties…

good representative for cluster (pivot)

vs.

good center for cluster

Pivots: Some Subtleties…

recruit pts to right region

good representative for cluster (pivot)

vs.

good center for cluster

Pivots: Some Subtleties…

recruit pts to right region

good representative for cluster (pivot)

vs.

good center for cluster

low cost w.r.t obj. fn.

Pivots: Some Subtleties…

recruit pts to right region

good representative for cluster (pivot)

vs.

good center for cluster

low cost w.r.t obj. fn.

However: centers *do* come into play when we increase #pivots

Pivots: Some Subtleties…

recruit pts to right region

good representative for cluster (pivot)

vs.

good center for cluster

low cost w.r.t obj. fn.

However: centers *do* come into play when we increase #pivots

- - - - - = well-separated

Pivots: Some Subtleties…

recruit pts to right region

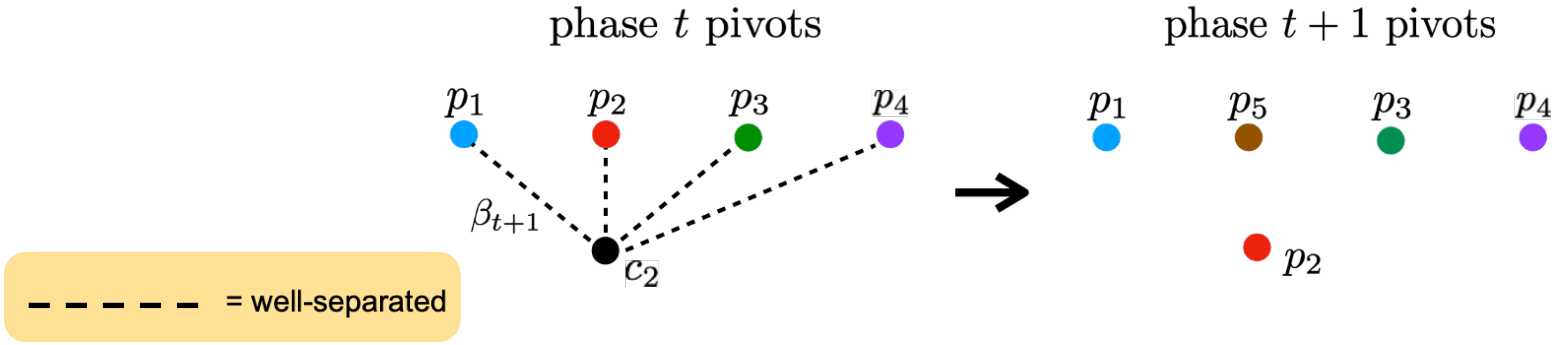
good representative for cluster (pivot)

vs.

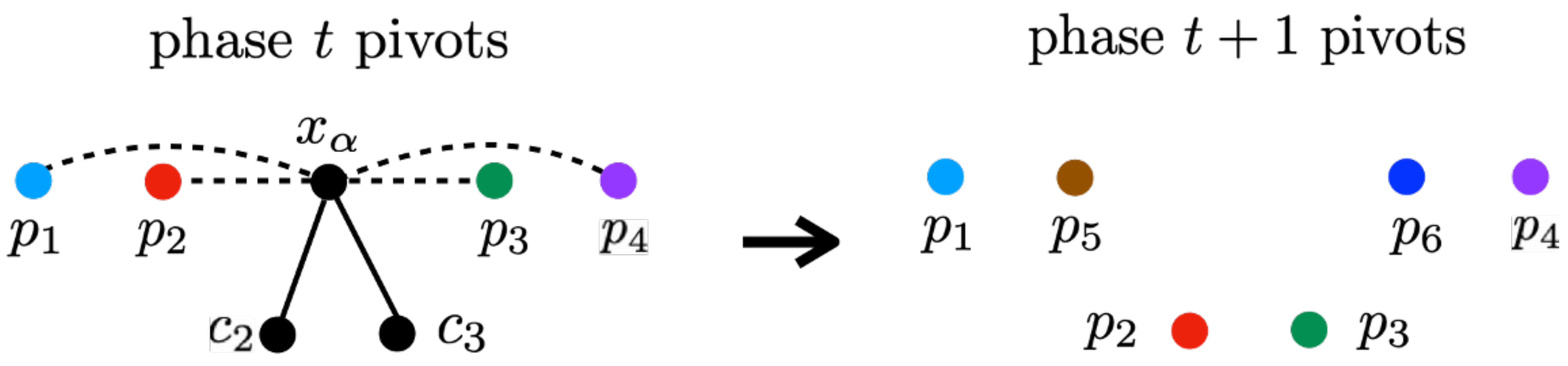
good center for cluster

low cost w.r.t obj. fn.

However: centers *do* come into play when we increase #pivots



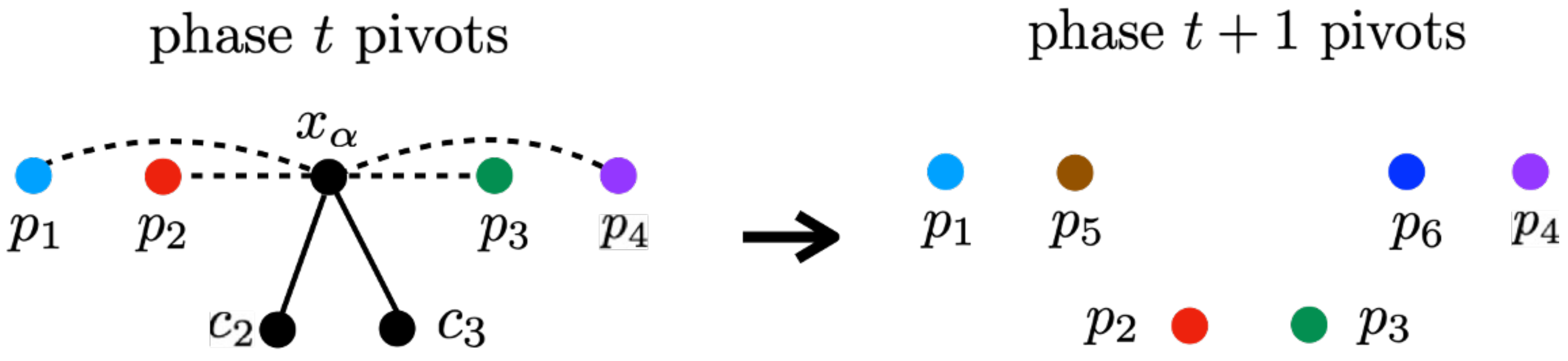
Pivots: Some Subtleties…



- - - - - = well-separated

———— = not well-separated

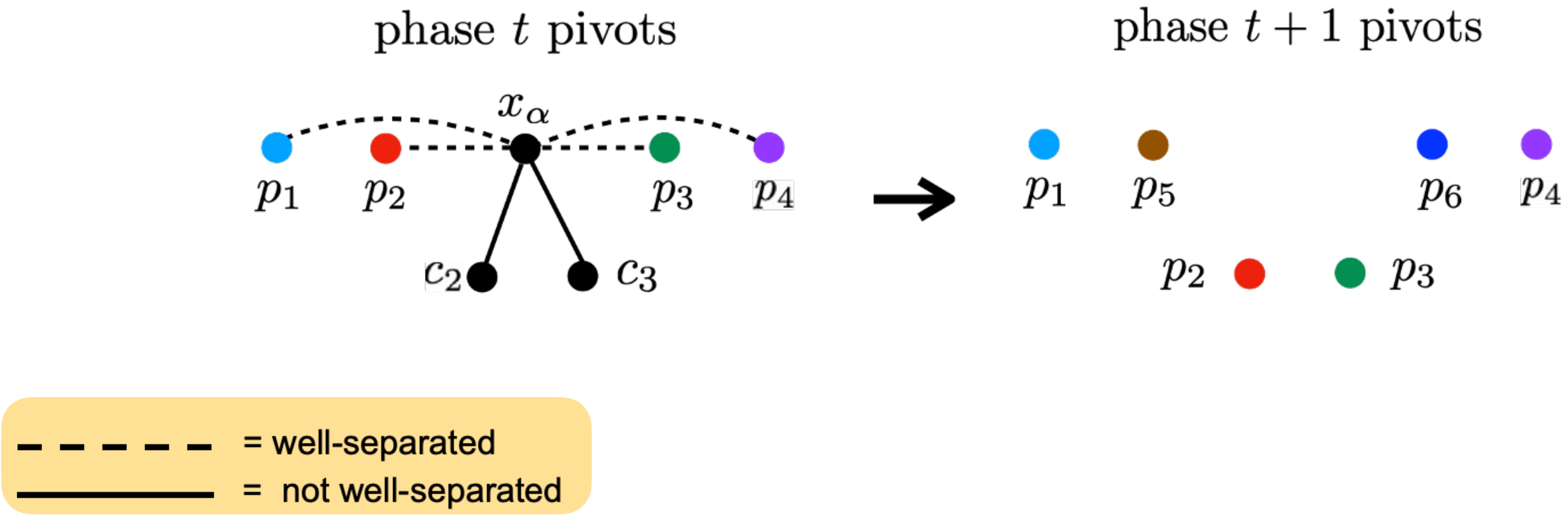
Pivots: Some Subtleties…



- - - - - = well-separated
——— = not well-separated

Upshot: need to handle delicately

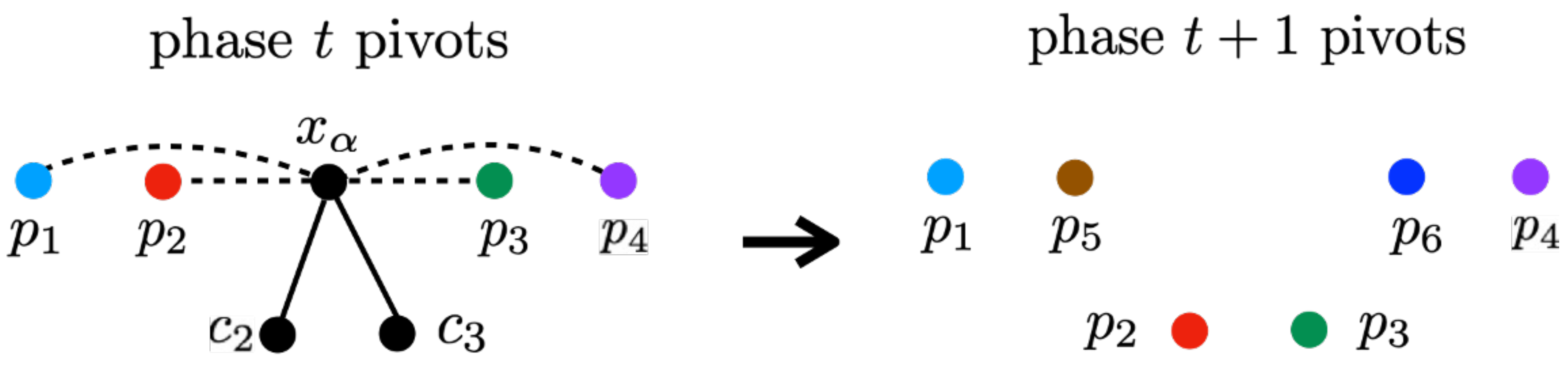
Pivots: Some Subtleties…

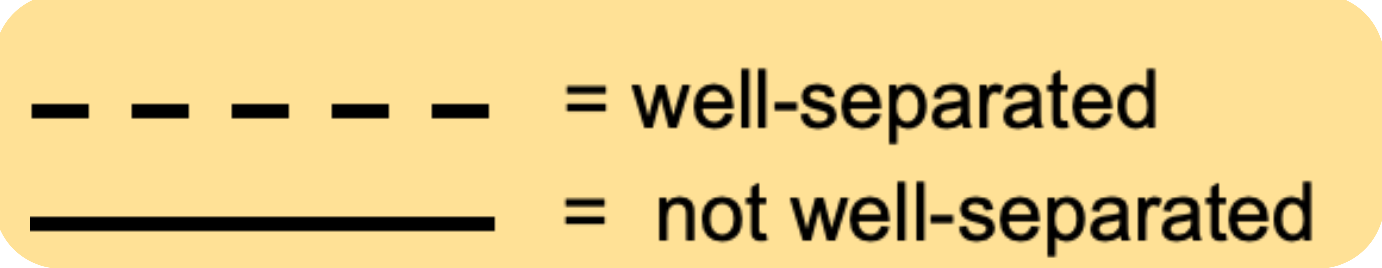


Upshot: need to handle delicately

1) which locations to add to set of pivots

Pivots: Some Subtleties…

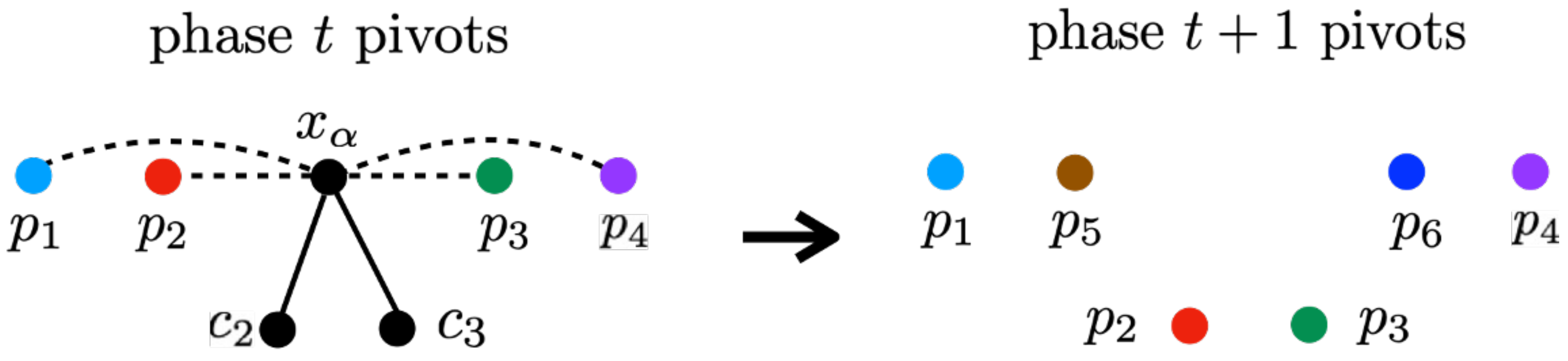


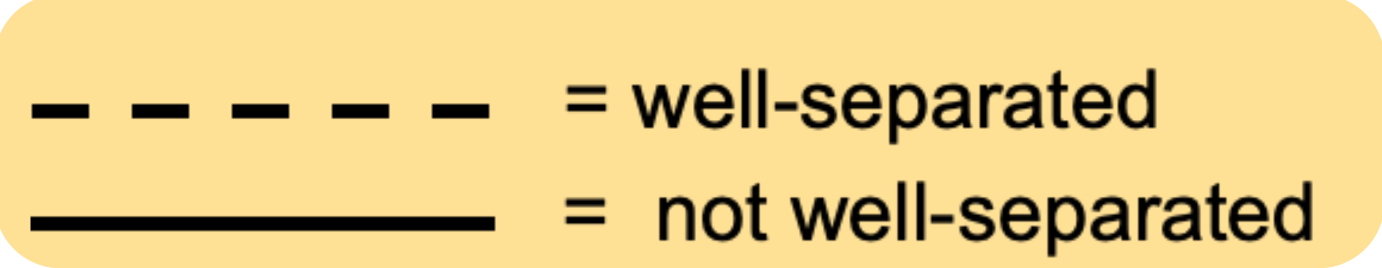


Upshot: <u>need to handle delicately</u>

1) which locations to add to set of pivots
2) which labels are given to which pivots

Pivots: Some Subtleties…





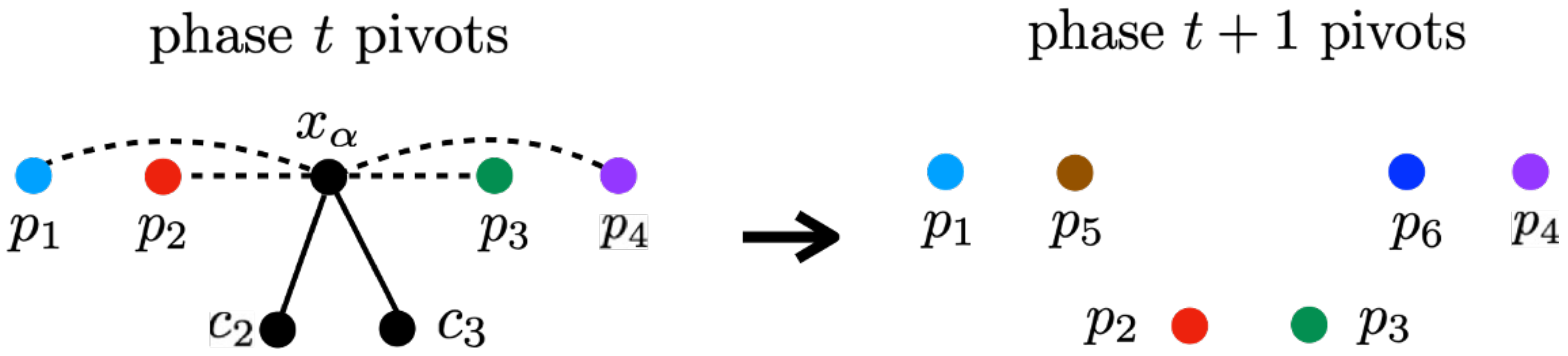
Upshot: need to handle delicately

1) which locations to add to set of pivots

2) which labels are given to which pivots

by incorporating information about centers

Pivots: Some Subtleties…



- - - - - = well-separated
——— = not well-separated

Upshot: need to handle delicately
1) which locations to add to set of pivots
2) which labels are given to which pivots
by incorporating information about centers

Conclusion

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of $\underline{k}$ labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers
- First cluster-based algorithm

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of $\underline{k}$ labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers

- First cluster-based algorithm

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers
- First cluster-based algorithm
- Not previously known whether such an algorithm could have bounded worst-case guarantees

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers
- First cluster-based algorithm
- Not previously known whether such an algorithm could have bounded worst-case guarantees

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers
- First cluster-based algorithm
- Not previously known whether such an algorithm could have bounded worst-case guarantees
- Open q: find optimal dependence on k

Conclusion

Main Result: There is a (deterministic, poly-time) online algo that, given budget B, irrevocably gives each point one of k labels on arrival, with cost $O(k^5 \cdot 3^k \cdot B)$.

- First algorithm with bounded competitive ratio that does not recluster or use more centers
- First cluster-based algorithm
- Not previously known whether such an algorithm could have bounded worst-case guarantees
- Open q: find optimal dependence on k

Thank You

hanewman@andrew.cmu.edu