Università
degli Studi
della Campania
Luigi Vanvitelli

DIPARTIMENTO DI
INGEGNERIA
INDUSTRIALE E
DELL'INFORMAZIONE

IDENTICAZIONE SISTEMA DI CARICHE PUNTIFORMI DA MISURE DI POTENZIALE

CORSO DI METODI DI OTTIMIZZAZIONE

A.A. 2017/18

Luigi Previdente
Giuseppe Valletta A18000263
Mario Baldi A18000260

Ch.mo Prof.
Raffaele Martone

INDICE

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

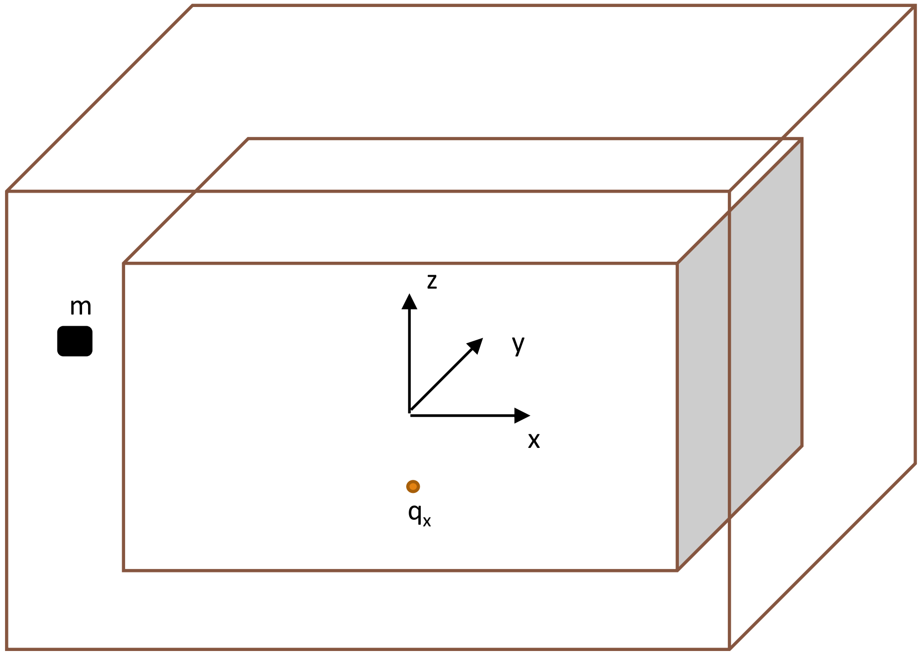
- **1. Introduzione al problema**
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

Descrizione del problema di identificazione

Sistema:

- #N cariche:
  - **#N-1 note** (posizione, carica)
  - **#1 non nota**

Le N cariche si trovano in un brick interno, su uno esterno effettuiamo le M misurazioni del potenziale totale.



- **1. Introduzione al problema**
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

APPROCCIO DEL PROBLEMA

- Il **potenziale totale** V_t è noto dai misurati
- Il **potenziale delle N-1 cariche** note V_n è calcolato analiticamente
- Il **potenziale della carica ignota** V_N è calcolato grazie al principio di sovrapposizione per differenza tra V_t e V_n
- I misuratori si trovano all’esterno del *brick* contente le cariche per evitare singolarità
- I misuratori rilevano il potenziale totale delle N cariche
- Qual è il numero minimo di misuratori necessari?

- 1. Introduzione al problema
- **2 Modello matematico**
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

MODELLO MATEMATICO

Potenziale generica carica

$$V = \frac{1}{4\pi\varepsilon}\frac{q}{r}$$

Il potenziale di N cariche visto da un misuratore

$$V_t = \frac{1}{4\pi\varepsilon}\sum_{i=1}^{N}\frac{q_i}{||\vec{\boldsymbol{p}_i}-\vec{\boldsymbol{m}}||}$$

- 1. Introduzione al problema
- 2 Modello matematico
  - **2.1 Normalizzazioni**
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

NORMALIZZAZIONI

Possiamo effettuare 2 tipi di normalizzazioni:

- Possiamo normalizzare la costante moltiplicativa

$$\frac{1}{4\pi\varepsilon}$$

- Normalizziamo sul numero **M** di misurazioni effettuate

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - **2.2 Funzione obbiettivo**
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

FUNZIONE OBIETTIVO

Ricavando V_N dalla differenza V_t e V_n

$$4\pi\varepsilon V_N = 4\pi\varepsilon\,(V_t - V_n) = \frac{q_N}{\left\|\overrightarrow{\boldsymbol{p_N}} - \overrightarrow{\boldsymbol{m}}\right\|}$$

Calcoliamo la differenza di potenziale tra V_N e quello di una generica carica **q***.

$$4\pi\varepsilon\Delta V = 4\pi\varepsilon\,(V_N - V^*) = V_N - \frac{q^*}{\left\|\overrightarrow{\boldsymbol{p}^*} - \overrightarrow{\boldsymbol{m}}\right\|}$$

La funzione obiettivo è somma quadratica delle differenze di potenziale viste da ogni misuratore

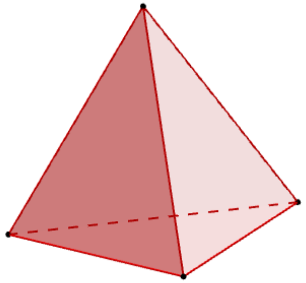
$$\frac{E\left(q^*,\overrightarrow{\boldsymbol{p}^*}\right)}{M} = \sum_{j=1}^{M} 4\pi\varepsilon\Delta V\Big|_{m=m_j} = \sum_{j=1}^{M}\left(V_{Nj} - \frac{q^*}{\left\|\overrightarrow{\boldsymbol{p}^*} - \overrightarrow{\boldsymbol{m}_j}\right\|}\right)^2$$

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- **3 Algoritmo ed implementazione**
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

SIMPLESSO

L'algoritmo del simplesso è un metodo numerico per risolvere problemi di programmazione lineare.

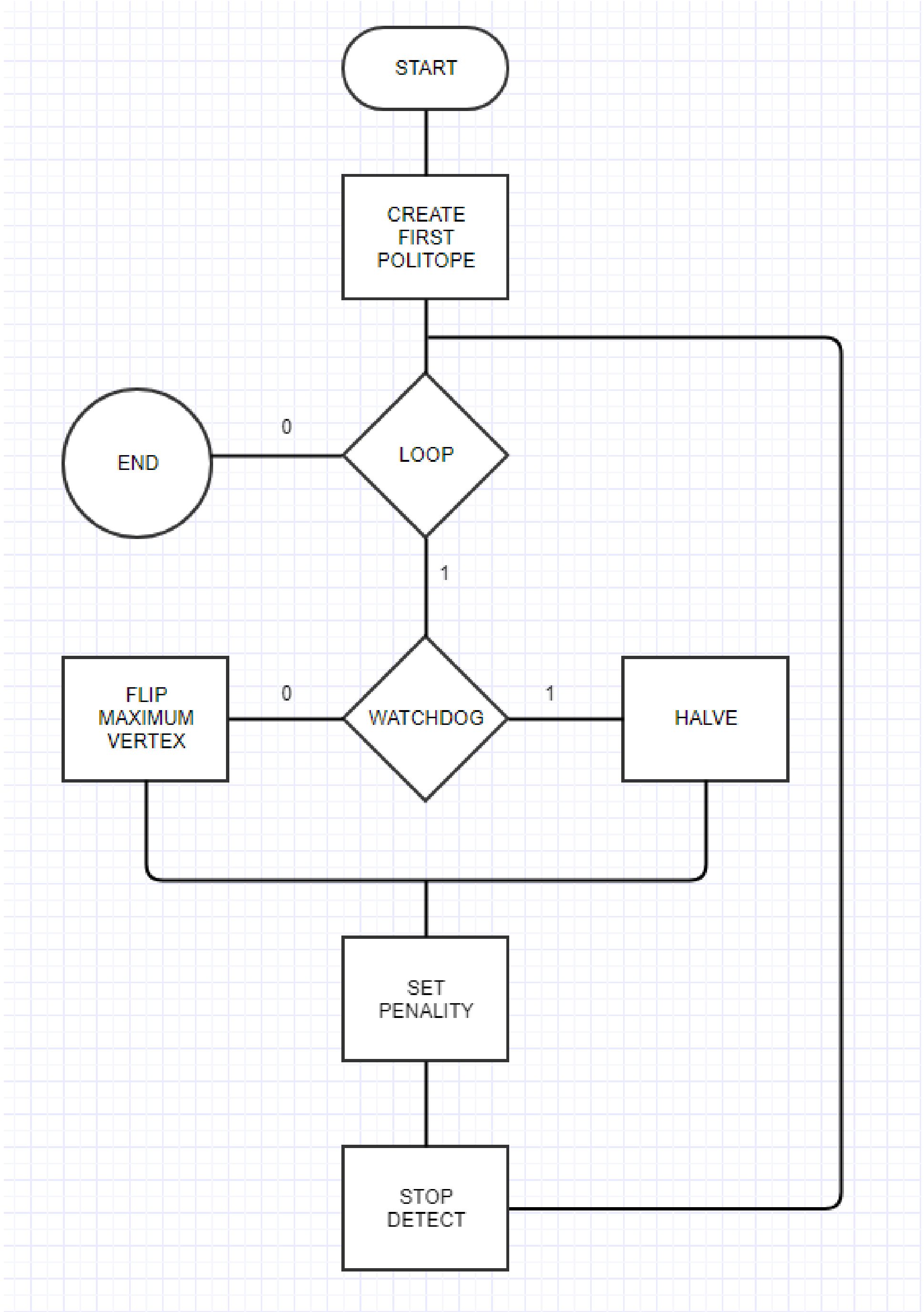
Il **Politopo** è figura geometrica *n*-dimensionale col numero di vertici pari ad n+1. In uno spazio **a tre dimensioni** è un **tetraedro**.



Il vertice con valore più grande viene **ribaltato** creando un nuovo politopo. In caso di ribaltamenti ripetuti attorno ad un minimo si effettua *l'operazione di **contrazione*** dove viene conservato il vertice migliore (minimo).

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - **3.1 Flow-Chart**
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

FLOW-CHART



- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- **3 Algoritmo ed implementazione**
  - **3.1 Flow-Chart**
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

IMPLEMENTAZIONE

Analizziamo più nel dettaglio i metodi necessari:

get_first_polytope: genera il primo politopo partendo dal punto iniziale

halve: effettua la contrazione del tetraedro

find_maximum: ritorna il vertice massimo

find_minimum: ritorna il vertice minimo

flip: ribalta il vertice Massimo (find_maximum) dando luogo ad un nuovo politopo

watchdog: controlla se avviene un ribaltamento ripetuto (lista ultimi vertici)

set_penality: assegna ai vertici una penalità nel caso in cui non soddisfano I vincoli

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - **3.2 Parametri iniziali**
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

CONSIDERAZIONI (PARAMETRI INIZIALI)

- Superficie iniziale del politopo
  - Diminuzione:
    - Aumentano le iterazioni necessarie ad avvicinarsi al minimo
  - Aumento:
    - Veloce se il politopo parte lontano dal minimo, i dimezzamenti sul pivot (vertice minimo) permettono spostamento veloci iniziali

- Posizione iniziale
  - L’algoritmo converge più lentamente al minimo se è lontano da questo

- Condizione di arresto sul lato del politopo
  - Condiziona la precisione dell’approssimazione
  - Se troppo basso causa un numero eccessivo di iterazioni, inutili se non si ha bisogno di approssimazioni raffinate

- Condizione di arresto su numero di iterazioni
  - Necessaria in caso di comportamenti anomali dell’algoritmo

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - **3.3 Condizioni di arresto**
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

CONDIZIONI DI ARRESTO

1. Lato minimo del politopo raggiunto dopo un certo numero di iterazioni

2. Numero di iterazioni effettuate

- 1. Introduzione al problema
- 2 Modello del problema
  - 2.1 Formule matematiche
  - 2.2 Funzione di costo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - **3.4 Problematiche**
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

PROBLEMI

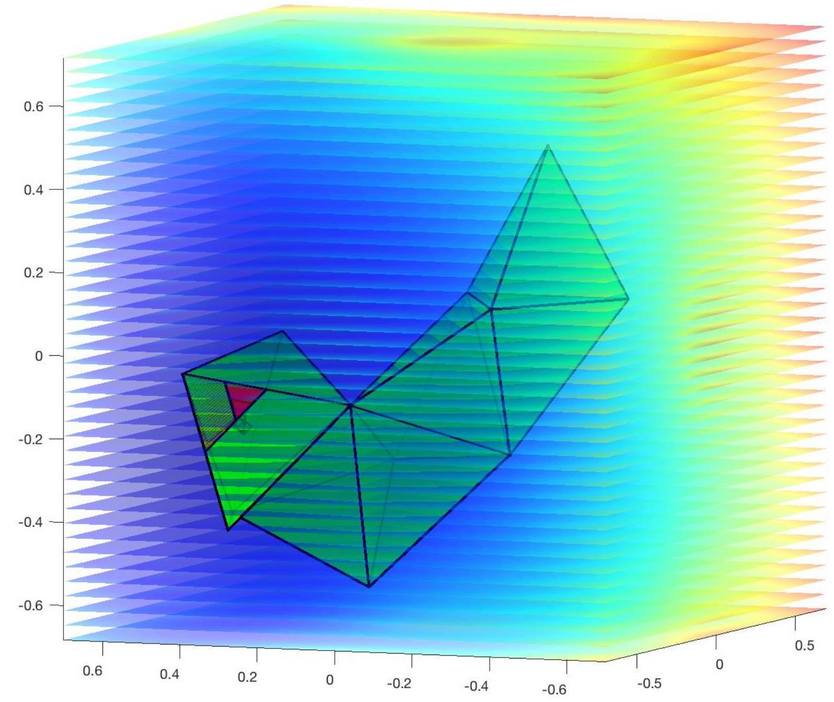
- Il politopo non rispetta il vincolo imposto uscendo dalla sua frontiera.

- Necessità di disegnare un grafico quadridimensionale

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - **3.4 Problematiche**
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

SOLUZIONI

- Implementazione del metodo delle penalità
- Suddivisione dello spazio in piani colorati da linee isolivello



- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- **4 Risultati**
  - **Incognite (x, y, q)**
    - **4.1 Risultato non vincolato**, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

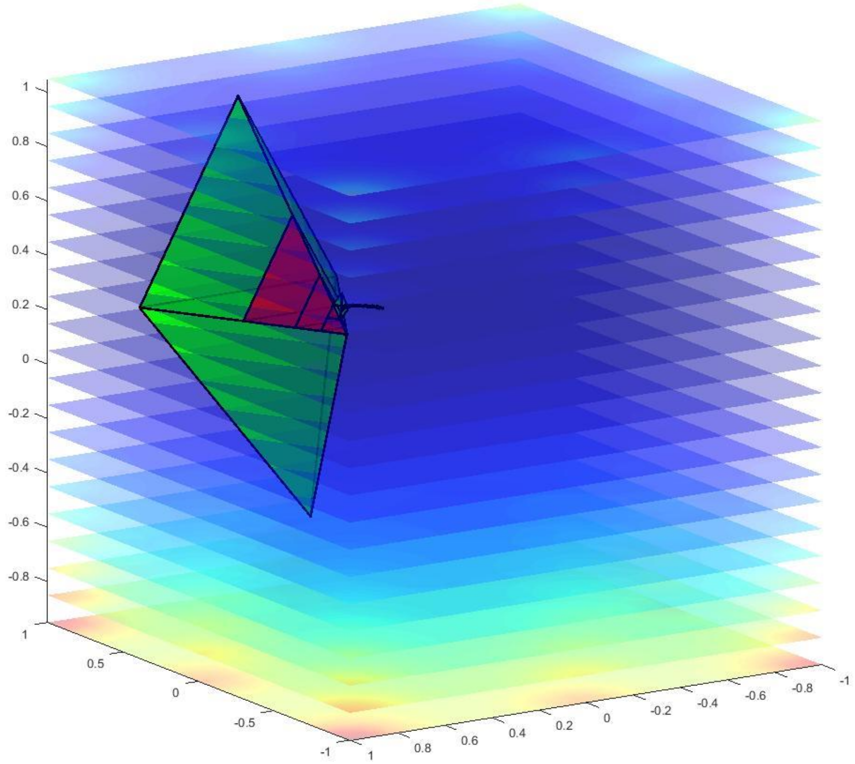
RISULTATO SENZA VINCOLO (inc. x, y, q)

```
Simplex(@cost_function, { }, [-.2 .5 .3], .3, 1e-5, 500);
```

Risultati

- Valore nell’ultimo vertice = 2.9227e-04
- risultato = x: -0.2799 y: 0.3995 z: 0.4000
- iterazioni = 500
- dimezzamenti = 13
- lato finale = 7.7591e-04
- Errore (x,y,q) = 1.6150e-06*(0.0201, 0.0005, 0.0000)

Minimo: (-0.3, 0.4, 0.4)



- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- **4 Risultati**
  - **Incognite (x, y, q)**
    - 4.1 Risultato non vincolato, **vincolato** e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

RISULTATO CON VINCOLO (inc. x, y, q)

```
Simplex(@cost_function, {@bound}, [-.2 .5 .3], .3, 1e-5, 500);
```

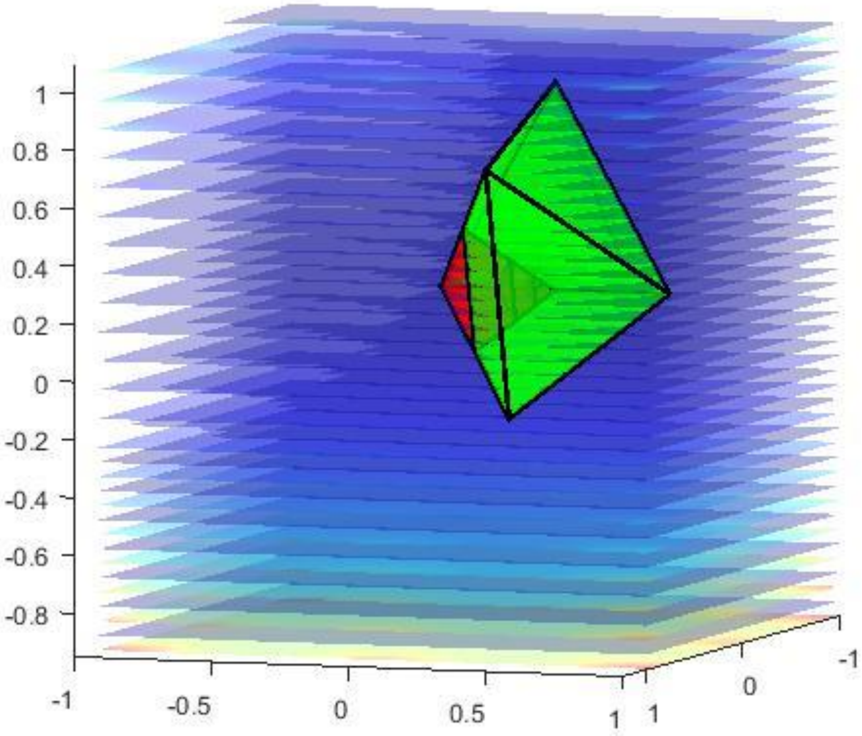
Vincolo:

$$(x + 0.6)^2 + (y - 0.6)^2 + (z - 0.5)^2 - 1.2^2 < 0$$

Risultati

- Valore nell’ultimo vertice = 2.9227e-04
- risultato =   x: -0.1386   y: 0.1517 q: 0.3993
- iterazioni = 346
- dimezzamenti = 20
- lato finale = 5.6550e-06
- errore = x: 0.1614   y: 0.2483 z: 0.0007

Minimo: (-0.3, 0.4, 0.4)



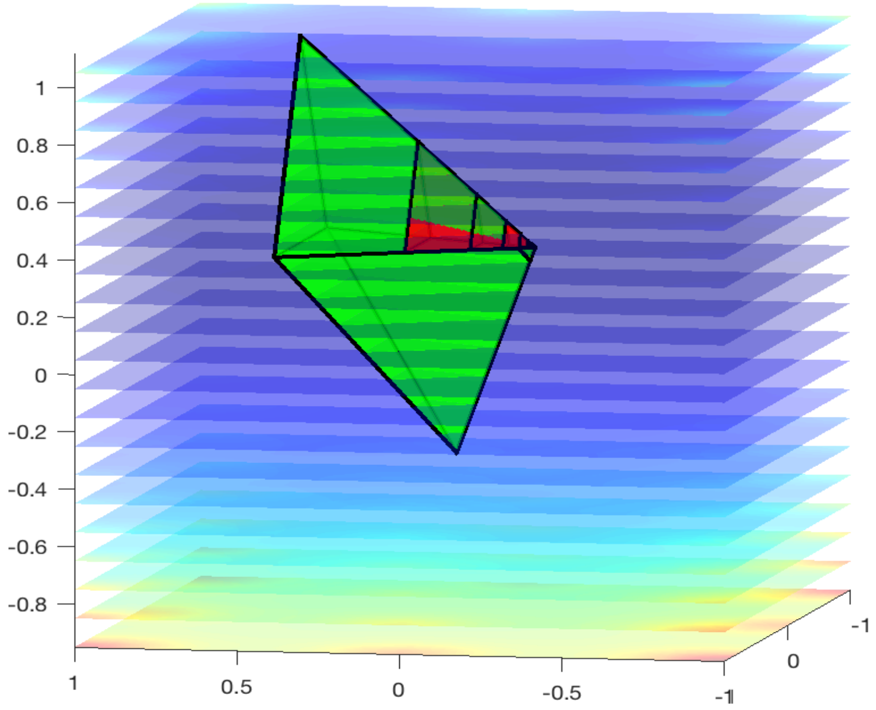
- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- **4 Risultati**
  - **Incognite (x, y, q)**
    - 4.1 Risultato non vincolato, vincolato e **crimine inverso**
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- 5 Considerazioni

RISULTATO CRIMINE INVERSO (inc. x, y, q)

```
Simplex(@cost_function, {}, [-.3 .4 .4], .3, 1e-5, 500);
```

- Valore nell’ultimo vertice = 0.000
- Risultato = x: -0.3000 y: 0.4000 q: 0.4000
- iterazioni = 35
- dimezzamenti = 20
- lato finale = 6.7356e-06
- errore = x: 0.0000 y: 0.0000 q: 0.000

Minimo: (-0.3, 0.4, 0.4)



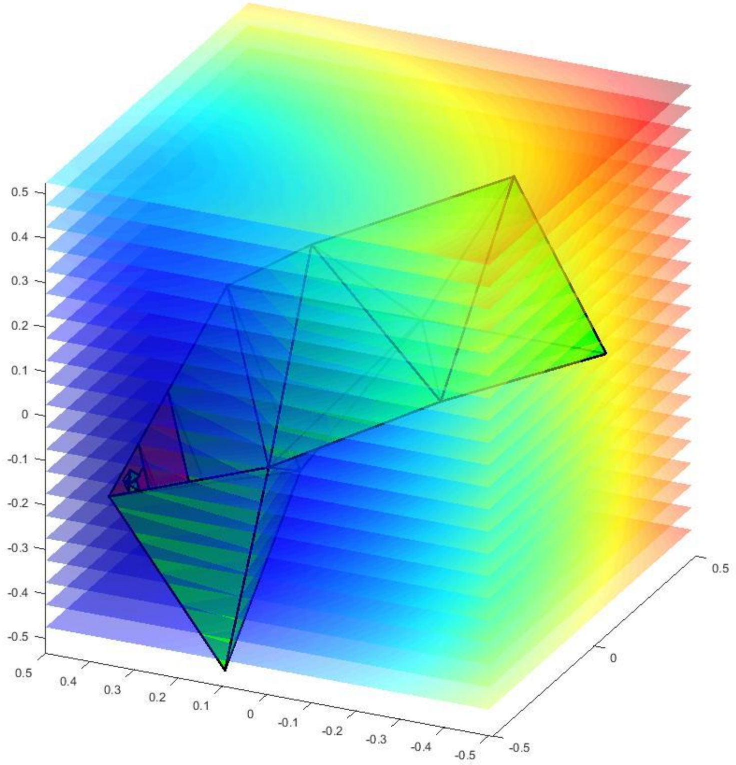
- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- **4 Risultati**
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - **Incognite (x, y, z)**
    - **4.2 Risultato non vincolato**, vincolato e crimine inverso
- 5 Considerazioni

RISULTATO SENZA VINCOLO (inc. x, y, z)

```
Simplex(@cost_function, { }, [-.3 -.3 .1], .3, 1e-15, 250);
```

Risultati

- Valore nell’ultimo vertice = 5.9165e-31
- risultato = x: -0.3000 y: 0.4000 z: -0.2000
- iterazioni = 250
- dimezzamenti = 167
- lato finale = 2.1499e-16
- Errore (x, y, z) =
  1.0e-15 * (0.3331 0.0000 0.4163)



Minimo: (-0.3, 0.4, -0.2)

- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- **4 Risultati**
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - **Incognite (x, y, z)**
    - 4.2 Risultato non vincolato, **vincolato** e crimine inverso
- 5 Considerazioni

RISULTATO CON VINCOLO (inc. x, y, z)

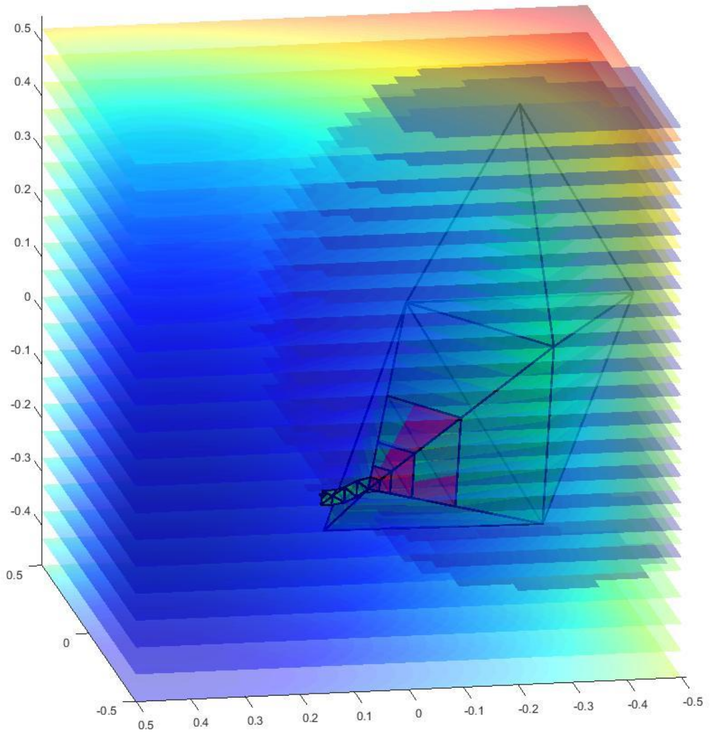
```
Simplex(@cost_function, {@bound}, [-.3 -.3 .1], .3, 1e-15,
250);
```

Vincolo:

$$(x+0.3)^2+(y+0.3)^2+(z-0.1)^2-0.5^2<0$$

- Valore nell'ultimo vertice = 0.0762
- risultato = x: -0.3122 y: 0.1306 z: -0.1539
- iterazioni = 250
- dimezzamenti = 72
- lato finale = 4.1499e-11
- errore = x: 0.0122 y: 0.2694 z: 0.0461

Minimo: (-0.3, 0.4, -0.2)



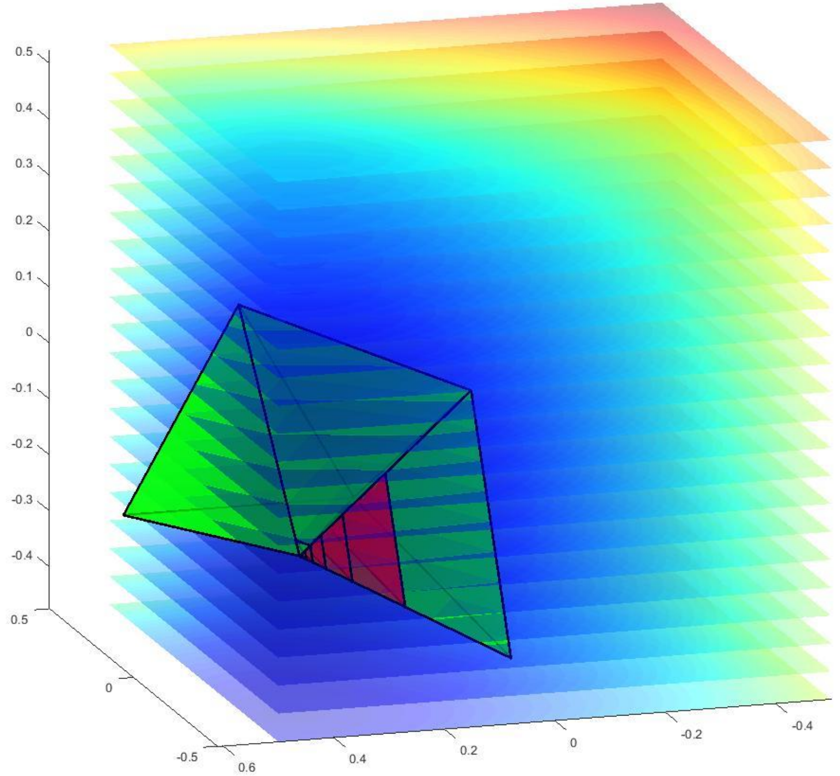
- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- **4 Risultati**
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - **Incognite (x, y, z)**
    - 4.2 Risultato non vincolato, vincolato e **crimine inverso**
- 5 Considerazioni

RISULTATO CRIMINE INVERSO (inc. x, y, z)

```
Simplex(@cost_function, { }, [-.3 -.3 .1], .3, 1e-15, 250);
```

- Valore nell’ultimo vertice = 0.0000
- risultato = x: -0.3000 y: 0.4000 z: -0.2000
- iterazioni = 250
- dimezzamenti = 215
- lato finale = 1.9230e-16
- errore = x: 0.0000 y: 0.0000 z: 0.000

Minimo: (-0.3, 0.4, -0.2)



- 1. Introduzione al problema
- 2 Modello matematico
  - 2.1 Normalizzazioni
  - 2.2 Funzione obbiettivo
- 3 Algoritmo ed implementazione
  - 3.1 Flow-Chart
  - 3.2 Parametri iniziali
  - 3.3 Condizioni di arresto
  - 3.4 Problematiche
- 4 Risultati
  - Incognite (x, y, q)
    - 4.1 Risultato non vincolato, vincolato e crimine inverso
  - Incognite (x, y, z)
    - 4.2 Risultato non vincolato, vincolato e crimine inverso
- **5 Considerazioni**

CONSIDERAZIONI

È possibile notare dai due esempi come cambi la rappresentazione della funzione di costo e quanto incida dipendenza delle incognite sul percorso del politopo e la **velocità di convergenza** dell’algoritmo.

Nel problema con le coordinate **x, y** e la carica **q** incognite, l’algoritmo tende a minimizzare prima **q** per la sua **dipendenza lineare** e poi a muoversi su un piano per minimizzare i parametri rimanenti.

Nel secondo caso, le incognite **x, y** e **z** hanno tutte una dipendenza **iperbolica** e l’algoritmo le minimizza contemporaneamente

Il **numero minimo di misuratori** deve essere pari al **numero di incognite** del nostro problema per far si che il **sistema** sia **determinato**.

Nell’eventualità che le misurazioni siano affette da **rumore** ne riduciamo il disturbo calcolando la media delle misurazioni.