

# Progetto di Intelligenza Artificiale e Laboratorio

## Modulo 3: Uncertainty

Alessio Ballone, Francesco Odierna

### 1 Introduzione

Il presente lavoro come oggetto l'implementazione del *Kalman Filter*, un filtro che valuta lo stato di un *sistema dinamico* a partire da una serie di misurazioni soggette a rumore. Quando parliamo di sistema dinamico ci riferiamo ad un sistema che evolve nel tempo secondo una legge deterministica.

Di seguito verrà introdotta la notazione necessaria all'implementazione dell'algoritmo.

#### 1.1 Kalman Filter

L'algoritmo consiste di due fasi:

- **Prediction:** a partire da uno stato noto si stima lo stato successivo e la covarianza dell'errore. In questo modo otteniamo una stima *a priori* per lo stato successivo.
- **Correction:** viene effettuata la misurazione nel nuovo stato, che può essere affetta da rumore, e viene confrontata con la stima effettuata allo step precedente. Le stime vengono aggiornate utilizzando una media pesata con cui si dà maggior peso alle stime più certe. In questo modo si incorpora la nuova misurazione nella stima *a priori* così da ottenere una stima *a posteriori*.

Il processo da stimare è un il seguente:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}$$

dove  $w_{k-1} \sim \mathcal{N}(0, Q)$  rappresenta la quantità di rumore che affligge la transizione di stato e  $Q$  è la matrice di covarianza che governa il rumore di processo.

### 1.1.1 Prediction

In questa fase si parte dallo stato  $\hat{x}_{k-1}$  e si stima lo stato successivo  $\hat{x}_k$  sulla base del un modello di transizione rappresentato da una matrice  $A$ , di una matrice di controllo  $B$  e di un vettore di controllo  $u$ . Viene inoltre stimata la covarianza dell'errore  $P_k$ :

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k$$

$$P_k^- = AP_{k-1}A^T + Q$$

### 1.1.2 Correction

In questa fase viene effettuata la misurazione, che può essere affetta da rumore, e si confronta la stima con la misurazione. In fase di correzione teniamo conto del *Kalman gain*  $K$ , che permette di confrontare l'incertezza della predizione e l'incertezza della misurazione, scegliendo come nuova stima quella che minimizza l'incertezza. In particolare il Kalman gain quantifica la percentuale di errore nella stima rispetto all'errore totale, ossia:

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1}$$

Valori alti di  $K$  (vicini a 1) fanno sì che l'incertezza della misurazione sia bassa e quindi la nuova stima sarà più vicina alla misurazione. Valori bassi di  $K$  (vicini a 0) fanno sì che l'incertezza della misurazione sia elevata e quindi la nuova stima sarà più vicina alla stima precedente. La matrice  $H$  rappresenta il modello di osservazione che permette di mappare il vero stato in quello osservato, ossia  $z_k$ :

$$z_k = H\hat{x}_k + v_k$$

Il valore misurato è affetto da una quantità di rumore di osservazione pari a  $v_k \sim \mathcal{N}(0, R)$ , dove  $R$  è la matrice di covarianza che governa il rumore di osservazione. La stima viene aggiornata con la misurazione  $z_k$  nel seguente modo:

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$$

Inoltre la covarianza dell'errore viene aggiornata nel seguente modo:

$$P_k = (I - KH)P_k^-$$

## 2 Implementazione

L'algoritmo è stato implementato in Python, seguendo la notazione descritta in precedenza. Il codice è organizzato in due moduli:

- **utilities**: contiene le funzioni `kalman predict` e `kalman update` che permettono di realizzare gli step di predict e update, rispettivamente, descritti in precedenza.
- **main**: simula il processo di spostamento di un veicolo in moto uniformemente accelerato in due dimensioni.

In Figura 1 troviamo il codice della funzione `kalman predict` e in Figura 2 troviamo il codice della funzione `kalman correct`

```
def kalman_predict(A, B, P_curr, Q, x_curr, u):  
    '''  
    :param A: modello di transizione  
    :param B: control input  
    :param P_curr: covarianza errore stato corrente  
    :param Q: covarianza della distribuzione  $N(0, Q)$  che governa il rumore di transizione  
    :param x_curr: stato corrente  
    :param u: vettore di controllo  
    :return:  
        x_pred: stato predetto  
        P_pred: covarianza errore predetta  
    '''  
  
    #  $x_{k+1} = Ax_k + Bu_k$   
    x_pred = A * x_curr + B * u  
  
    #  $P_{k+1} = A P_k A^T + Q$   
    P_pred = A * P_curr * A.T + Q  
  
    return x_pred, P_pred
```

Figura 1: Funzione che implementa lo step predict

## 3 Esperimenti e risultati

### 3.1 Il processo

Come accennato nella sezione precedente, il processo simulato è lo spostamento di un veicolo in moto uniformemente accelerato in due dimensioni. Nel caso generale sappiamo che se la posizione del nostro veicolo al tempo

```

def kalman_correct(H, P_pred, R, z, x_pred):
    '''
    :param H: modello di osservazione che mappa x_real nella misurazione z
    :param P_pred: covarianza dell'errore predetta
    :param R: covarianza distribuzione [N(0, R)] che governa il rumore di osservazione
    :param z: misurazione del vero stato
    :param x_pred: stato predetto
    :return:
        K: Kalman gain
        x_meas: stato reale misurato
        P_real: covarianza errore reale
    '''

    # Kalman gain computation

    #  $S = H P_{pred} H^T + R$ 
    S = H * P_pred * H.T + R

    #  $K = P_{pred} H^T S^{-1}$ 
    K = P_pred * H.T * S.I

    # Update

    #  $y = z - Hx_{pred}$ 
    y = z - H*x_pred

    #  $x_{meas} = x_{pred} + Ky$ 
    x_meas = x_pred + K*y

    #  $P_{real} = (I-KH)P_{pred} = P_{pred} - KHP$ 
    P_real = P_pred - K * H * P_pred

    return K, x_meas, P_real

```

Figura 2: Funzione che implementa lo step correct

$t$  è  $x$  allora al tempo  $t + 1$  la sua posizione sarà  $x + [(t + 1) - t]\dot{x}$ , dove  $\dot{x}$  è la velocità dell'oggetto. Considerando anche i valori dell'accelerazione possiamo dire che la posizione dell'oggetto al tempo  $t + 1$  è pari a

$$x + [(t + 1) - t]\dot{x} + \frac{1}{2}a_x[(t + 1) - t]^2$$

Di seguito formalizziamo le equazioni appena viste per stati rappresentati da un vettore a quattro componenti:  $x, y, \dot{x}, \dot{y}$ , dove  $x$  e  $y$  rappresentano la posizione del veicolo e  $\dot{x}, \dot{y}$  rappresentano la velocità del veicolo:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}$$

I cambiamenti di tempo sono denotati con  $\Delta t$ . Le equazioni per tracciare la velocità e la posizione del veicolo sono le seguenti:

$$x_{t+1} = x_t + \Delta t \dot{x} + \frac{1}{2}a_x \Delta t^2$$

$$y_{t+1} = y_t + \Delta t \dot{y} + \frac{1}{2}a_y \Delta t^2$$

$$\dot{x}_{t+1} = \dot{x}_t + a_x \Delta t$$

$$\dot{y}_{t+1} = \dot{y}_t + a_y \Delta t$$

In forma matriciale risulta:

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$$

Dove  $\mathbf{A}$  è la matrice di transizione,  $\mathbf{B}$  è la matrice di controllo e  $\mathbf{u}$  è il vettore di controllo e sono definiti come segue:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} a_x \\ a_y \end{bmatrix}$$

Nei test effettuati il vettore  $\mathbf{u}$  è assunto costante, tuttavia questa subisce delle variazioni dovute al rumore di processo che segue le distribuzioni  $\mathcal{N}(0; \sigma_{accX}^2)$  e  $\mathcal{N}(0; \sigma_{accY}^2)$ , rispettivamente per la componente  $x$  e  $y$ . Vista l'influenza che i cambiamenti di accelerazione hanno sulla posizione e sulla

velocità dell'oggetto, la matrice di covarianza del rumore di processo  $\mathbf{Q}$  è definita come:

$$\mathbf{Q} = \begin{bmatrix} (\frac{1}{2}\Delta t^2\sigma_{accX})^2 & 0 & 0 & 0 \\ 0 & (\frac{1}{2}\Delta t^2\sigma_{accY})^2 & 0 & 0 \\ 0 & 0 & (\Delta t\sigma_{accX})^2 & 0 \\ 0 & 0 & 0 & (\Delta t\sigma_{accY})^2 \end{bmatrix}$$

per semplicità abbiamo assunto che tutte le variabili di stato osservate siano tra loro indipendenti, cosa non vera nella realtà in quanto c'è un legame tra la posizione e la velocità.

Anche la posizione misurata è soggetta a rumore e questo segue le distribuzioni  $\mathcal{N}(0, \sigma_{posX}^2)$  e  $\mathcal{N}(0, \sigma_{posY}^2)$ . La matrice di covarianza  $\mathbf{R}$  è definita come segue:

$$\mathbf{R} = \begin{bmatrix} \sigma_{posX}^2 & 0 \\ 0 & \sigma_{posY}^2 \end{bmatrix}$$

Anche in questo caso assumiamo le misurazioni e gli errori siano indipendenti.

In questo modo possiamo variare  $\sigma_{accX}, \sigma_{accY}, \sigma_{posX}$  e  $\sigma_{posY}$  agendo sul rumore che affligge il processo reale e le misurazioni ottenute dai sensori.

## 3.2 Esperimenti

Abbiamo effettuato diversi test agendo sul rumore di processo e quello di misurazione, facendo quindi variare  $\sigma_{accX}, \sigma_{accY}, \sigma_{posX}, \sigma_{posY}$ . Nei test sono stati utilizzati i seguenti valori iniziali:  $\mathbf{x}_0 = [0 \ 0 \ 0 \ 0]^T$  e  $\mathbf{P}_0 = \mathbf{I}$  e abbiamo assunto  $\sigma_{accX} = \sigma_{accY}$  e  $\sigma_{posX} = \sigma_{posY}$  a cui ci riferiremo con  $\sigma_{acc}$  e  $\sigma_{pos}$ , rispettivamente. Dove non specificato lo stato iniziale e lo stato reale, indicato con  $\mathbf{x}_{real}$ , coincidono. Per praticità vengono riportati solo i valori reali, misurati e stimati relativi alla posizione  $x$ .

### 3.2.1 Rumore di processo e di misurazione bassi

Il caso più semplice è quello in cui ipotizziamo che il processo reale segua il modello di processo e che le misurazioni effettuate dai sensori siano poco rumorose. Abbiamo posto  $\sigma_{acc} = 10^{-4}$ ,  $\sigma_{pos} = 10^{-6}$ . Come si può osservare in Figura 3 il processo reale segue il modello di processo e l'errore sulle misurazioni è molto basso in quanto queste risultano sovrapposte al valore reale.

### 3.2.2 Rumore di processo medio e rumore di misurazione basso

In questo caso ipotizziamo che il processo reale si discosti leggermente dal modello di processo e che le misurazioni effettuate dai sensori siano affette

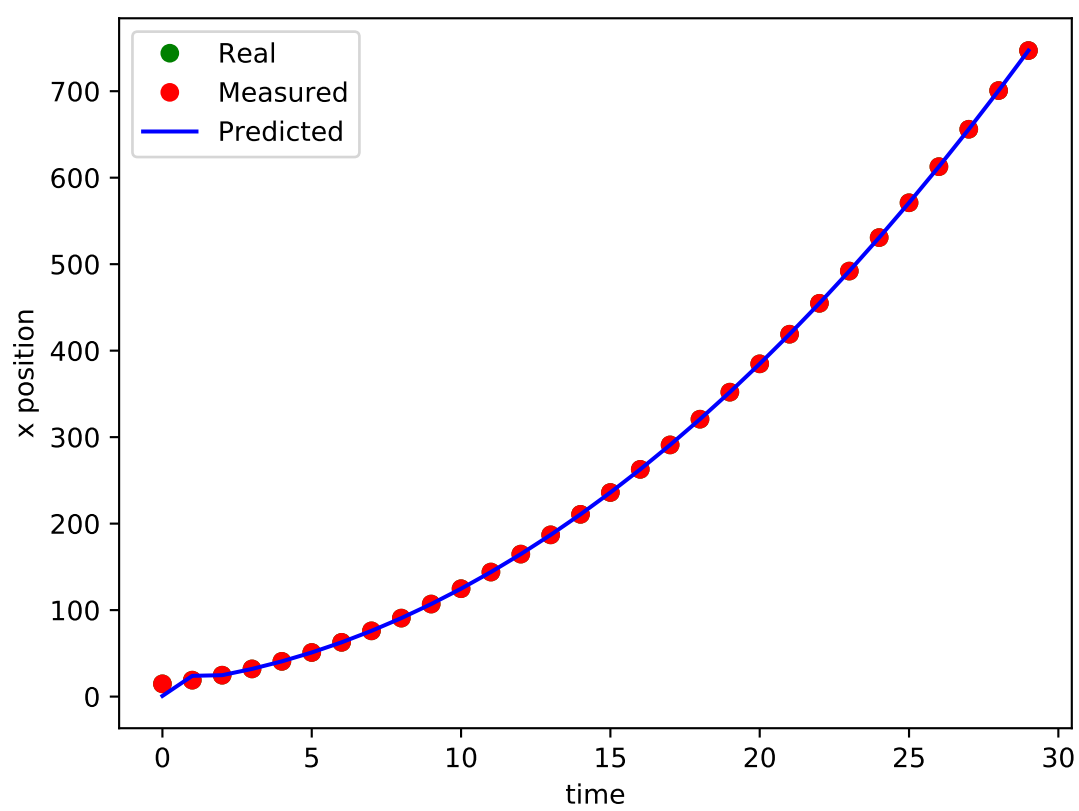


Figura 3: Stima di  $x$  con basso rumore di processo e di misurazione

da un basso rumore. Abbiamo posto  $\sigma_{acc} = 20, \sigma_{pos} = 10^{-6}$ . Come si può osservare in Figura 4 il processo reale è affetto da rumore, tuttavia l'errore commesso nelle misurazioni è molto basso, infatti queste si sovrappongono al processo reale e il filtro riesce comunque a effettuare buone predizioni.

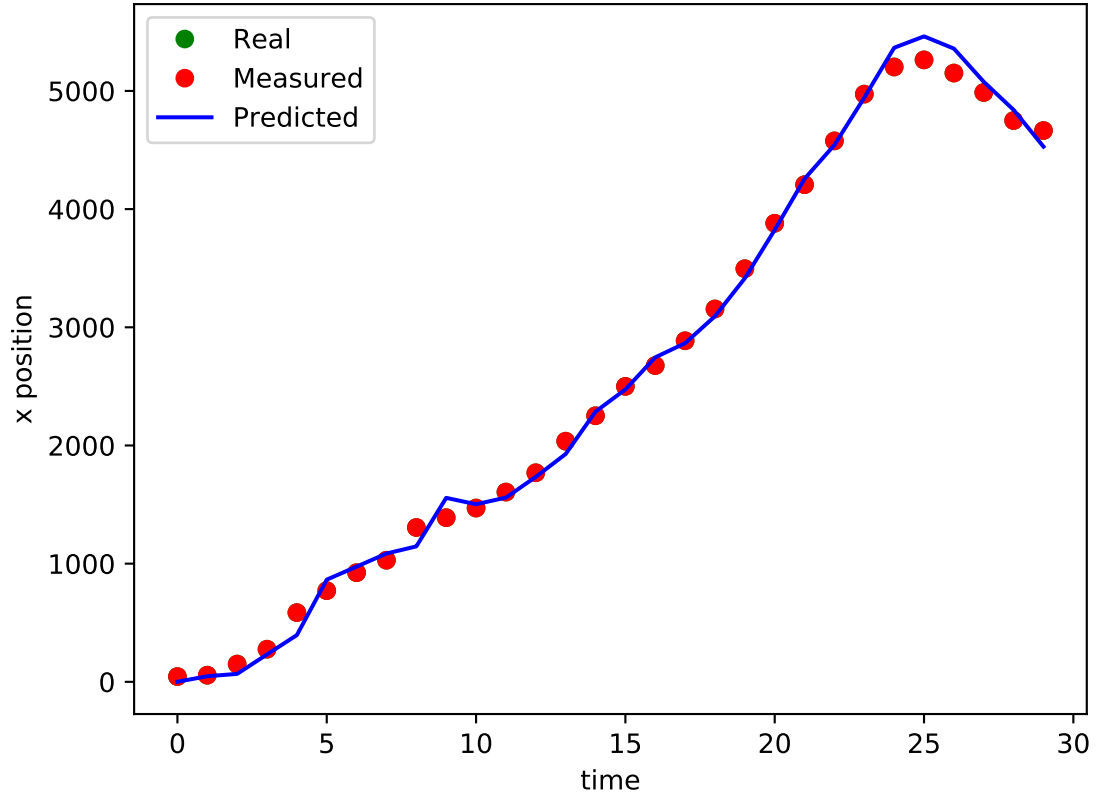


Figura 4: Stima di  $x$  con rumore di processo medio e rumore di misurazione basso

### 3.2.3 Rumore di processo alto e rumore di misurazione basso

In questo caso ipotizziamo che il processo reale si discosti completamente dal modello di processo e che le misurazioni effettuate dai sensori siano affette da un basso rumore. In questo caso abbiamo posto  $\sigma_{acc} = 150, \sigma_{pos} = 10^{-6}$ . Come si può osservare in Figura 5 il rumore che affligge il processo reale non impedisce al filtro di effettuare una predizione accettabile, ciò è dovuto al fatto che le misurazioni sono precise.



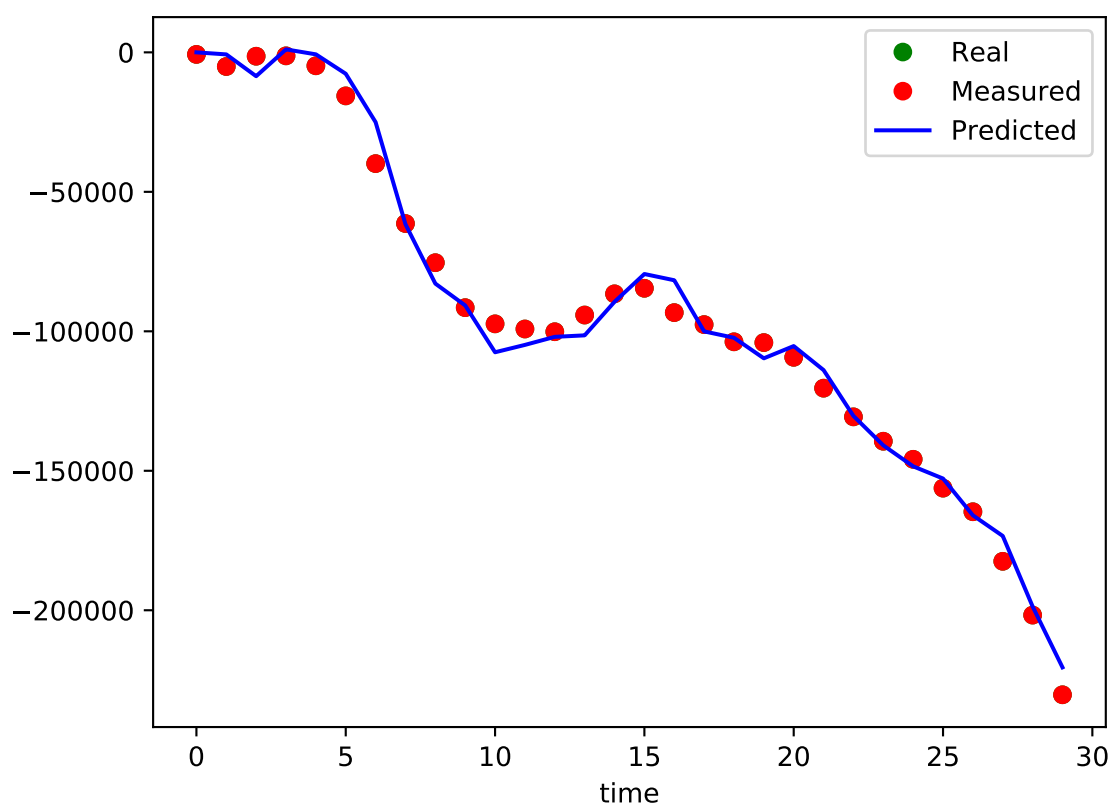


Figura 5: Stima di  $x$  con rumore di processo alto e rumore di misurazione basso

### 3.2.4 Casi speculari: rumore di processo basso e rumore di misurazione crescente

Nei casi speculari in cui si tiene basso il rumore di processo e si fa crescere il rumore di misurazione, le misurazioni saranno rumorose e quindi le predizioni, man mano che il rumore aumenta, saranno sempre meno accurate. Per dare un'idea di quanto sosteniamo in Figura 6 troviamo il caso in cui il rumore di misurazione è elevato e il rumore di processo è basso. Ciò fa sì che i valori misurati si discostino totalmente dal processo reale e di conseguenza le predizioni sono errate. I valori sono stati impostati rispettivamente a  $\sigma_{acc} = 10^{-4}$ ,  $\sigma_{pos} = 150$ .

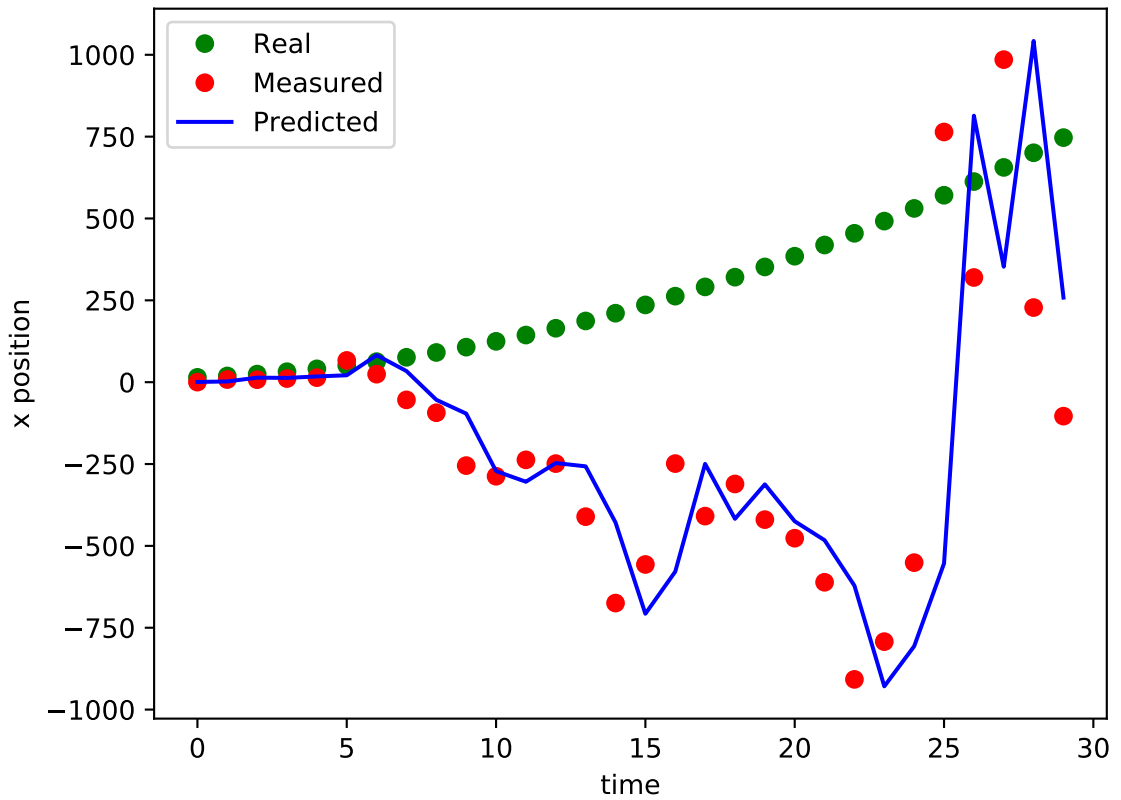


Figura 6: Stima di x con rumore di processo basso e rumore di misurazione elevato

### 3.2.5 Casi reali

Fino ad ora abbiamo considerato i casi semplici, ossia quelli in cui si assume che uno tra i due rumori (di processo o di misurazione) sia basso e l'altro subisce variazioni. Nella realtà le cose sono diverse e solitamente sono presenti entrambi i rumori. Di seguito vediamo dei casi in cui sia il processo che le misurazioni sono afflitti da rumore.

**Rumore medio** In questo caso il rumore di processo e di misurazione sono medi. I valori sono stati impostati rispettivamente a  $\sigma_{acc} = 10, \sigma_{pos} = 20$ . In Figura 7 possiamo notare che il rumore influenza le predizioni, tuttavia la predizione non è male in quanto le misurazioni, a parte qualche errore, seguono il processo reale. Tuttavia ripetendo l'esperimento più volte abbiamo notato che possono verificarsi dei casi in cui le misurazioni sono totalmente diverse dal processo reale. In questo caso, come si può osservare in Figura 8, le predizioni sono totalmente sbagliate.

**Rumore alto** In questo caso il rumore di processo e di misurazione sono elevati. I valori sono stati impostati rispettivamente a  $\sigma_{acc} = 60, \sigma_{pos} = 90$ . In Figura 9 si può osservare che le misurazioni si avvicinano a tratti al processo reale e quindi l'approssimazione, anche se lontana dall'essere una buona approssimazione, non è malvagia. Ovviamente questo caso capita raramente. Il caso più frequente è riportato in Figura 10. Si nota che le predizioni sono totalmente errate in quanto le misurazioni di discostano di molto dal processo reale.

### 3.2.6 Esperimenti con variazione delle condizioni iniziali

In tutti gli esperimenti visti fino ad ora abbiamo ipotizzato che lo stato reale  $\mathbf{x}_{real}$  e lo stato iniziale  $\mathbf{x}_0$  coincidano. Cosa succede se ciò non accade? Negli esperimenti che seguono abbiamo ipotizzato che lo stato reale e lo stato iniziale non coincidano, in particolare lo stato reale è  $\mathbf{x}_{real} = [70 \ 120 \ 80 \ 160]^T$  e lo stato iniziale è  $\mathbf{x}_0 = [0 \ 0 \ 0 \ 0]^T$ , inoltre entrambi i rumori (di processo e di misurazione) possono variare.

**Rumore basso** In questo caso i rumori di processo e di misurazione sono bassi. I valori sono stati impostati rispettivamente a  $\sigma_{acc} = 10^{-4}, \sigma_{pos} = 10^{-6}$ . In Figura 11 possiamo notare che il filtro fornisce una buona approssimazione nonostante lo stato iniziale sia diverso da quello reale. Ciò è dovuto ad un rumore basso sia sul processo che sulle misurazioni. A parte la predizione iniziale, che risulta sfasata, le altre predizioni sono corrette.

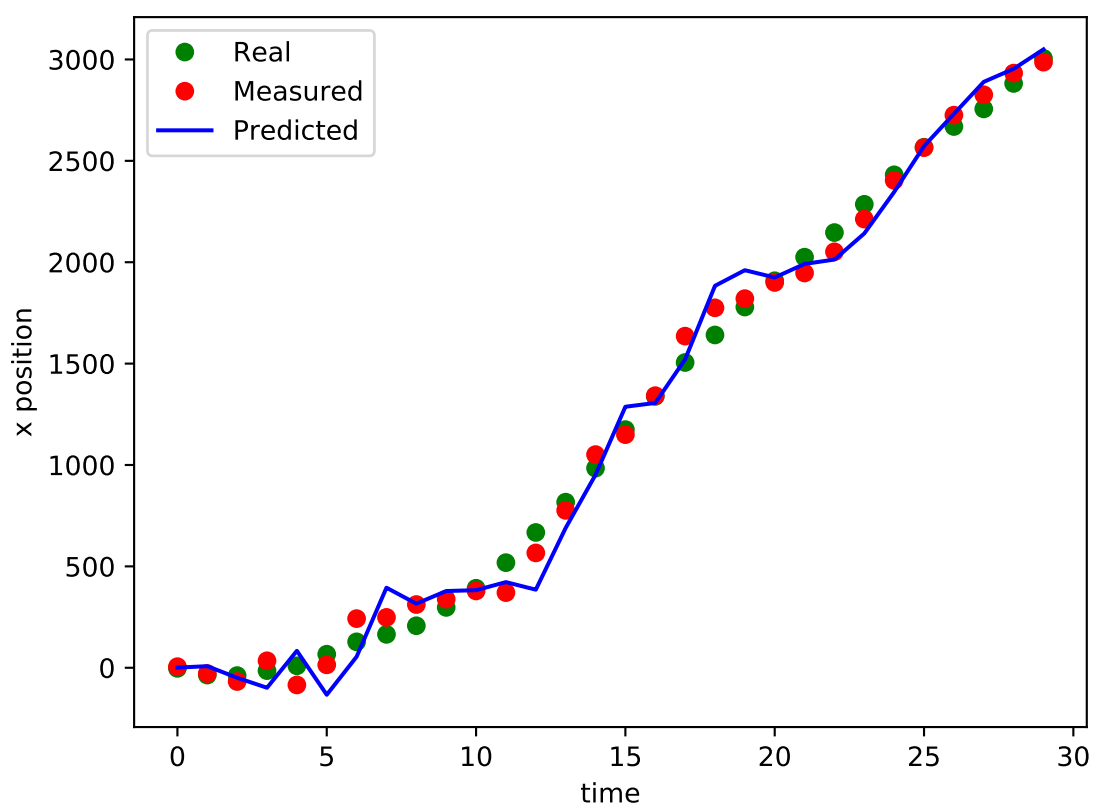


Figura 7: Stima di  $x$  con rumore di processo e di misurazione medio. Caso ottimo.

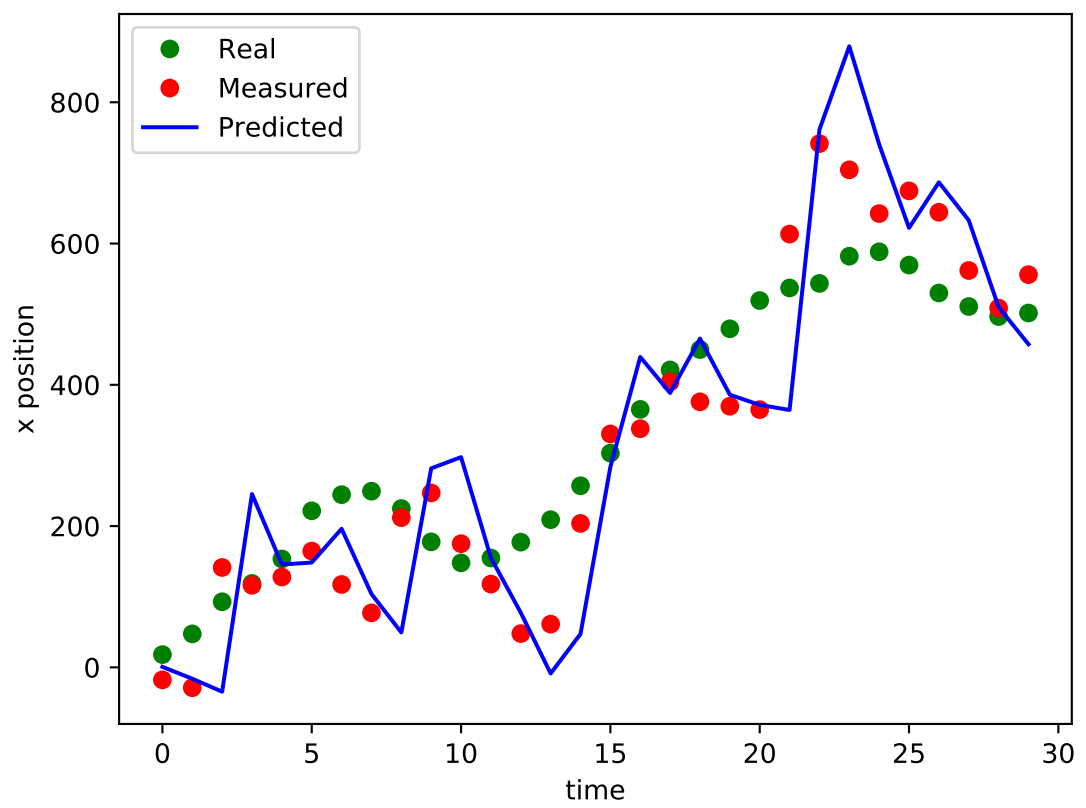


Figura 8: Stima di  $x$  con rumore di processo e di misurazione medio. Caso pessimo.

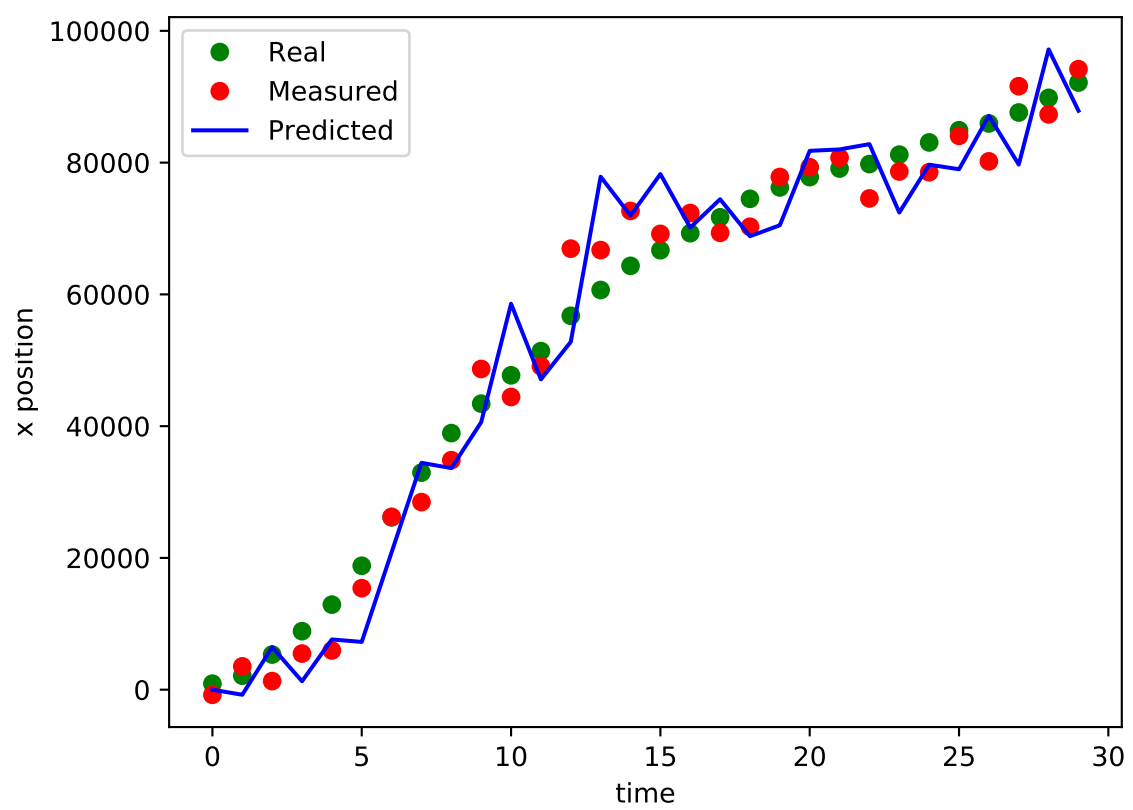


Figura 9: Stima di  $x$  con rumore di processo e di misurazione alto. Caso ottimo.

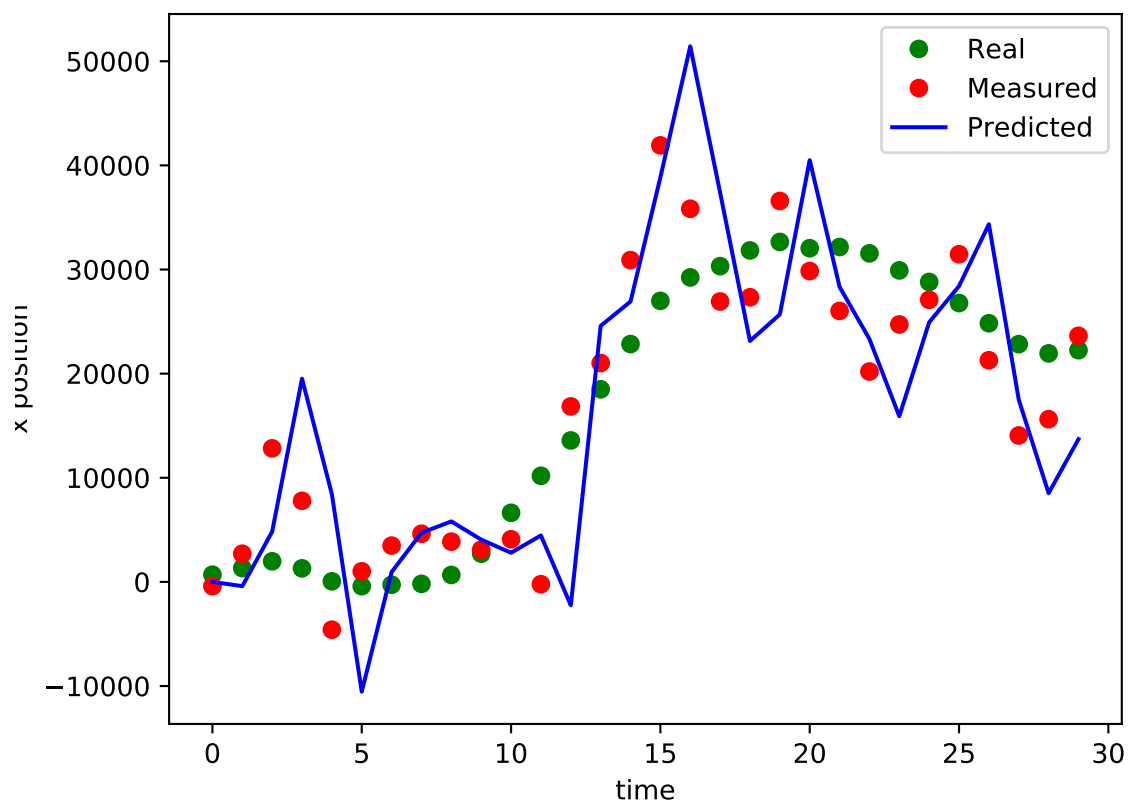


Figura 10: Stima di  $x$  con rumore di processo e di misurazione alto. Caso pessimo.

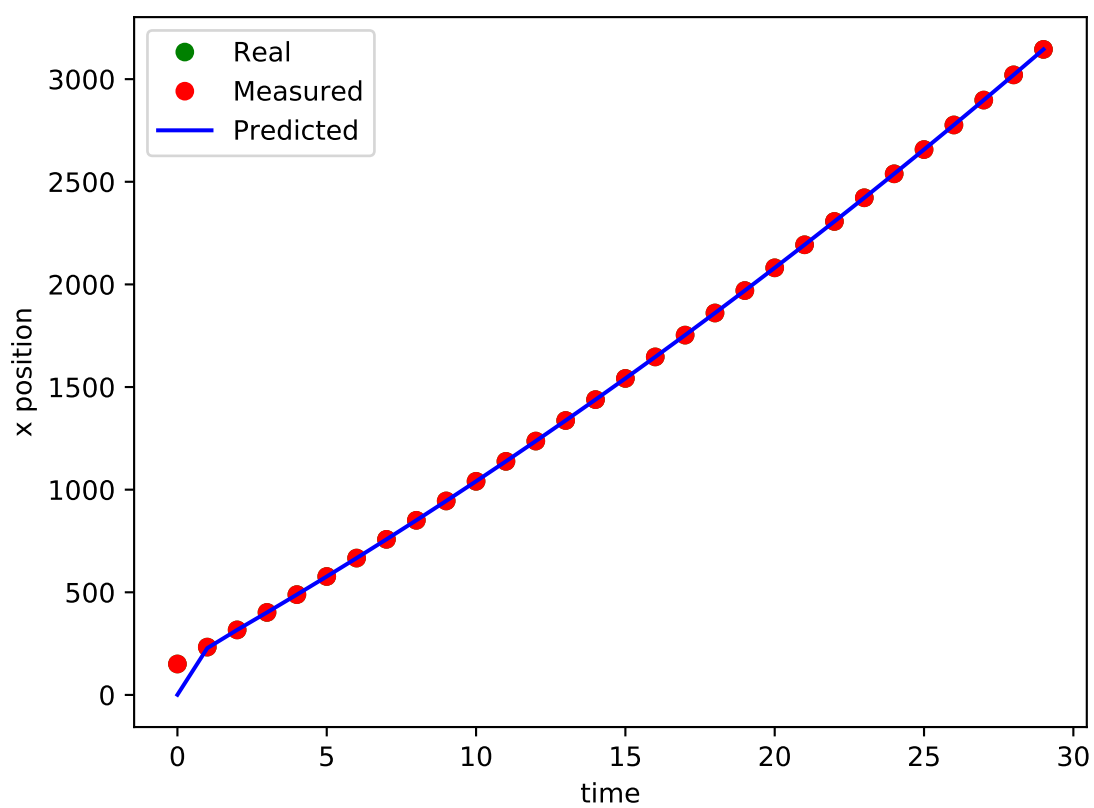


Figura 11: Stima di  $x$  con rumore di processo e di misurazione basso e stato iniziale diverso dallo stato reale.



**Rumore medio** In questo caso il rumore di processo e di misurazione sono medi. I valori sono stati impostati rispettivamente a  $\sigma_{acc} = 10, \sigma_{pos} = 20$ . In Figura 12 possiamo notare che, a differenza di prima, la partenza errata ha una maggiore influenza e, anche nel caso migliore, la predizione non è buona come nel caso in cui la partenza sia perfetta. Il caso pessimo è mostrato in Figura 13.

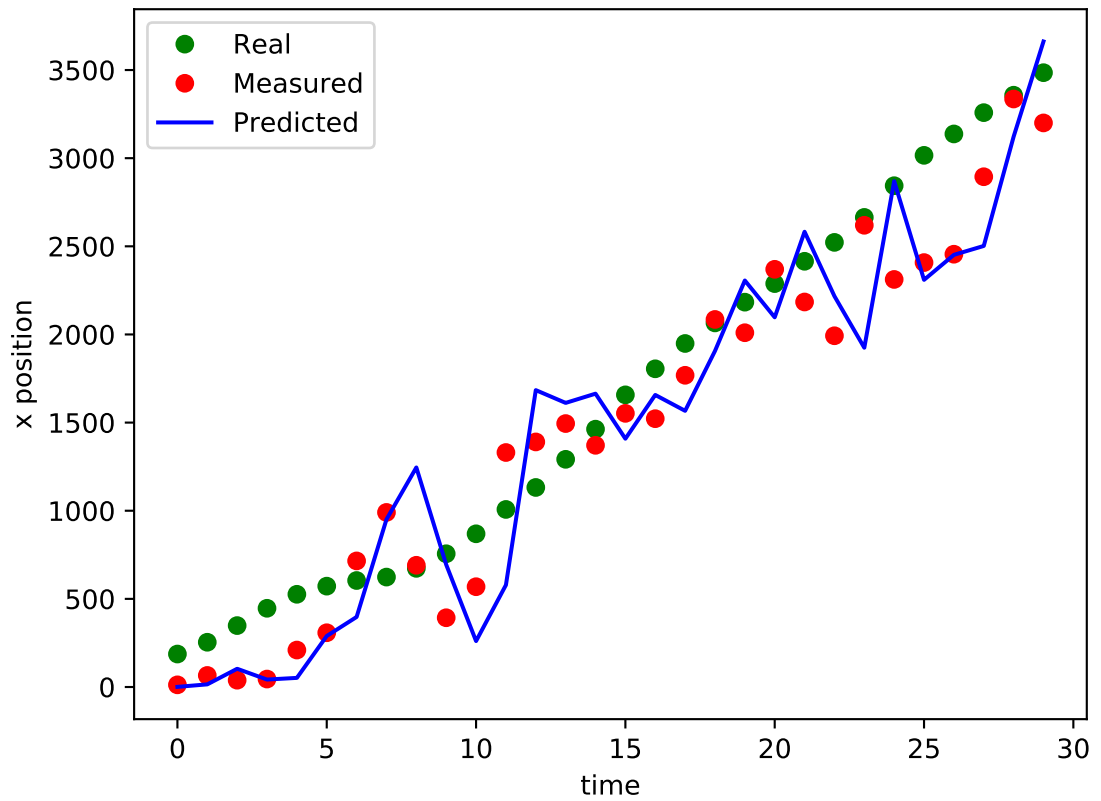


Figura 12: Stima di  $x$  con rumore di processo e di misurazione medio e stato iniziale diverso dallo stato reale. Caso ottimo.

**Rumore alto** In questo caso il rumore di processo e di misurazione sono elevati. I valori sono stati impostati rispettivamente a  $\sigma_{acc} = 60, \sigma_{pos} = 90$ . Come si può notare dalla Figura 14 le predizioni, influenzate anche da una condizione di partenza errata, sono totalmente sbagliate in quanto le misurazioni si discostano di molto dal processo reale.

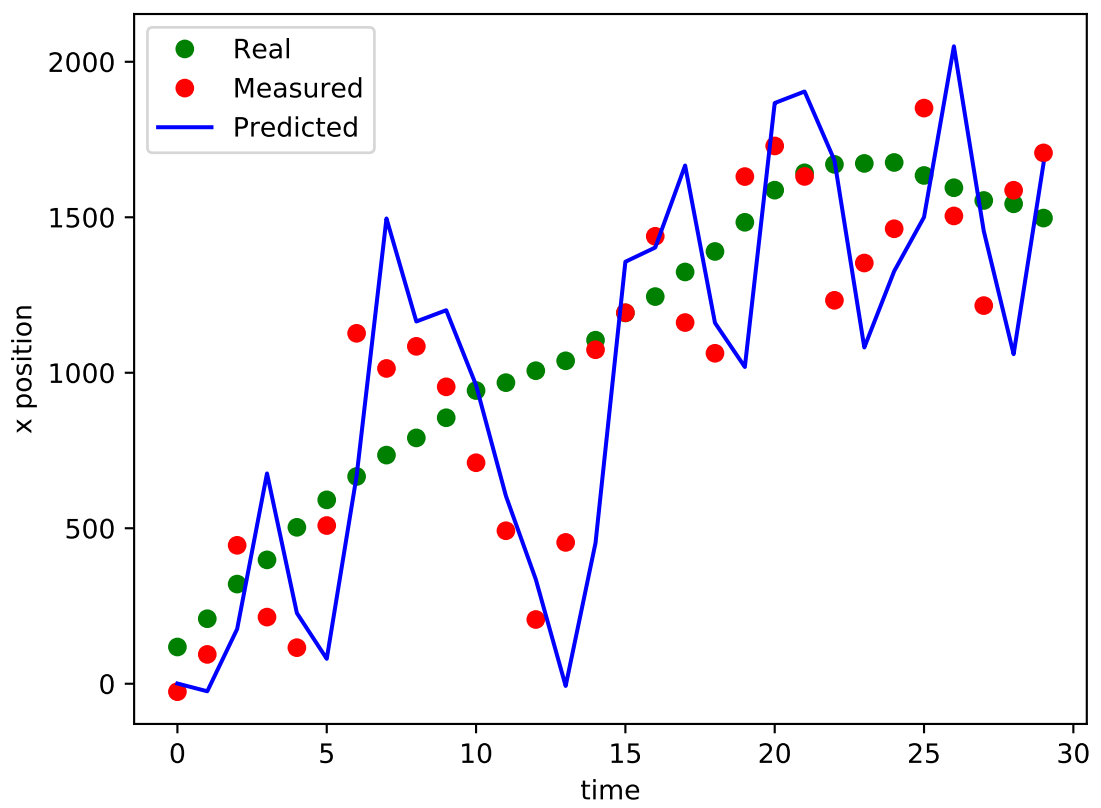


Figura 13: Stima di  $x$  con rumore di processo e di misurazione medio e stato iniziale diverso dallo stato reale. Caso pessimo.

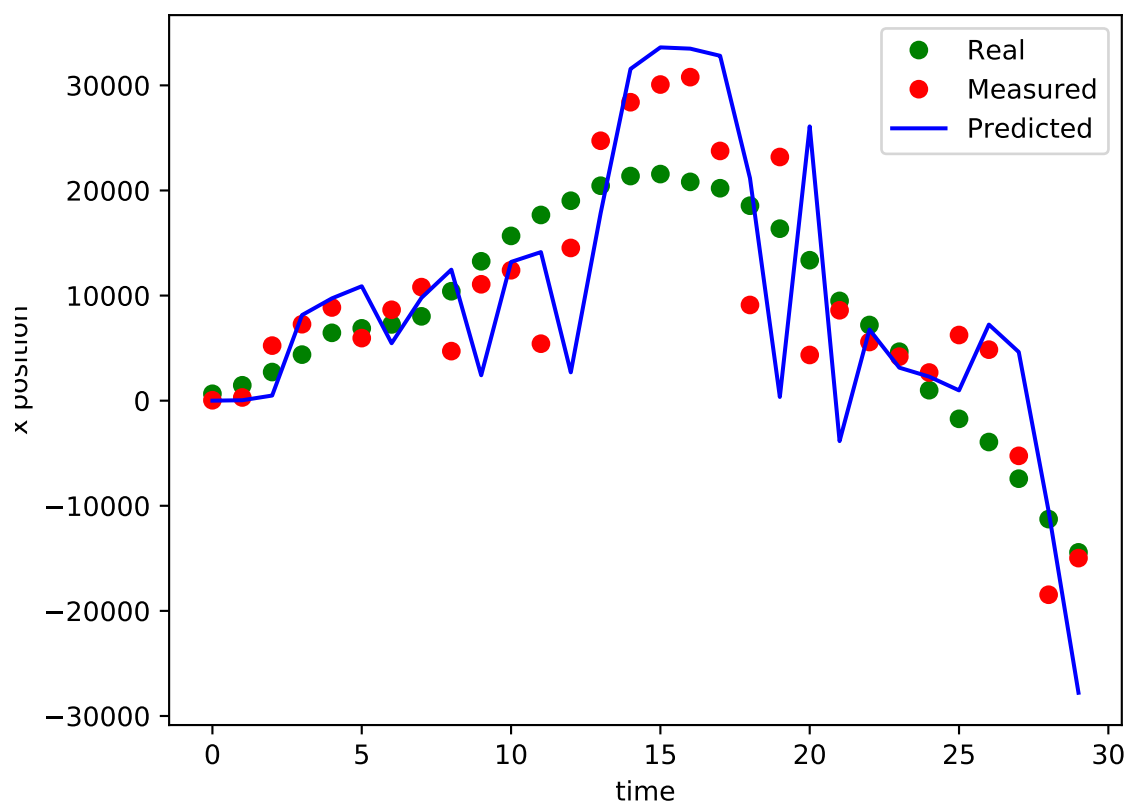


Figura 14: Stima di  $x$  con rumore di processo e di misurazione alto e stato iniziale diverso dallo stato reale.

## 4 Conclusioni

Nel presente lavoro è stata presentata un'implementazione del Kalman Filter e sono stati effettuati diversi esperimenti in diverse condizioni di rumore, sia sul processo che sulle misurazioni. Dai risultati ottenuti possiamo dire che l'algoritmo fornisce predizioni accurate nella maggior parte dei casi, anche in presenza di una quantità media di rumore di processo e di misurazione. Tuttavia se il rumore è molto elevato le predizioni risultano errate. Le predizioni sono inoltre influenzate dalle condizioni iniziali: se queste sono errate, allora le predizioni risultano meno accurate. A tal proposito possiamo dire che le condizioni iniziali, se errate, aggiungono altro rumore.