Conception et Pratique de l'Algorithmique

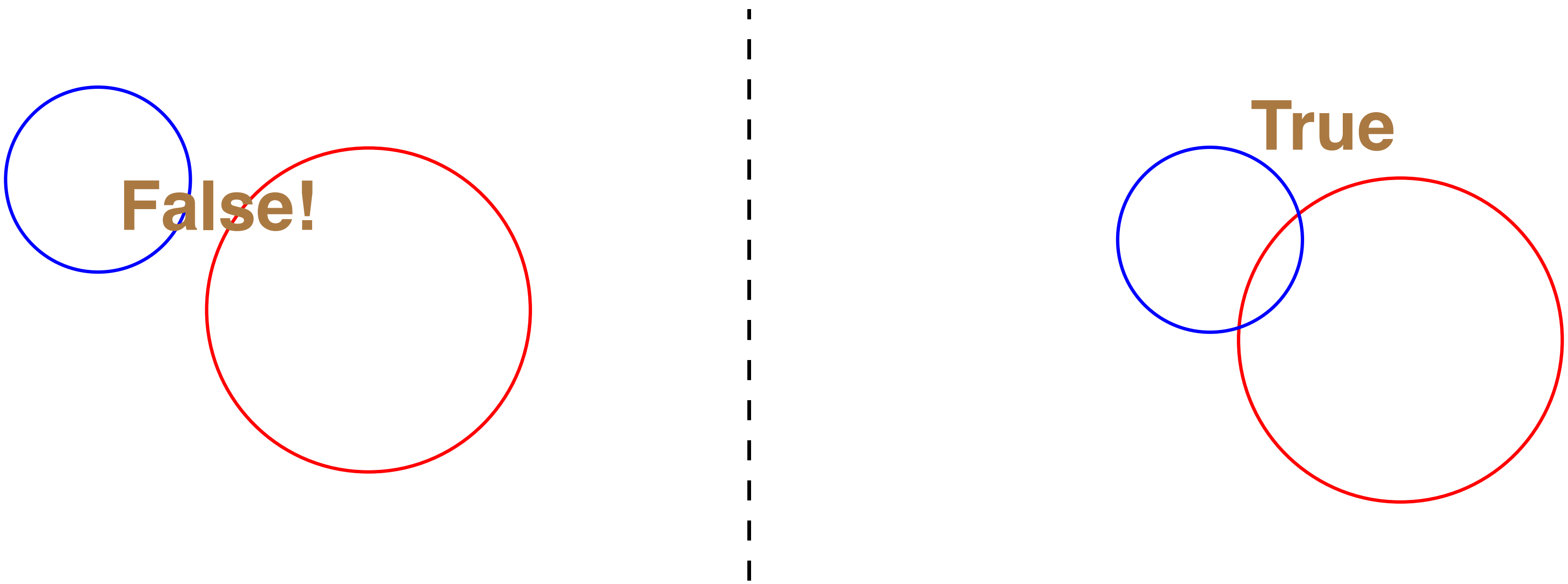
http://www-apr.lip6.fr/~buixuan/cpaad2020

Binh-Minh Bui-Xuan

CFA INSTA
Institut National Supérieur des Technologies Avancées

PARIS, Janvier 2021

Question : touché ?

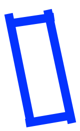


Collision d'objets géométriques

QUESTION : touché ?

Collision d'objets géométriques

QUESTION : touché ?



Exercice : collision entre cercles

EXERCICE : Soient deux cercles `c1` et `c2` de rayons `c1.radius` et `c2.radius`, dont les coordonnées des centres sont `(c1.x,c1.y)` et `(c2.x,c2.y)`. **Déterminer une condition nécessaire et suffisante pour que les deux cercles s'intersectent.**

SUPPORT :

`http://www-apr.lip6.fr/~buixuan/files/RBB_collision_canevas.html`

Exercice : collision entre cercles

Exercice : Soient deux cercles `c1` et `c2` de rayons `c1.radius` et `c2.radius`, dont les coordonnées des centres sont (`c1.x`,`c1.y`) et (`c2.x`,`c2.y`). Déterminer une condition nécessaire et suffisante pour que les deux cercles s'intersectent.

Support :
`http://www-apr.lip6.fr/~buixuan/files/RBB_collision_canevas.html`

Question : erreurs d'arrondi ?

NOM DU PROBLEME

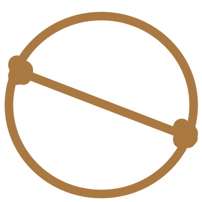
Problème du cercle minimum

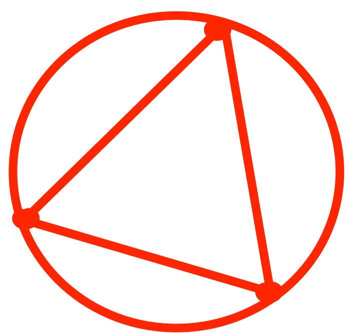
In : `Points`, une liste de coordonnées de points en 2D

Out : cercle couvrant tous les points de la liste, de rayon minimum

public Circle calculCercleMin(ArrayList<Point> points);

Point p = new Point(x,y); //x et y de type double
Circle c = new Circle(x,y,radius); //x et y et radius de ty
double





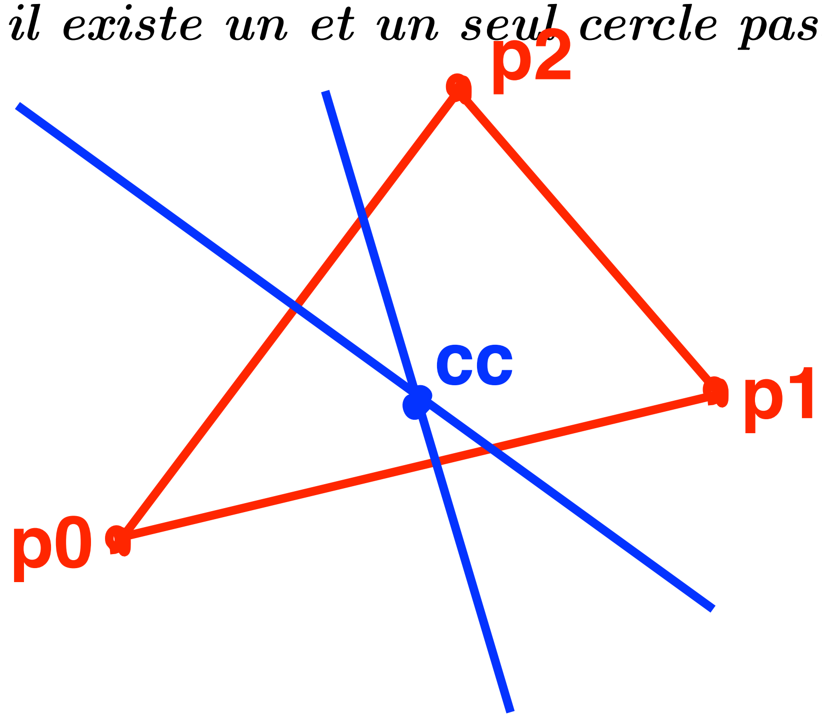
cc.x = cercleCirconscrit_x(p0,p1,p2)
cc.y = cercleCirconscrit_y(p0,p1,p2)

p0 = points.get(0); ....

Solution naïve

LEMME 1 : *si un cercle de diamètre égale à la distance de deux points de la liste couvre tout autre point de la liste, alors ce cercle est un cercle couvrant de rayon minimum.*

LEMME 2 : *en 2D, il existe un et un seul cercle passant par 3 points non-colinéaires.*



Solution naïve

LEMME 1 : *si un cercle de diamètre égale à la distance de deux points de la liste couvre tout autre point de la liste, alors ce cercle est un cercle couvrant de rayon minimum.*

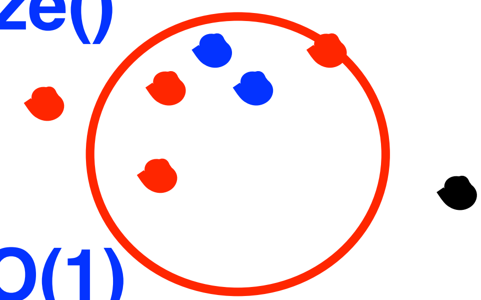
LEMME 2 : *en 2D, il existe un et un seul cercle passant par 3 points non-colinéaires.*

THÉORÈME : *le problème du cercle minimum peut être résolu en temps* $O(n^4)$.

QUESTION : algorithme prouvant ce théorème ?

Notes pour révision

n = points.size()



L'agorithme naïf en question :

O(n^3)

pour tout p dans `Points` O(n^2)

pour tout q dans `Points`

c $\leftarrow$ cercle de centre $\frac{p+q}{2}$ de diamètre $|pq|$ O(1)

si c couvre tous les points de `Points` alors retourner c O(n)

`resultat` $\leftarrow$ cercle de rayon infini rayon = MAX_DOUBLE O(1)

O(n^4)

pour tout p dans `Points` O(n^3)

pour tout q dans `Points`

pour tout r dans `Points`

c $\leftarrow$ cercle circonscrit de p, q et r O(1)

si c couvre `Points` et c plus petit que `resultat` O(n)

alors `resultat` $\leftarrow$ c

retourner ~~c~~ resultat O(1)

=> au total O(n^4) = O(n^3) + O(n^4) + O(1)

Exercice : estimation du temps

2^30 = 2^10*2^10*2^10 = 1024 * 1024 * 1024 ~ 10^3*10^3*10^3

QUESTION : un ordinateur de l'ordre du Giga-Hertz exécutant un algorithme en $O(n^4)$, avec $n = 10000$, quel est le temps d'exécution attendu (en ordre de grandeur) ?

– algorithme en $O(n^3)$?
– algorithme en $O(n^2)$?
– algorithme en $O(n)$?
– algorithme en $O(\log n)$?

Pour n = 10.000

pour O(n^3) --> ~h
pour O(n^2) --> ~sec

big

n	Kb	Mb	Gb	Tb
O(n)	~us	~ms	~sec	~h
O(nlog n)				
O(n^2)				
O(n^3)				
O(n^4)	~h	~1Ma		

Trade-off algorithmique

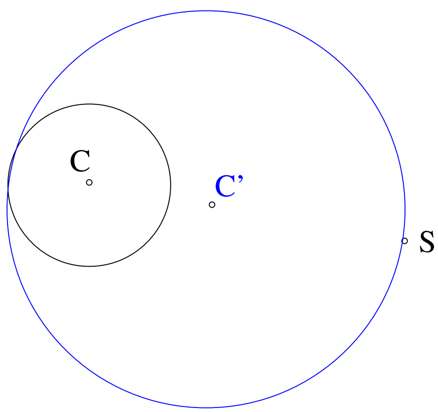
Qualité du résultat vs. temps d'exécution :

Qualité gagne	Temps gagne	Trade-off
imagerie médicale	audio-visuel	concours de prog.
systèmes critiques	appli. en temps réel	critère d'optimisation

Techniques d’approximation

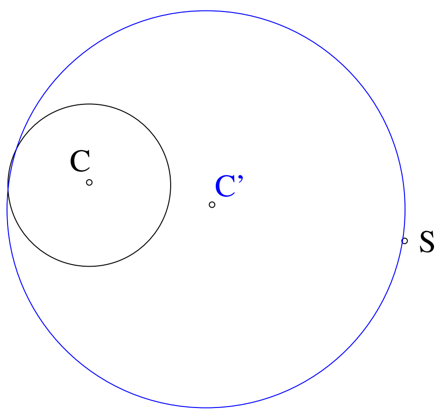
Algorithme incrémental

IDÉE : si un cercle ne couvre pas tous les points, on l'agrandit pour couvrir l'ancien cercle, plus au moins un nouveau point.



Algorithme incrémental

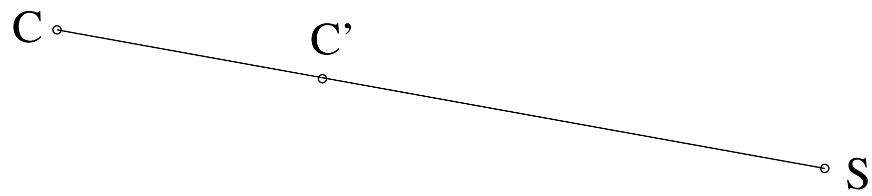
IDÉE : si un cercle ne couvre pas tous les points, on l'agrandit pour couvrir l'ancien cercle, plus au moins un nouveau point.



QUESTION : coordonnées de C' sachant C, S, ancien rayon r ?

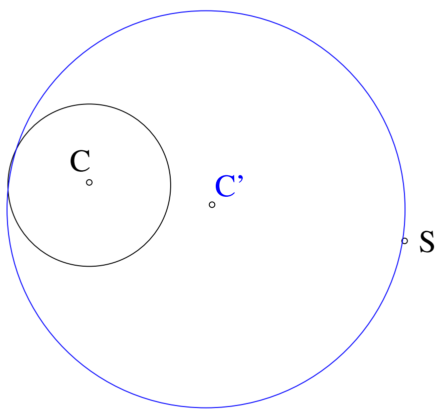
Coordonnées barycentriques

FORMULE : $C' = \alpha \cdot C + \beta \cdot S$, avec $\alpha = \frac{|C'S|}{|CS|}$ et $\beta = \frac{|C'C|}{|CS|}$.



Algorithme incrémental

IDÉE : si un cercle ne couvre pas tous les points, on l'agrandit pour couvrir l'ancien cercle, plus au moins un nouveau point.



QUESTION : coordonnées de C' sachant C, S, ancien rayon r ?

Peut on faire mieux ?

Vue générale de l'algorithmique

BRUTE-FORCE : parcours exhaustif de l'espace de recherche

EXEMPLE : pour toute coordonnée possible du centre du cercle, pour toute valeur possible du rayon du cercle, vérifier si tous les points sont couverts.

Vue générale de l'algorithmique

RECHERCHE ORDONNÉE : réorganisation de l'espace de recherche

EXEMPLE : pour toute coordonnée possible du centre *définie à partir de deux ou de trois points de la liste, soit* $\times\times\times$ *la seule valeur intéressante du rayon*, vérifier si tous les points sont couverts.

EXEMPLE : algorithme naïf pour cercle couvrant.

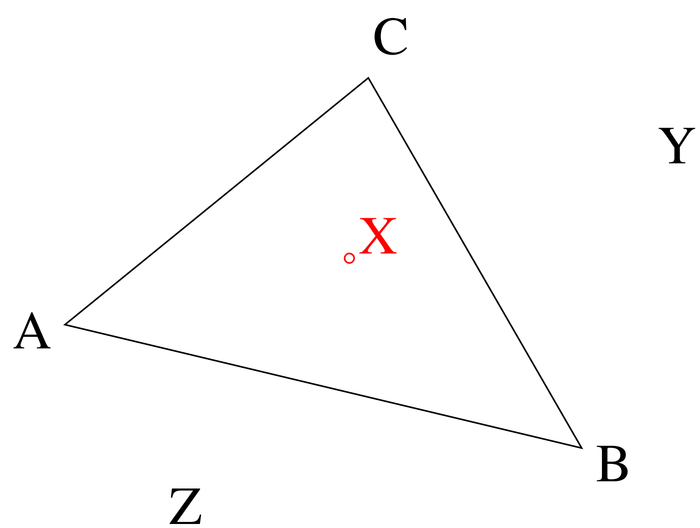
Vue générale de l'algorithmique

RECHERCHE PARCIMONIEUSE : réorganisation de l'espace de recherche + localisation

IDÉE : filtrer l'espace de recherche par un précalcul.

Borner la recherche par un précalcul

IDÉE : filtrer le "INPUT" pour écarter les zones de recherche inutiles



QUESTION : X est inutile, comment le détecter, numériquement ?

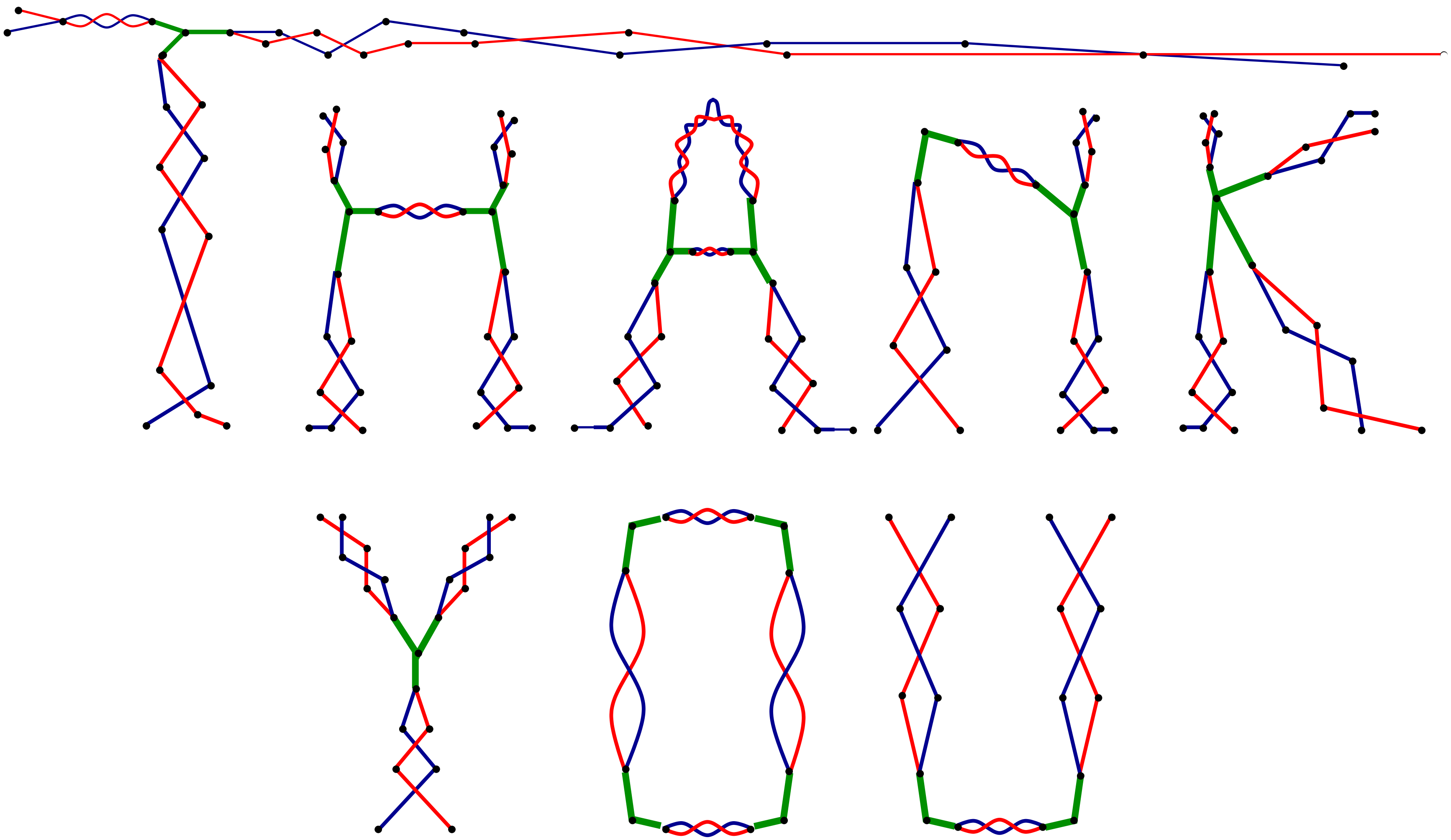
Conclusion, question

CONCLUSION :

- algorithme naïf
- technique incrémental
- technique de filtrage (précalcul)

QUESTION :

- meilleur précalcul ? (voir TME pour une réponse)


THANK
YOU