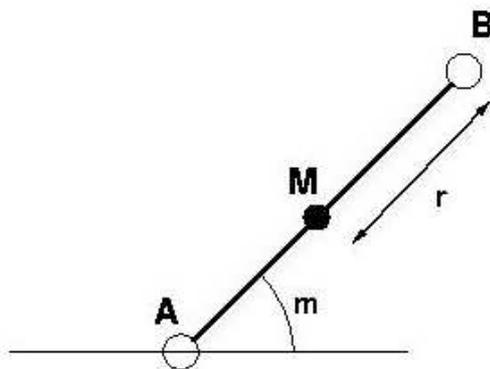


# Capitolo 1

## Strutture dati

Di ogni arco AB aggiunto o da aggiungere si considerano le seguenti caratteristiche:



$\{N_{from}, X_a, Y_a, X_M, Y_M, N_{to}, X_b, Y_b, r, m\}$

- $N_{from}$ : Numero del nodo di partenza;
- $X_a$ : Coordinata X del nodo di partenza;
- $Y_a$ : Coordinata Y del nodo di partenza;
- $X_M$ : Coordinata X del punto medio;
- $Y_M$ : Coordinata Y del punto medio;
- $N_{to}$ : Numero del nodo di arrivo;
- $X_b$ : Coordinata X del nodo di arrivo;
- $Y_b$ : Coordinata Y del nodo di arrivo;
- $r$ : raggio  $\frac{d_{AB}}{2}$ ;
- $m$ : coefficiente angolare;

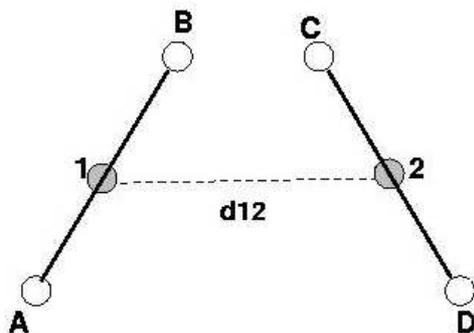
Inoltre i lati all'interno del file vanno (o lo devono essere già) ordinati per lunghezza crescente.

## Capitolo 2

# Algoritmo

Per quanto riguarda l'algoritmo questo si basa fondamentalmente su due livelli di check basati su considerazioni sia geometriche che informatiche. Il primo di semplice realizzazione consente di effettuare una grossa scrematura per selezionare solo i casi 'a rischio' in modo da non appesantire la complessità computazionale. Il secondo invece molto più complesso e raffinato permette di individuare con certezza i lati da aggiungere alla GT o da scartare mediante una serie di controlli più sofisticati e numerosi del primo.

Considerati quindi due lati AB (appartenente alla GT) e CD (da testare), e detti  $m_1$ ,  $m_2$  i coefficienti angolari,  $r_1$ ,  $r_2$  i raggi dei due segmenti ed indicando con 1 e 2 i punti medii di questi si ha che:



### 2.1 Step 1

Se  $d_{12} > (r_1 + r_2)$  Allora sicuramente CD non interseca AB;

Se  $d_{12} \leq (r_1 + r_2)$  Bisogna passare allo step 2;

### 2.2 Step 2

Supponendo che i vertici di un lato siano ordinati in modo che:

$$X_a < X_b$$

$$X_c < X_d$$

Costruiamo il sistema lineare associato alle rette individuate dai due segmenti

$$\begin{cases} y - y_1 = m_1(X - x_1) & X \in [X_a, X_b] \\ y - y_2 = m_2(X - x_2) & X \in [X_c, X_d] \\ y_1 + m_1(X - x_1) = y_2 + m_2(X - x_2) \end{cases}$$

$$y_1 - y_2 - m_1x_1 + m_2x_2 = (m_2 - m_1)X$$

$$\mathbf{X} = \frac{y_1 - m_1x_1 - (y_2 - m_2x_2)}{m_2 - m_1}$$

Che costituisce la proiezione sull'asse X dell'intersezione tra le due rette. Analogamente si può ottenere:

$$\mathbf{Y} = \frac{m_2y_1 - m_1y_2 + m_1m_2(x_2 - x_1)}{m_2 - m_1}$$

Che costituisce la proiezione sull'asse Y dell'intersezione tra le due rette. Noti questi due parametri vediamo di elencare i check utilizzati per scoprire se un lato appartiene alla GT oppure no.

### 2.2.1 Intersezione X

Se  $X \in ([X_a, X_b] \wedge [X_c, X_d])$ , allora i due lati si intersecano sicuramente ed il lato può essere scartato.

### 2.2.2 Estremi coincidenti

Il check descritto sopra è molto potente ma va corretto perchè se uno dei due estremi coincide con quello dell'altro lato il lato viene scartato nonostante esso appartenga alla GT. Per questo motivo bisogna tenere conto in sede di implementazione dei lati che hanno tali caratteristiche.

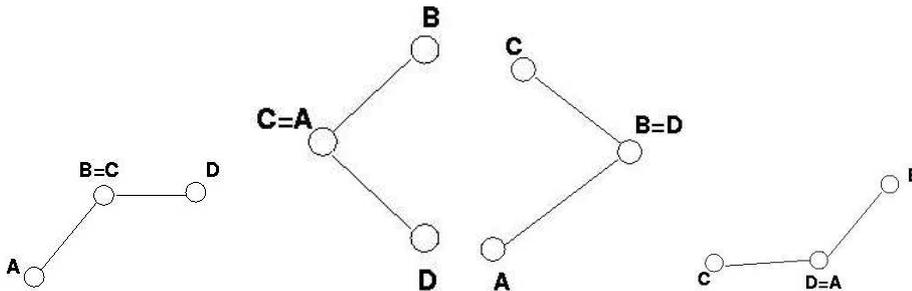


Figura 2.1: Lati con estremi coincidenti

### Lati paralleli

Inoltre bisogna tenere conto dei 'lati paralleli' che si sovrappongono a quelli già presenti inserendo un ulteriore check in sede di implementazione.

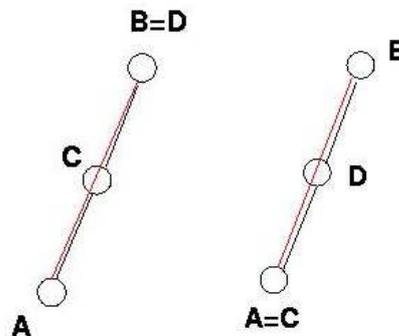


Figura 2.2: Lati che si sovrappongono a loro stessi. I lati in rosso sono quelli che si sovrappongono

### 2.2.3 Intersezioni con lati paralleli ad un asse

Purtroppo il check basato solo sul parametro X (o eventualmente su Y) non é sufficiente ad eliminare tutti i lati estranei alla GT. In particolar modo si é osservato che in presenza di uno o piú lati paralleli ad un asse l'algorithm mostra dei bug che possono essere eliminati aggiungendo un ulteriore controllo del tipo visto su X ma basato sull'altro parametro cioé Y. Questo sistem infatti permette di verificare particolari configurazioni che si possono incontrare.

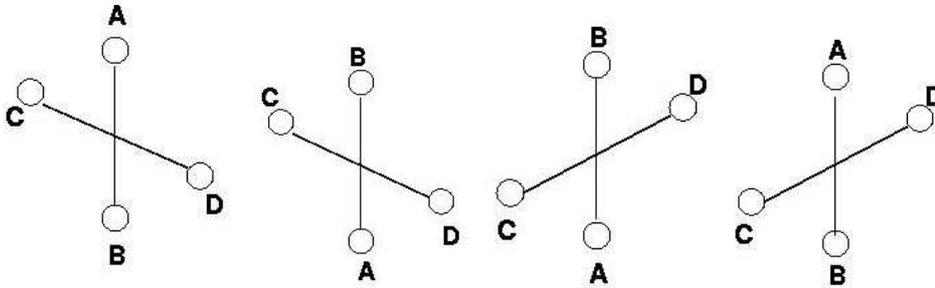


Figura 2.3: Lati paralleli all'asse Y

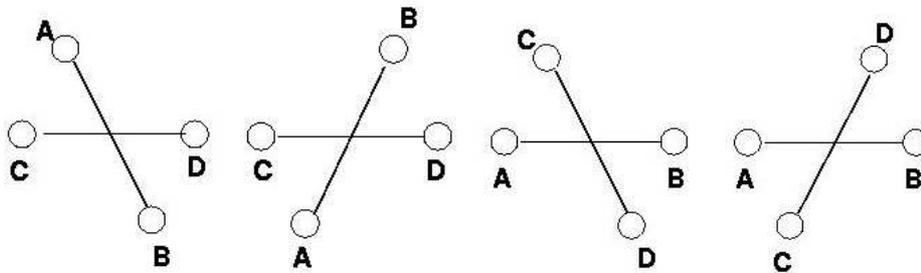


Figura 2.4: Lati paralleli all'asse X

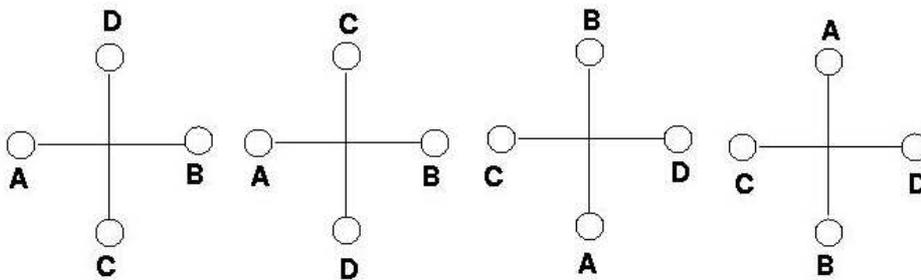


Figura 2.5: Lati perpendicolari