Laboratorio di Informatica T (Ch12)

Esercizio: Equilibrio Termodinamico

Esercizio: Equilibrio Termodinamico

Consideriamo una reazione chimica nella forma:

$$\nu_1 C_1 + \nu_2 C_2 + \ldots \longrightarrow \ldots \nu_{n-1} C_3 + \nu_n C_4$$

- Ogni C_i rappresenta una sostanza (reagente o prodotto)
- ν_i è il corrispondente coefficiente stechiometrico
- I reagenti sono considerati con coefficiente negativo

Le quantità n_i delle sostanze all'equilibrio devono soddisfare:

$$\ln K = \left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i} \prod_{i=1}^{n} n_i^{\nu_i}$$

- Dove P è la pressione e K è la costante di equilibrio termodinamico
- K dipende da pressione e temperatura
- n_{tot} è la somma di tutti gli n_i

Vedrete (avete visto?) questi argomenti nel corso di termodinamica

Esercizio: Equilibrio Termodinamico

La quantità n_i di ogni sostanza all'equilibrio…

…dipende dal "grado di avanzamento" ξ della reazione:

$$n_i = n_i^{(0)} + \xi \nu_i$$

- Dove $n_i^{(0)}$ è la quantità iniziale della sostanza

Quindi, dobbiamo trovare uno ξ tale che valga:

$$\ln K = \left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i} \prod_{i=1}^{n} n_i^{\nu_i}$$

$$\text{con: } n_i = n_i^{(0)} + \xi \nu_i$$

$$n_{tot} = \sum_{i=1}^{n} n_i$$

Si tratta quindi di risolvere una equazione non lineare

Esercizio: Equilibrio Termodinamico

Come caso specifico, consideriamo la reazione:

$$CH_4 + H_2O \longrightarrow CO + 3H_2$$

- A 25 atm e 800 K, la costante K vale 163
- Inizialmente abbiamo 4 moli di metano e 10 d'acqua

Trattiamo la nostra equazione come una funzione da azzerare:

$$\ln K - \left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i} \prod_{i=1}^{n} n_i^{\nu_i} = 0$$

- Con $n_i = n_i^{(0)} + \xi\nu_i$ e $n_{tot} = \sum_{i=1}^{n} n_i$

Per prima cosa, dovremo incapsularla in una funzione Matlab

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
end
```

I parametri sono:

- Il vettore **`n0`** con le quantità iniziali
- Il grado di avanzamento **`xi`**
- Il vettore dei coefficienti stechiometrici **`nu`**
- La costante di equilibrio **`K`**
- La pressione **`P`**

Nota: usando dei vettori il codice potrà funzionare per qualsiasi reazione!

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
  n = n0 + xi .* nu;
end
```

Corrisponde a:

$$n_i = n_i^{(0)} + \nu_i \xi$$

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
  n = n0 + xi .* nu;
  ntot = sum(n);
end
```

Corrisponde a:

$$n_{tot} = \sum_{i=1}^{n} \nu_i$$

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
  n = n0 + xi .* nu;
  ntot = sum(n);
  A = log(K);
end
```

Corrisponde a:

$$\underbrace{\ln K}_{A} - \left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i} \prod_{i=1}^{n} n_i^{\nu_i} = 0$$

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
  n = n0 + xi .* nu;
  ntot = sum(n);
  A = log(K);
  B = (P / ntot)^sum(nu);
end
```

Corrisponde a:

$$\ln K - \underbrace{\left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i}}_{B} \prod_{i=1}^{n} n_i^{\nu_i} = 0$$

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
  n = n0 + xi .* nu;
  ntot = sum(n);
  A = log(K);
  B = (P / ntot)^sum(nu);
  C = prod(n.^nu);
end
```

Corrisponde a:

$$\ln K - \left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i} \underbrace{\prod_{i=1}^{n} n_i^{\nu_i}}_{C} = 0$$

Esercizio: Equilibrio Termodinamico

Ci conviene fare i calcoli un passo per volta:

```
function z = thermoeq(n0, xi, nu, K, P)
  n = n0 + xi .* nu;
  ntot = sum(n);
  A = log(K);
  B = (P / ntot)^sum(nu);
  C = prod(n.^nu);
  z = A - B * C;
end
```

Corrisponde all'intera equazione:

$$\ln K - \left(\frac{P}{n_{tot}}\right)^{\sum_{i=1}^{n} \nu_i} \prod_{i=1}^{n} n_i^{\nu_i} = 0$$

Esercizio: Equilibrio Termodinamico

A partire dal file `es_thermoeq.m`

Codificare (come visto nelle slides precedenti) la funzione:

```
function z = thermoeq(n0, xi, nu, K, P)
```

- Definite una funzione anonima per esporre l'interfaccia richiesta da **`fzero`**
- Disegnate la funzione da azzerare
  - **Attenzione:** la nostra **`thermoeq`** assume che **`xi`** sia uno scalare
  - Per disegnare il suo valore occorrerà valutarla ripetutamente!
  - Cercate di capire quanti zeri vi siano
- Utilizzate **`fzero`** per individuare il punto di equilibrio
  - Scegliete il punto di partenza in modo da convergere allo zero corretto

Metodi di Quadratura in Matlab

Esempio: Lunghezza di Curve Parametriche

Supponiamo di voler calcolare la lunghezza di una curva ellittica

Per farlo, ci serve una descrizione formale della traiettoria

- Di solito, un traiettoria si descrive mediante una curva parametrica...
- ...Cioè una funzione con input scalare ed output vettoriale:

$$F : \mathbb{R} \mapsto \mathbb{R}^n$$

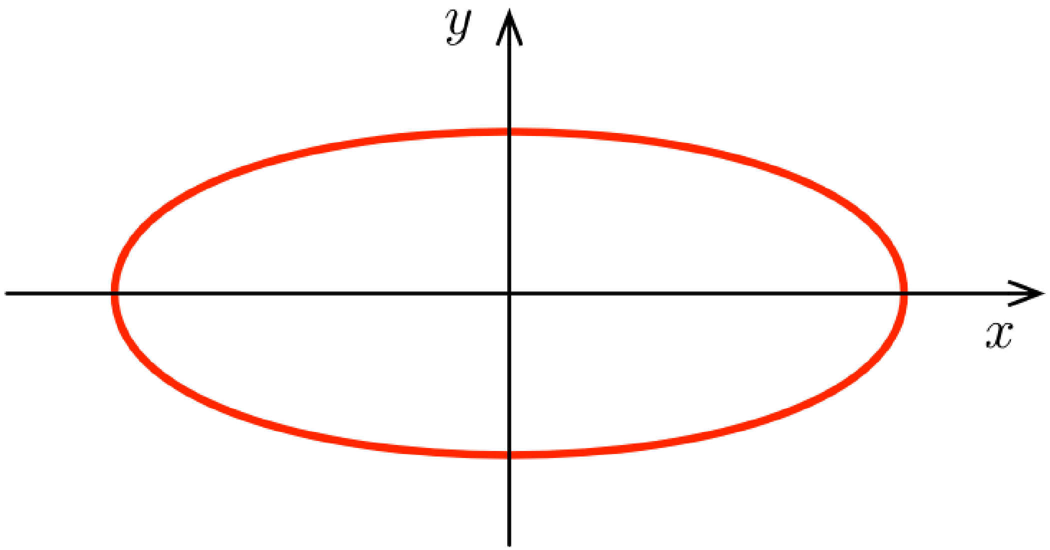
In particolare, una ellissi è descritta da:

$$F(t) = \begin{pmatrix} f_1(t) \\ f_2(t) \end{pmatrix} = \begin{pmatrix} a \cos t \\ b \sin t \end{pmatrix}$$

- a e b sono le lunghezze dei due semi-assi
- L'unica variabile che compare è in questo caso t

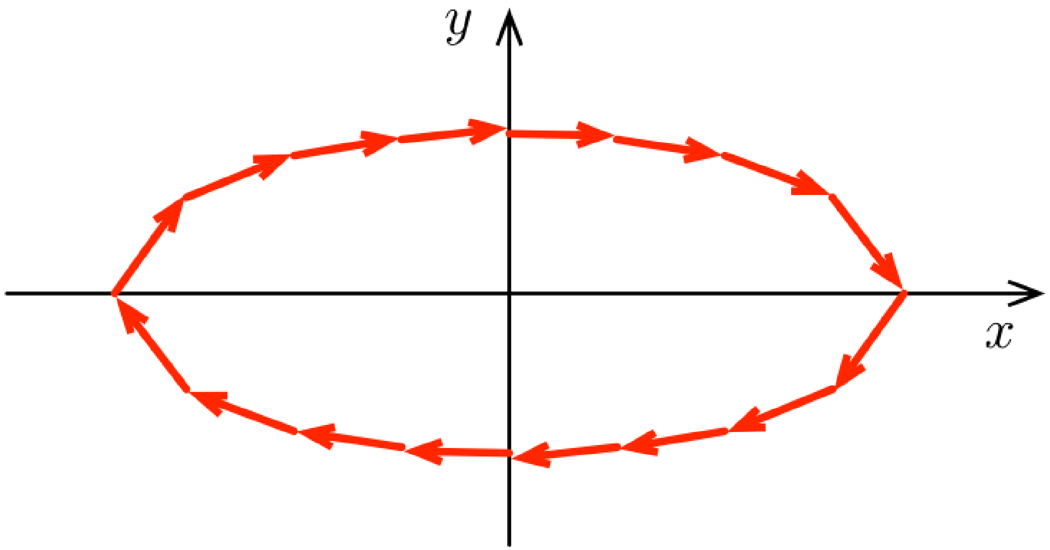
Esempio: Lunghezza di Curve Parametriche

Il risultato può essere qualcosa di questo genere:



Esempio: Lunghezza di Curve Parametriche

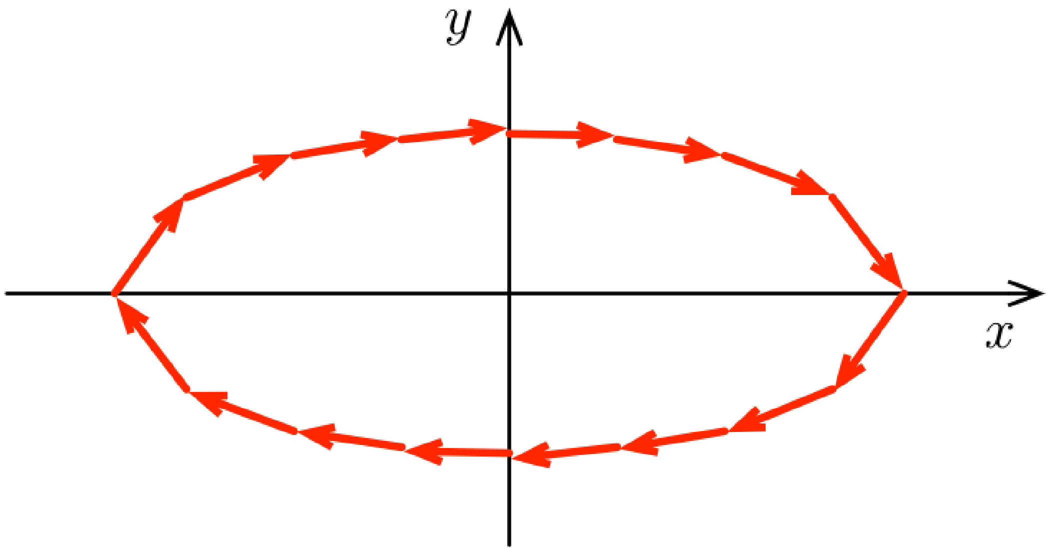
Il risultato può essere qualcosa di questo genere:



- Per calcolare la lunghezza di una curva parametrica…
- …Possiamo immaginare di dividerla in segmenti infinitesimi

Esempio: Lunghezza di Curve Parametriche

Il risultato può essere qualcosa di questo genere:

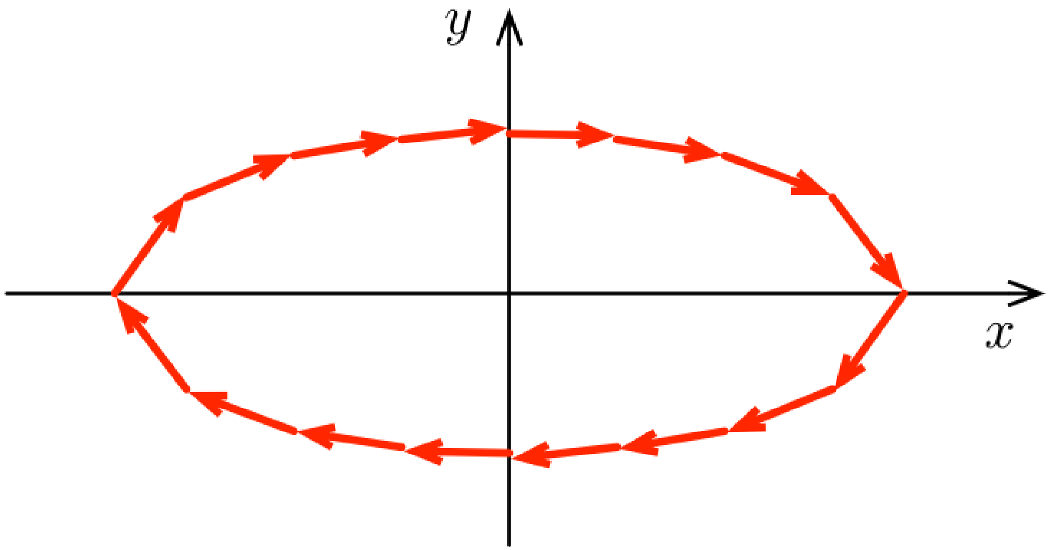


- Ogni segmento infinitesimo corrisponde ad un vettore tangente
- L'equazione si ottiene derivando ogni componente di $F(t)$

$$F'(t) = \begin{pmatrix} f_1'(t) \\ f_2'(t) \end{pmatrix} = \begin{pmatrix} -a \sin t \\ b \cos t \end{pmatrix}$$

Esempio: Lunghezza di Curve Parametriche

Il risultato può essere qualcosa di questo genere:

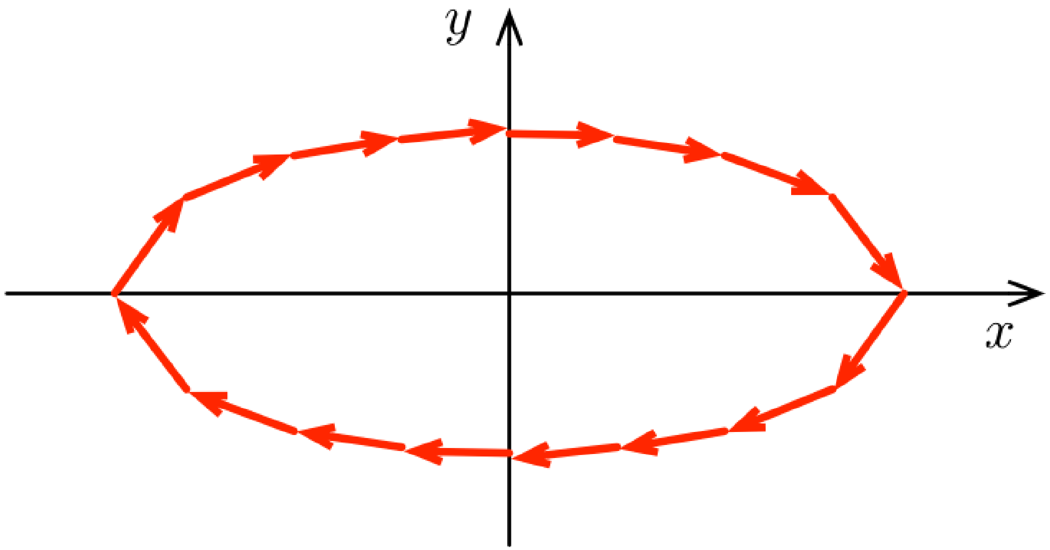


La lunghezza di un vettore tangente è quindi data da:

$$\|F'(t)\| = \sqrt{f_1'(t)^2 + f_2'(t)^2}$$

Esempio: Lunghezza di Curve Parametriche

Il risultato può essere qualcosa di questo genere:



La lunghezza della curva si ottiene integrando quella del vett. tangente:

$$\int_{t_0}^{t_1} \|F'(t)\| dt = \int_{t_0}^{t_1} \sqrt{f_1'(t)^2 + f_2'(t)^2}\, dt$$

- Spesso si ottiene un integrale difficile da calcolare per via simbolica!

Metodi di Quadratura in Matlab

Matlab offre due funzioni principali per effettuare integrazione:

```
function Q = integral(F,XMIN,XMAX)
function Q = trapz(X,Y)
```

Funzionano in modo radicalmente diverso:

`integral` richiede una funzione `F` che abbia un singolo parametro:

```
function Y = F(X)
```

- L'intervallo di integrazione **`XMIN`**..**`XMAX`** viene diviso in sotto-intervalli
- Per ogni sotto-intervallo, **`F`** viene invocata per ottenere campioni
- L'integrale sui sotto-intervalli viene approssimato in base ai campioni
- Eventualmente, si ripete la suddivisione per aumentare la precisione

Metodi di Quadratura in Matlab

Matlab offre due funzioni principali per effetturare integrazioni:

```
function Q = integral(F,XMIN,XMAX)
function Q = trapz(X,Y)
```

Funzionano in modo radicalmente diverso:

`trapz` utilizza il metodo dei trapezi

- Si assume che la funzione da integrare sia stata già campionata
- I vettori **`X`** e **`Y`** contengono le coordinate x e y dei campioni
- Viene calcolata l'area dell'interpolazione lineare a tratti

La funzione **`trapz`** è particolarmente utile per dati sperimentali

- Non c'è una vera funzione da integrare, ma solo delle misurazioni!

Esempio: Lunghezza di Curve Parametriche

Nel caso della nostra ellissi, abbiamo:

$$L = \int_0^{2\pi} \sqrt{(-a \sin t)^2 + (b \cos t)^2}\, dt$$

Per calcolare l'integrale, innanzitutto definiamo la funzione da integrare

- Se l'espressione è semplice, possiamo usare una funzione anonima:

```
dl = @(t) sqrt((-a.*sin(t)).^2 + (b.*cos(t)).^2)
```

- Altrimenti, definiamo un nuova funzione con **`function`**
- In entrambi i casi, usiamo gli operatori elemento per elemento…
- …Perché **`integral`** e **`trapz`** funzionano manipolando vettori

Esempio: Lunghezza di Curve Parametriche

Il prossimo passo dipende dal metodo di integrazione scelto

Se vogliamo usare **`integral`**, possiamo scrivere:

```
L = integral(dl, 0, 2*pi)
```

- La funzione Matlab si occupa del campionamento

Se vogliamo usare **`trapz`**, possiamo scrivere:

```
X = linspace(0, 2*pi) % Valori di t
Y = dl(X) % Lunghezze del vettore tangente
Q = trapz(X, Y)
```

- Il campionamento va fatto prima di invocare la funzione

Stima di Parametri

Esempio: Dimensionamento di una Pista

Vogliamo progettare una pista ellittica

- Assumiamo che il semi-asse b sia fissato
- Il semi-asse a va invece deciso in modo che…
- …La pista abbia la stessa lunghezza di quella di Indianapolis

Stima di Parametri

Sappiamo che la lunghezza della pista è data da:

$$L(a) = \int_0^{2\pi} \sqrt{(a \sin t)^2 + (b \cos t)^2}\, dt$$

- Solo a è variabile $\Rightarrow$ la lunghezza è una funzione di a, i.e. $L(a)$

Se L^* è la lunghezza desiderata, deve valere:

$$L(a) = L^*$$

Si tratta di una equazione non lineare in a!

- La cosa strana è che $L(a)$ è calcolata via integrazione numerica
- …Ma se risolviamo l'eq. con metodi numerici, questo non importa
- …Perché ci basta poter calcolare la funzione da azzerare

Stima di Parametri

Dobbiamo risolvere:

$$L(a) - L^* = 0$$

Quindi potremmo scrivere:

```
b = ...;  % Valore di b
Lt = ...; % Lunghezza di Indianapolis
fz = @(a) curve_length(a, b) - Lt;
[asol, fval, flag] = fzero(fz, a0) % a0: stima iniziale

function L = curve_length(a, b)
    dl = @(t) sqrt((a.*sin(t)).^2 + (b.*cos(t)).^2);
    L = integral(dl, 0, 2*pi);
end
```

Stima di Parametri

Molti problemi di progettazione si possono affrontare così

- Supponiamo che il parametro da determinare si chiami x
- E di avere un vincolo su una grandezza y che dipende da x

A questo punto:

- Prima si trova il modo di calcolare y assumendo che x sia noto
- Poi si incapsula il metodo di calcolo in una funzione $F(x)$
- Quindi si risolve una equazione del tipo:

$$F(x) = y^*$$

- Dove y^* è il valore desiderato per y

Vedremo diversi esempi di qui alla fine del corso!

Esercizio: Bacino Idrico

Esercizio: Bacino Idrico

Un piccolo bacino idrico è riempito artificialmente

Esercizio: Bacino Idrico

Un piccolo bacino idrico è riempito artificialmente

La portata d'acqua in ingresso (in m^3/h) è data da:

$$q(t) = a + b\sin\left(2\pi\frac{t}{24}\right) + c\sin\left(2\pi\frac{t}{13}\right)$$

- I coefficienti a, b, c sono noti

I dati del problema sono nel file `es_flow.m`

- La formula per la portata $q(t)$ è già definita nella funzione:

```
function Q = flow_rate(t, a, b, c)
```

Esercizio: Bacino Idrico

Q1: Si definisca una funzione:

```
function W = intake(t0, t1, a, b, c)
```

- Che calcoli la quantità totale d'acqua che entra nel bacino…
- …Tra due estremi di tempo `t0` e `t1` (misurati in ore)
- Occorrerà calcolare (per via numerica) un integrale!

Si determini quanta acqua entra nel bacino in 72 ore

Q2: Quanto tempo ci vuole perché entrino $200\ m^3$ d'acqua?

- Si tratta di un problema di stima di parametri
- Occorre determinare un valore di t_1 che soddisfi la condizione

Esercizio: Bacino Idrico (2)

Esercizio: Bacino Idrico (2)

Un bacino idrico artificiale è alimentato naturalmente

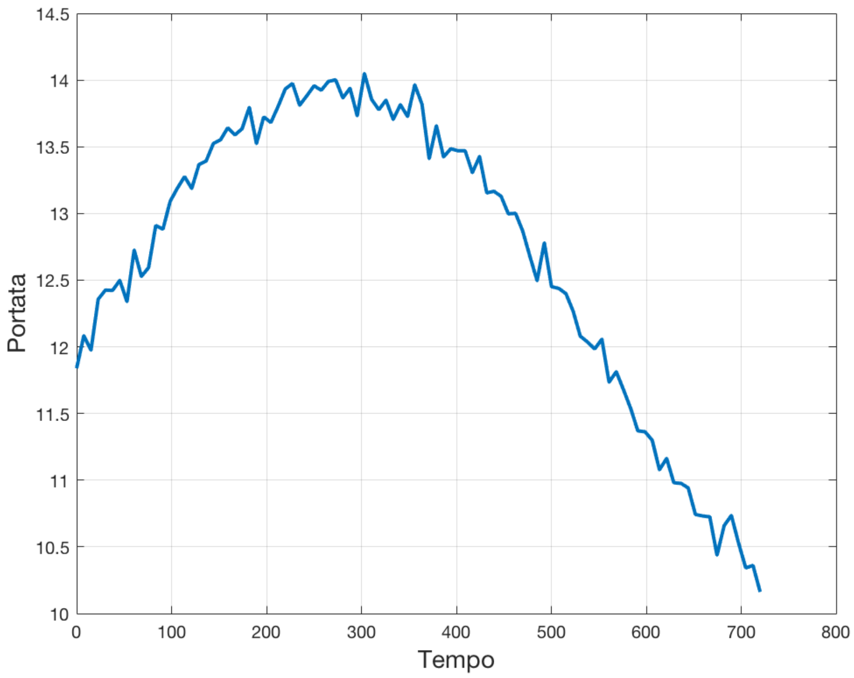
La portata in ingresso (in m^3/h) è misurata ad intervalli regolari

- Un certo numero di misurazioni sono nel file `flow.xlsx`
- Il codice di lettura è disponibile nel file `es_flow2.m`

Esercizio: Bacino Idrico (2)

Q1: Si stimi la quantità d’acqua entrata nel periodo considerato

- Si effettui una integrazione a partire dai dati sperimentali



- Si utilizzi il metodo dei trapezi (funzione `trapz`)

Esercizio: Bacino Idrico (2)

Si assuma poi che parte della portata in ingresso sia dirottabile

In particolare, dirottiamo tutta la portata sopra un certo limite

- In un istante di tempo t, la portata così limitata è data da:

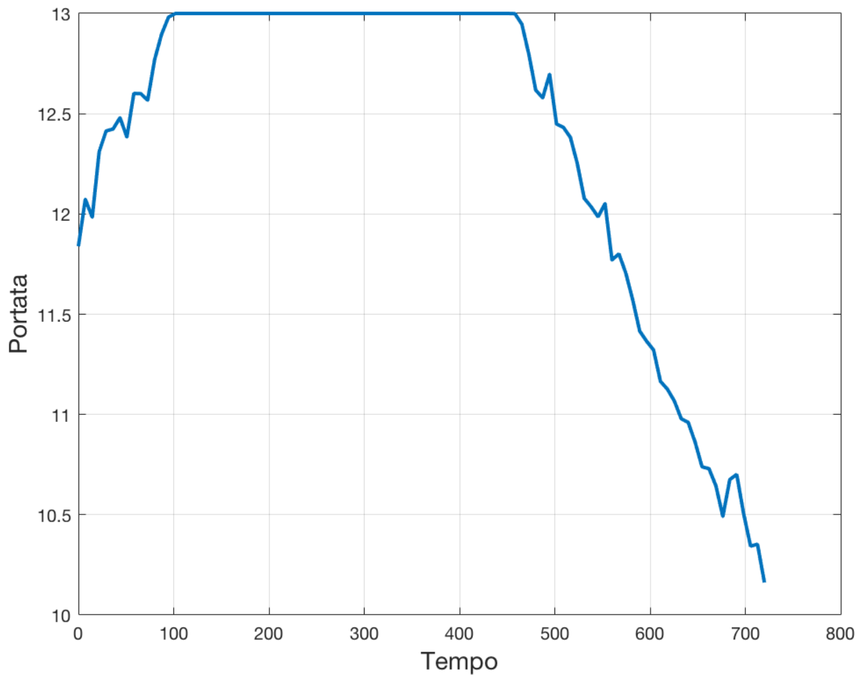
$$\min(L, pwl(T, Q, t))$$

- $pwl(T, Q, t)$ è l'approssimazione lineare a tratti della portata
  - T e Q sono i valori di tempi e portata nei dati sperimentali
  - È necessario usare una qualche forma di interpolazione perché…
  - …la vera portata è nota solo per gli istanti di tempo con misurazioni date
  - $pwl(T, Q, t)$ può essere calcolata in Matlab con `interp1`
- L è il limite al di sopra del quale si incanala l'acqua altrove

Esercizio: Bacino Idrico (2)

Si assuma poi che parte della portata in ingresso sia dirottabile

Con $L = 13$, la portata limitata appare così:



Esercizio: Bacino Idrico (2)

Q2: Si definisca la funzione:

```
function Wtot = flow_with_limit(T, Q, limit)
```

- Che dati i vettori `T` e `Q` con i tempi e le portate misurate…
- …E dato il valore `limit` del limite L…
- …Calcoli la quantità totale d'acqua arrivata nel periodo considerato

In pratica, la funzione deve calcolare:

$$\int_{\min(T)}^{\max(T)} \min(L, pwl(T, Q, t))\, dt$$

Si calcoli la quantità d'acqua arrivata, assumendo $L = 13$

Q3: Si determini il valore di L perché arrivino $8,000\, m^3$ d'acqua

Esercizio: Posizionamento di una Pompa Idraulica

Esercizio: Posizionamento di una Pompa

Si deve posizionare una pompa su una condotta:

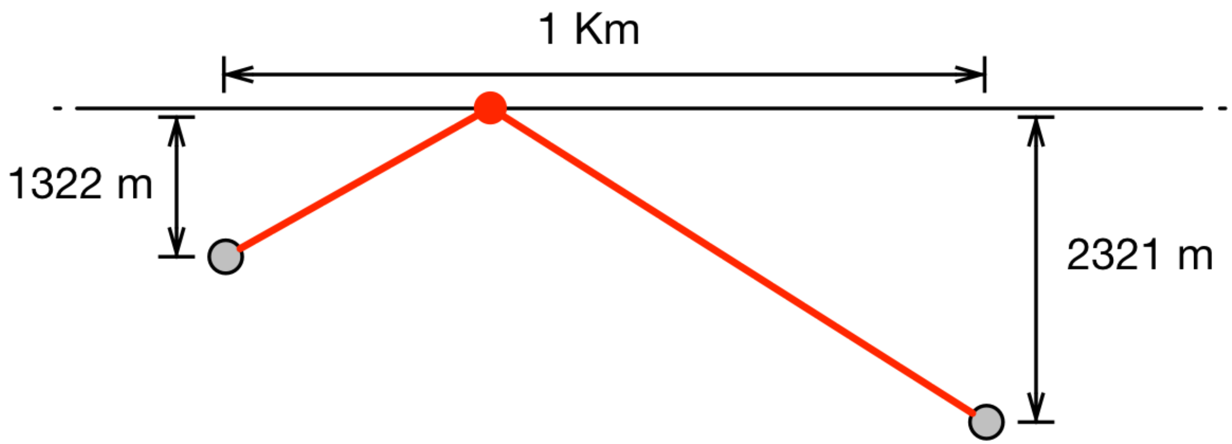
- La pompa deve servire due utenze
- La prima utenza si trova a $1322\ m$ dalla condotta
- La seconda utenza si trova a $2131\ m$ dalla condotta
- La seconda utenza è $1\ Km$ a valle della prima

Le due utenze verranno servite:

- Costruendo delle condotte rettilinee…
- …Che connettono la pompa alle utenze

Vogliamo determinare la posizione ottimale della pompa

Esercizio: Posizionamento di una Pompa



La distanza totale della pompa dalle due utenze è data da:

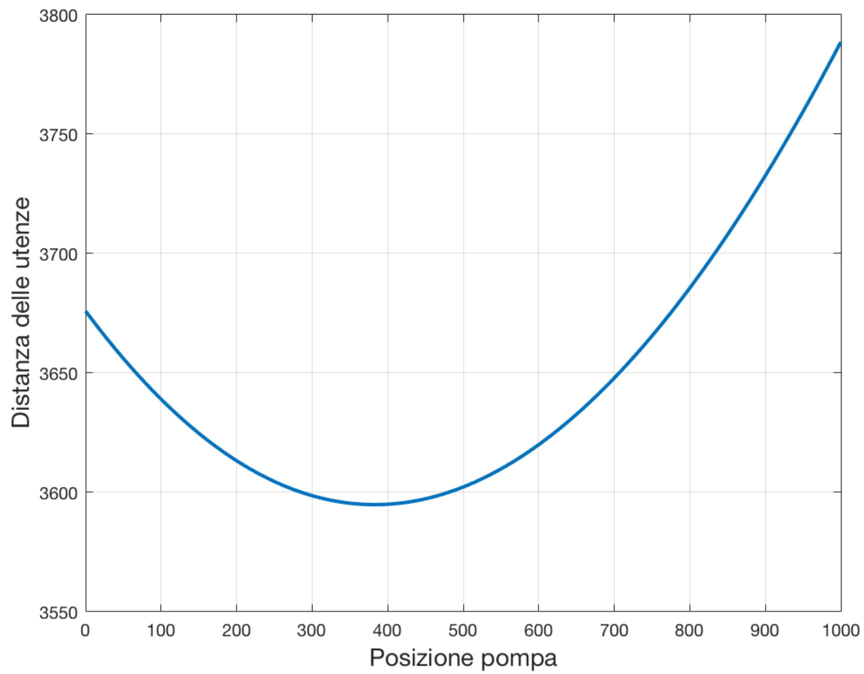
$$dist(a) = \sqrt{(a - x_0)^2 + y_0^2} + \sqrt{(x_1 - a)^2 + y_1^2}$$

- Dove a è la posizione orizzontale della pompa
- x_0, x_1, y_0, y_1 sono le posizioni orizzontali e verticali delle utenze
- Si assume che per la condotta ha $y = 0$

Esercizio: Posizionamento di una Pompa

La posizione ottimale è quella che minimizza la distanza

- Disegnando l'andamento di $dist(a)$ in funzione di a…
- …Possiamo notare che c'è un solo minimo



- Quindi un solo punto in cui la derivata si annulla

Esercizio: Posizionamento di una Pompa

Possiamo così calcolare il minimo risolvendo:

$$dist(a)' = 0$$

La derivata può essere calcolata in forma analitica:

$$dist(a)' = -\frac{x_0 - a}{\sqrt{(x_0 - a)^2 + y_0^2}} - \frac{x_1 - a}{\sqrt{(x_1 - a)^2 + y_1^2}}$$

In alternativa, possiamo approssimare la derivata per via numerica:

$$dist(a)' \simeq \frac{dist(a + \delta) - dist(a)}{\delta}$$

- Con $\delta = a\sqrt{eps}$
- eps è l'epsilon di macchina, accessibile mediante `eps`

Esercizio: Posizionamento di una Pompa

Il file `es_pump_location.m` contiene i dati del problema

Si calcoli il valore di a che minimizza la distanza, risolvendo $dist'(a) = 0$

- **Q1:** si utilizzi l'espressione analitica della derivata
  - Per calcolarla, definite una funzione:

```
function dd = ddist(a, x0, x1, y0, y1)
```

- **Q2:** utilizzate l'approssimazione numerica
  - Per calcolarla, definite una funzione (**`f`** è la funzione da derivare):

```
function dd = ddist2(f, a)
```

Confrontate i due risultati

Esercizio: Planata

Esercizio: Planata

Un aeroplanino di carta viene lanciato in orizzontale

Esercizio: Planata

Un aeroplanino di carta viene lanciato in orizzontale

La traiettoria nel tempo è descritta da una curva parametrica:

$$F(t) = \begin{pmatrix} v_x t \\ y_0 - \frac{1}{2} c g t^2 \end{pmatrix}$$

- v_x è la velocità con cui viene lanciato
- y_0 è la quota iniziale (in m)
- g è l'accelerazione di gravità
- c è un coefficiente noto che tiene conto della resistenza dell'aria

I dati del problema sono disponibili nel file `es_glide.m`

Esercizio: Planata

Q1: Si definisca la funzione:

```
function L = glide_length(vx, g, c, t0, tf)
```

- Che calcoli la strada percorsa dall'aeroplanino…
- …Tra due istanti di tempo `t0` e `tf`

Si determini la strada percorsa tra i due istanti specificati nel file

NOTA: strada percorsa = lunghezza della traiettoria percorsa

Q2: Si determini con che velocità v_x^* deve avvenire il lancio…

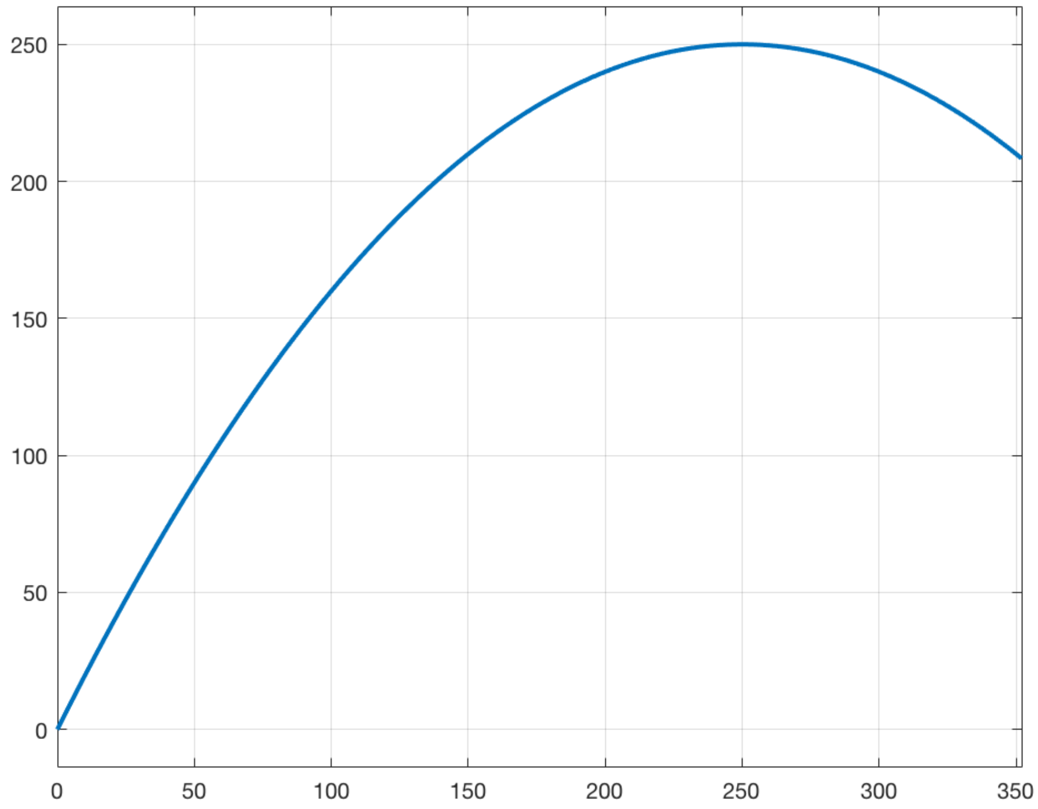
- …Perché la strada percorsa sia pari a $15m$

Esercizio: Punto Intermedio

Esercizio: Punto Intermedio

Si vuole piazzare un misuratore di velocità su un tratto di pista

Il tratto è definito da una parabola con estremi e coefficienti noti



Esercizio: Punto Intermedio

La parabola può essere vista come una curva parametrica

$$F(x) = \begin{pmatrix} x \\ ax^2 + bx + c \end{pmatrix}$$

- Il parametro in questo caso è x, che coincide con la prima coordinata
- Tutti i dati sono disponibili nel file **`es_halfway.m`**

Q1: si definisca una funzione:

```
function L = poly_length(p, x0, x1)
```

- Che, dati un polinomio **`p`** e due estremi per la coordinata x...
- ...Calcoli lunghezza della curva polinomiale tra **`x0`** e **`x1`**
- Ricordate che la derivata di un polinomio si può calcolare con **`polyder`**

Si utilizzi la funzione per calcolare la lunghezza del tratto di pista

Esercizio: Punto Intermedio

Q2: Il misuratore deve essere collocato a metà del tratto

- Si determinino le coordinate di un punto (x', y')...
- ...Che sia equidistante dai due estremi del tratto di pista

Il secondo quesito è un problema di stima di parametri

- Richiede di risolvere una equazione non lineare...
- ...In cui la funzione da azzerare è calcolata via integrazione numerica

Le distanze sono uguali se la loro differenza è nulla, i.e.:

$$F((x, y)) = 0$$

Dove:

- $F((x, y)) = D((x, y), (x_0, y_0)) - D((x, y), (x_1, y_1))$
- Dove $D((x, y), (x_0, y_0))$ è la distanza di (x, y) da (x_0, y_0), e così via