



# goGPS

free and constrained kinematic  
positioning with low cost receivers

**Eugenio Realini**

*Dottorato di ricerca  
in Geodesia e Geomatica, XXI ciclo*

# Motivazioni

realizzare un software  
che consenta di manipolare  
i parametri di un  
filtro di Kalman

(non possibile con algoritmi  
commerciali black-box)

sfruttare la disponibilità di:  
- reti di stazioni permanenti  
- connettività wireless

per migliorare l'accuratezza  
nella navigazione con  
dispositivi a basso costo



Kinematic surveying  
Precise off-road navigation  
Location Based Services  
GeoWeb (geotagging)



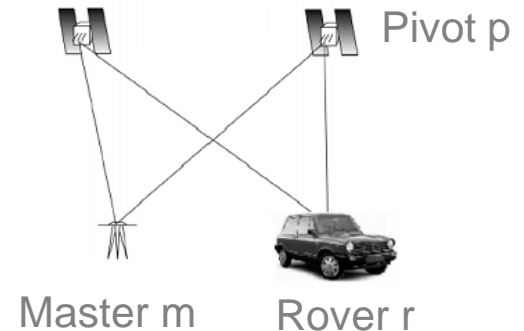
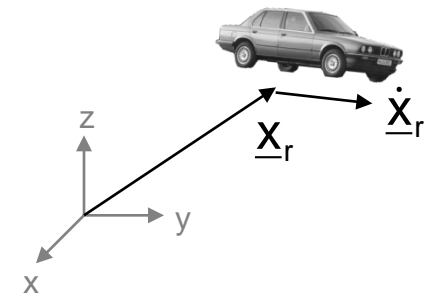
# Le variabili di stato

In goGPS le variabili di stato possono essere divise in due sottogruppi:

$$\underline{X}_t = \begin{bmatrix} x_r \\ \dot{x}_r \\ \vdots \\ y_r \\ \dot{y}_r \\ \vdots \\ z_r \\ \dot{z}_r \\ \vdots \\ N_{rm}^{p1} \\ \vdots \\ N_{rm}^{p32} \end{bmatrix}$$

parametri che descrivono il moto del ricevitore (posizione, velocità, accelerazione, ecc.)

ambiguità di fase (differenze doppie)



# Il modello dinamico

dinamica del ricevitore (per esempio velocità costante)

$$\begin{cases} \mathbf{x}_r(t+1) = \mathbf{x}_r(t) + \dot{\mathbf{x}}_r(t) \\ \dot{\mathbf{x}}_r(t+1) = 0 + \dot{\mathbf{x}}_r(t) + \varepsilon_{\dot{\mathbf{x}}_r}(t) \end{cases}$$

errore di modello

N.B.  $\sigma^2[\varepsilon_{\dot{\mathbf{x}}_r}(t)]$

parametro del filtro di Kalman

“dinamica” delle ambiguità di fase

$$\begin{cases} \mathbf{N}_{rm}^{p1}(t+1) = \mathbf{N}_{rm}^{p1}(t) \\ \vdots \\ \mathbf{N}_{rm}^{p32}(t+1) = \mathbf{N}_{rm}^{p32}(t) \end{cases}$$

# Equazioni d'osservazione

goGPS è basato su osservazioni alle differenze doppie per sfruttare le coordinate note della stazione Master.

$$\begin{cases} P_{rm}^{ps}(t) = \rho_{rm}^{ps}(t) + v_{\text{codice}} \\ \lambda \Phi_{rm}^{ps}(t) = \rho_{rm}^{ps}(t) + \lambda N_{rm}^{ps}(t) + v_{\text{fase}} \end{cases}$$

dipendenti dalle variabili  
di stato  $x_r, y_r, z_r$

( $x_m, y_m, z_m$  note a priori)

ambiguità di fase  
relativa al satellite  $s$

linearizzazione

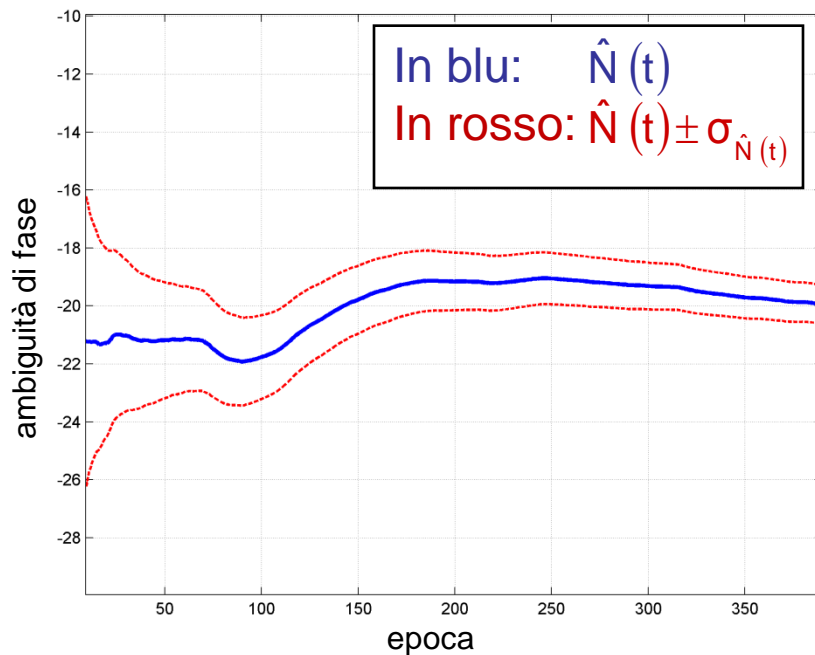
matrice disegno  
delle osservazioni

# Ambiguità di fase

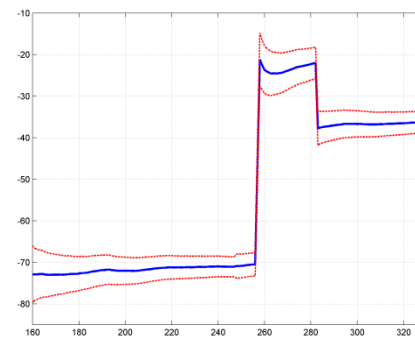
Stima float

Inizializzazione  $\longrightarrow$  confrontando osservazioni di codice e fase

$$\hat{N}_{rm}^{ps} = \frac{P_{rm}^{ps} - \lambda \Phi_{rm}^{ps}}{\lambda} \longrightarrow \sigma_{\hat{N}} \approx 5 \div 10 \text{ cicli}$$



Sono gestiti gli eventi:  
- aggiunta di un satellite  
- perdita di un satellite  
- cycle slip



Esempio di cycle slip

# "Vincolo" di superficie

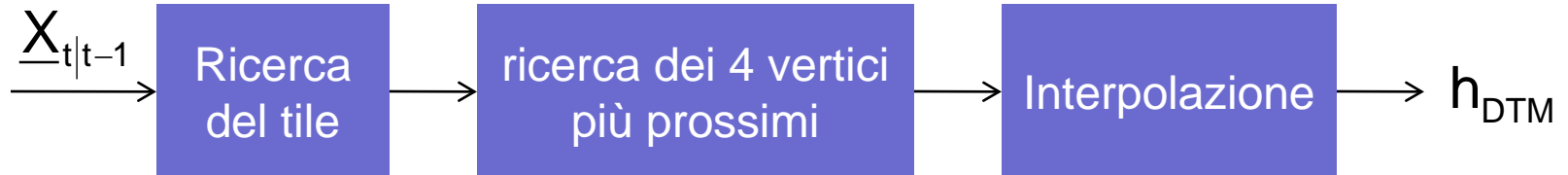
Al fine di migliorare la qualità del posizionamento in quota si introduce una nuova osservazione ricavata da un modello digitale del terreno (DTM):

$$h_{\text{DTM}} = h(x_r, y_r, z_r) + v_{\text{DTM}}$$

$$\sigma_v \approx 30 \text{ cm}$$

DTM ricavato da DSM  
LiDAR 2m×2m elaborato  
e fornito dalla Regione  
Lombardia

Per ottimizzare i tempi di caricamento il DTM è stato suddiviso in tile da caricare dinamicamente con un buffer.



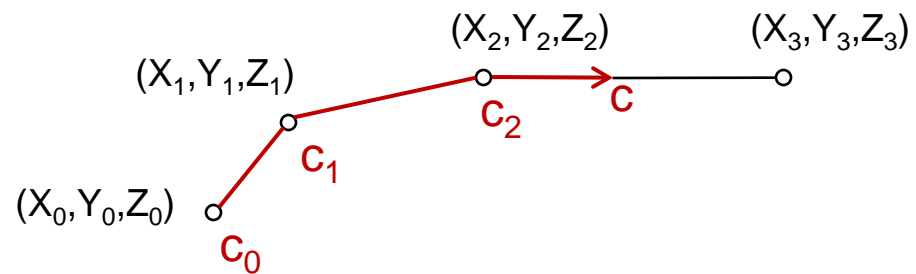
# Vincolo lineare

Nel caso in cui il veicolo si muova lungo un percorso noto a priori (per esempio su strada o rotaia) si può introdurre un vincolo lineare rendendo monodimensionale il moto.

Il vincolo è modellato come una spezzata in 3D e il moto viene descritto tramite una coordinata curvilinea  $c$ :

nuova  
variabile  
di stato:

$$\underline{X}_t = \begin{bmatrix} c_r \\ \dot{c}_r \\ \vdots \\ N_{rm}^{p1} \\ \vdots \\ N_{rm}^{p32} \end{bmatrix}$$

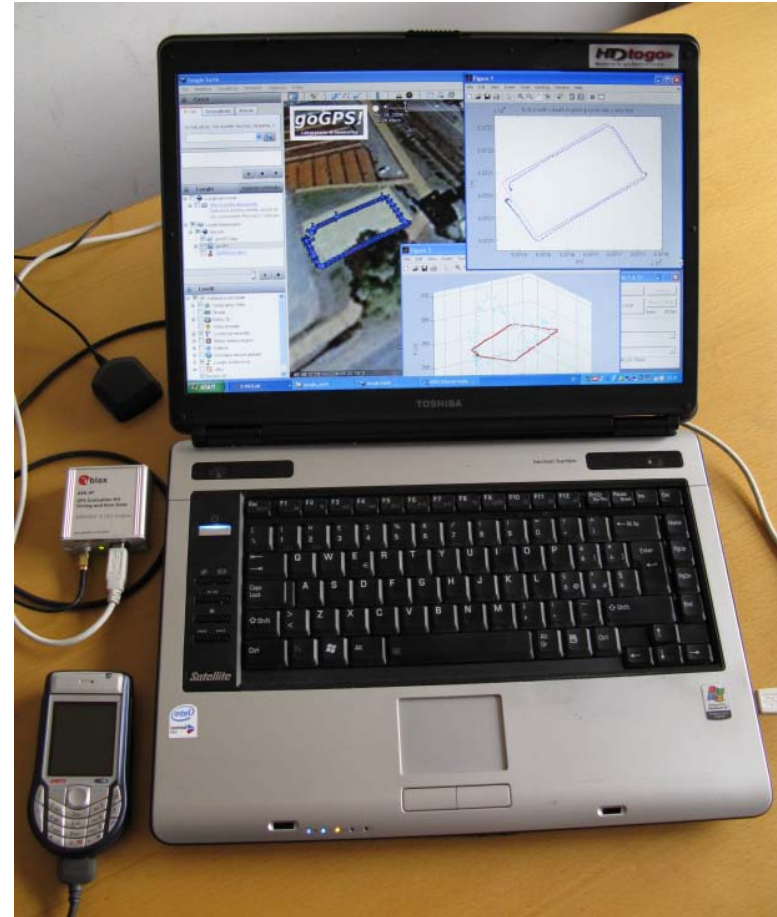


network di strade:

- dati TeleAtlas (completi, costosi)
- dati OpenStreetMap (incompleti, free)

# Software/1

- sviluppato in ambiente MATLAB
- acquisizione dati con frequenza 1s tramite il toolbox "instrument control" (standard TCP-IP e USB)
- aggiornamento della posizione del ricevitore in tempo reale (tempo di calcolo di circa 15 ms su processore Intel Centrino 1.66 GHz)
- visualizzazione della traiettoria in tempo reale sfruttando l'ambiente grafico di MATLAB oppure sovrapposta a immagini satellitari di Google Earth





# Strumentazione/1

eBonTek egps597  
chip ANTARIS 4  
segnale: C/A, L1

Sfrutta una connessione bluetooth restituendo in uscita in formato NMEA i dati già processati.



u-blox AEK-4T  
chip ANTARIS 4  
segnale: C/A, L1



E' dotato di un'antenna esterna e di un'uscita USB attraverso la quale restituisce dati grezzi e/o dati processati (formato NMEA). E' possibile adattare la configurazione alle diverse necessità.

# Strumentazione/2

Leica GPS1200  
segnale: C/A, P  
L1, L2

Ricevitore geodetico, con possibilità  
di effettuare posizionamento RTK.



Leica GS20  
segnale: C/A, L1

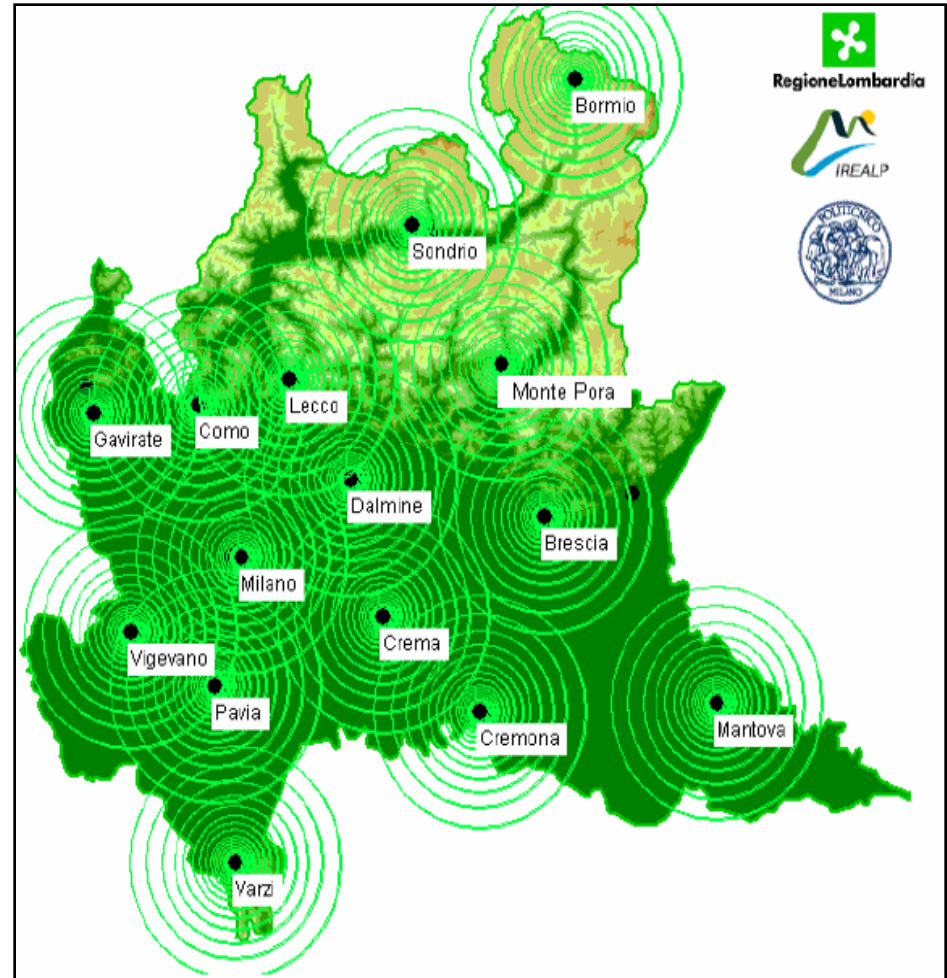
Ricevitore di medio livello, progettato per  
l'aggiornamento cartografico. Supporta il  
posizionamento DGPS.



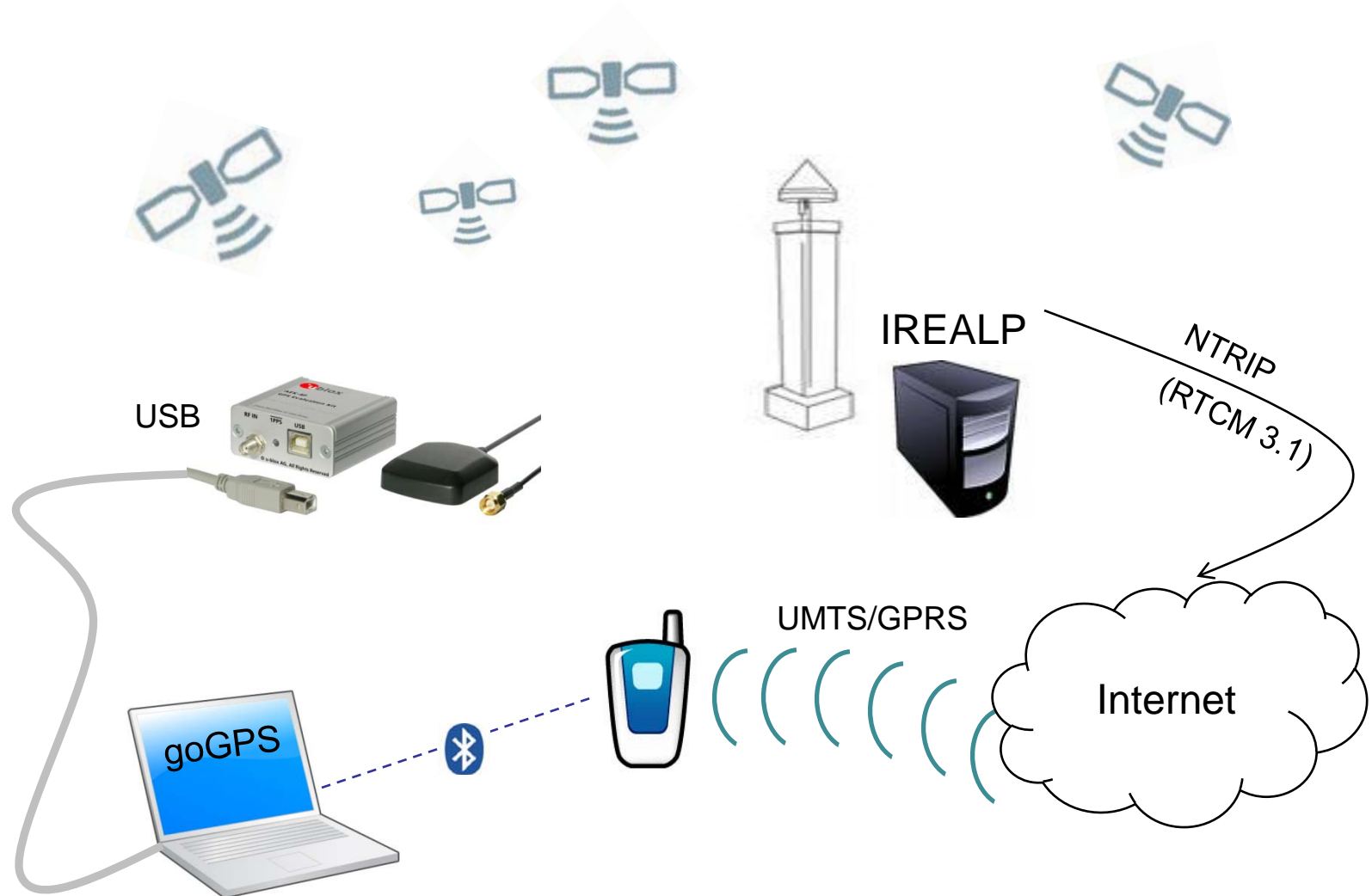
# Strumentazione/3

Stazione permanente di  
Como, usata come Master.  
(Rete di posizionamento GPS  
della Regione Lombardia)

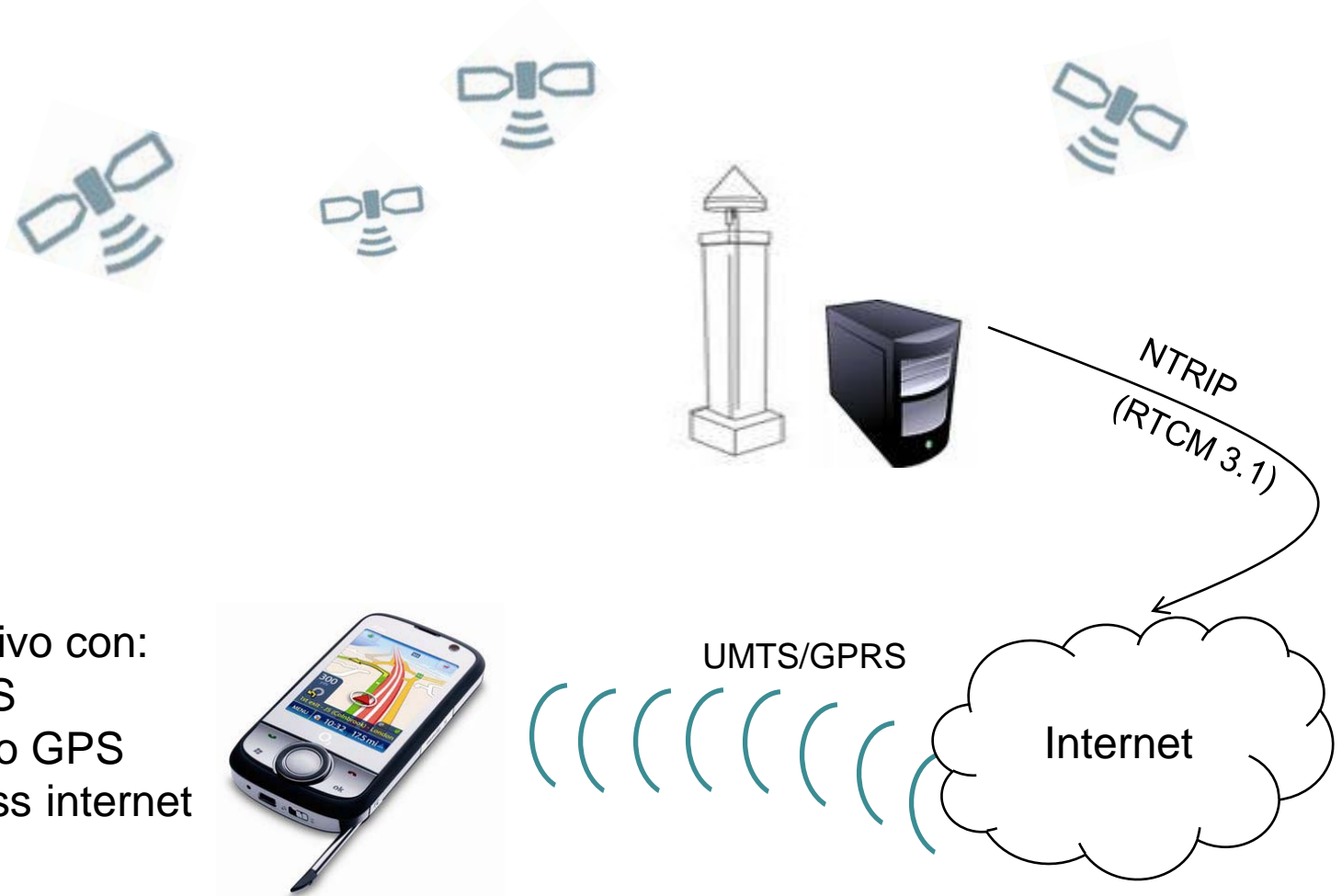
Osservazioni GPS,  
non correzioni differenziali.



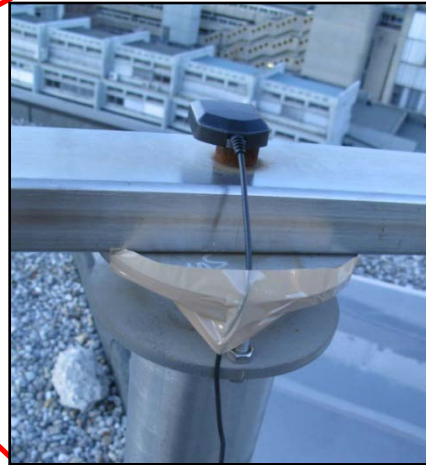
# Strumentazione/4



# Strumentazione/4b



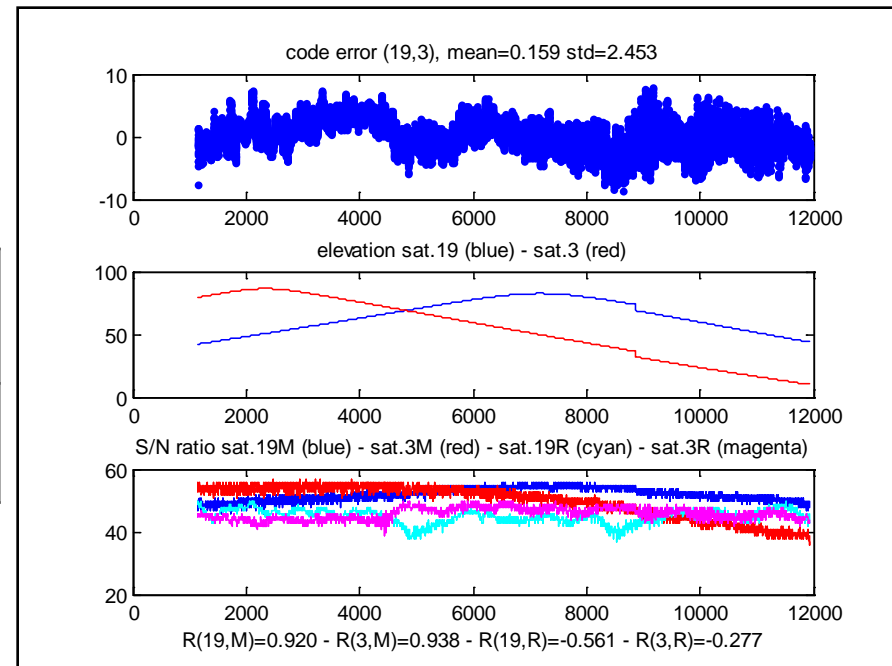
# Analisi u-blox: codice



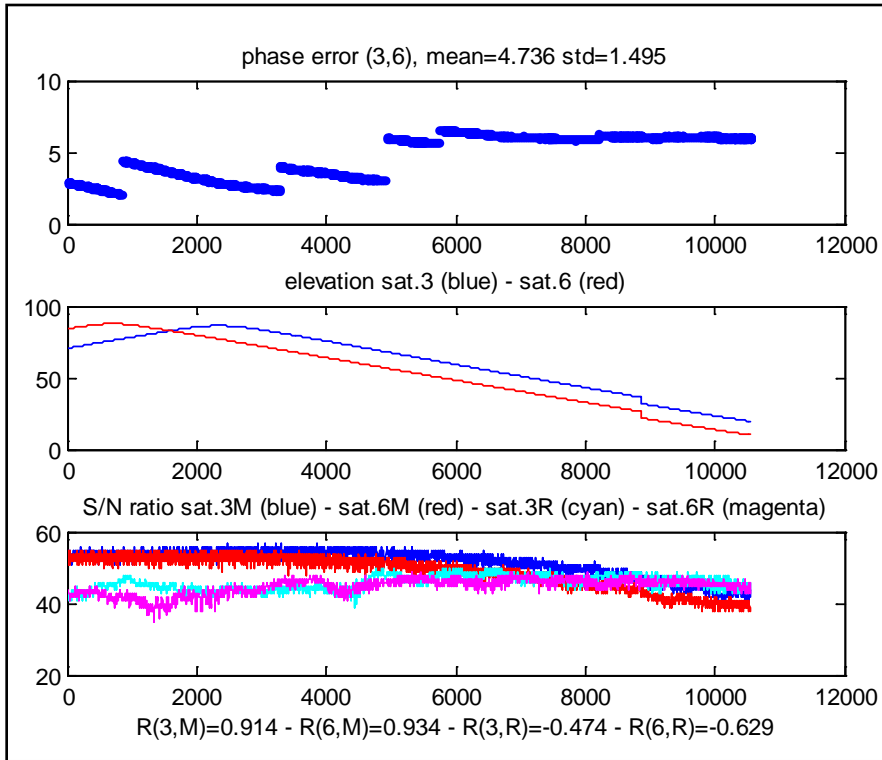
Circa 8 ore di test, u-blox posizionato su un pilastro a distanza nota dalla stazione permanente.

L'errore delle doppie differenze di codice è correlato nel tempo.

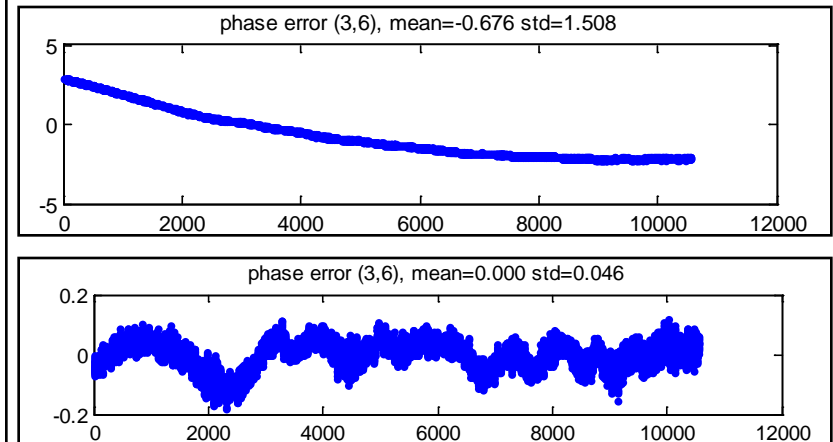
Mean [m]	Std. Dev. [m]	RMSE [m]	Num. of samples
0.565	2.982	3.035	795856



# Analisi u-blox: fase

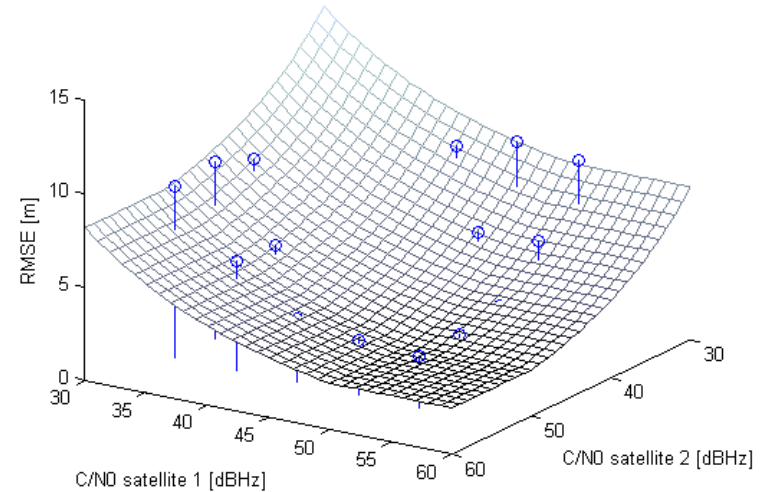
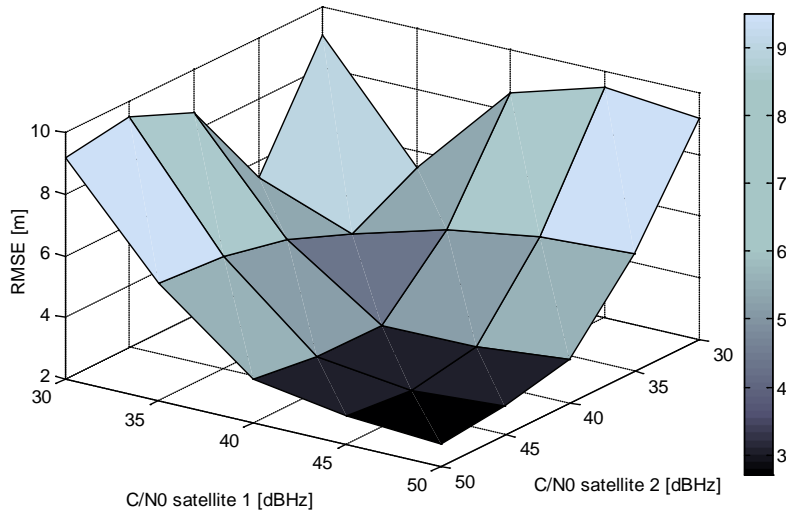


Discontinuità dovute probabilmente ai *clock drift* che caratterizzano questi ricevitori (Odiik et al., 2007)

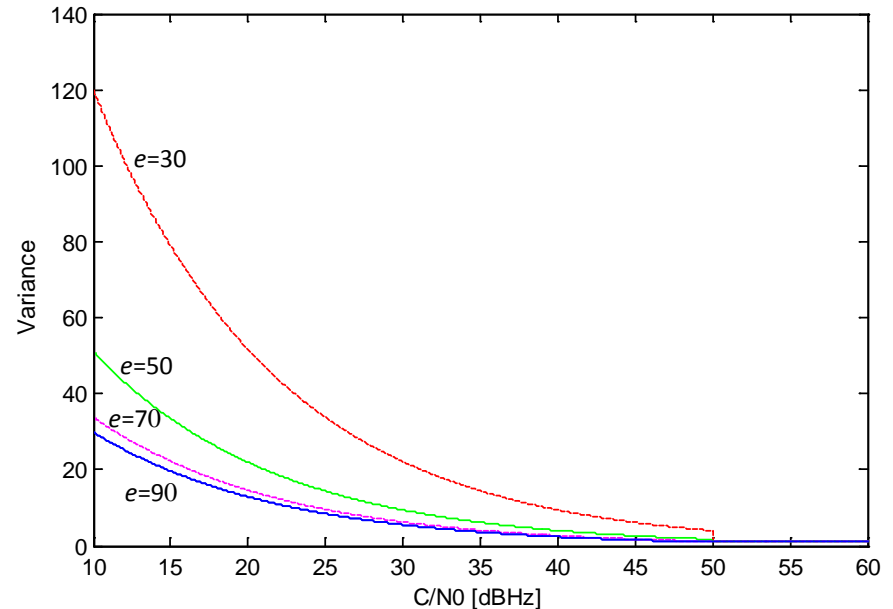


Mean [m]	Std. Deviation [m]	RMSE [m]	Num. of samples
0.000	0.032	0.032	32668

# Pesi delle osservazioni



Definizione di una funzione peso che tenga conto del rapporto segnale/rumore e dell'elevazione dei satelliti.



# Test/1

Strumenti:

- Leica GPS1200
- Leica GS20
- eBonTek eGPS 597
- ev. kit u-blox + goGPS

montati su un'auto circolante lungo un percorso caratterizzato da buona visibilità del cielo.



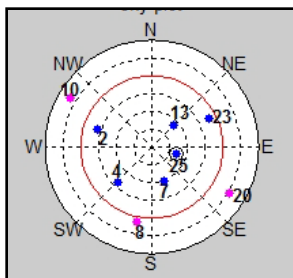
# Test/2

strumenti:

- Leica GPS1200 ●
- Leica GS20 ●
- eBonTek eGPS 597 ●
- ev. kit u-blox + goGPS ●

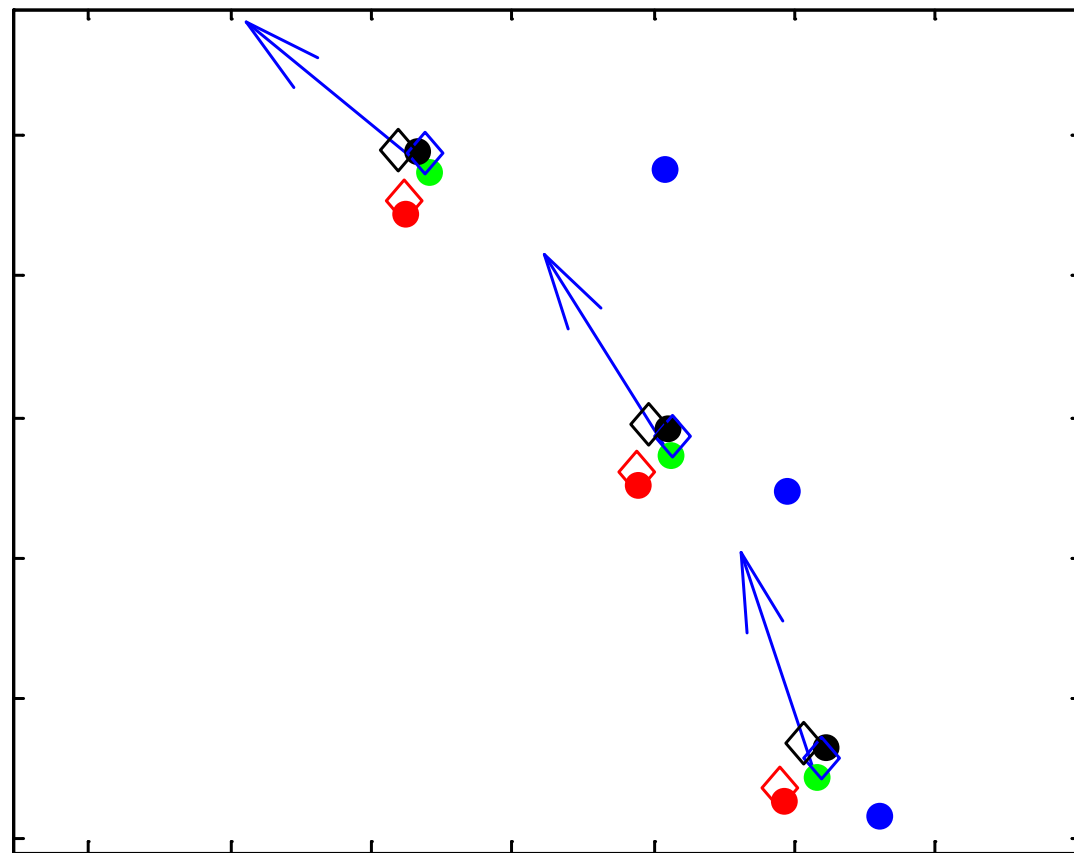
goGPS  
(cutoff = 10°)

media 1.13 m  
dev. st. 0.67 m



goGPS  
(cutoff = 30°)

media 0.78 m  
dev. st. 0.47 m



Leica GS20  
(mod. "Max Accuracy" - DGPS)

media 0.30 m  
dev. st. 0.15 m

eBonTek

media 4.03 m  
dev. st. 1.70 m

# Test/3

Strumenti:

- Leica GPS1200
- eBonTek eGPS 597
- TomTom MkII
- ev. kit u-blox + goGPS

montati su un'auto circolante lungo un percorso caratterizzato da una visibilità del cielo non buona.



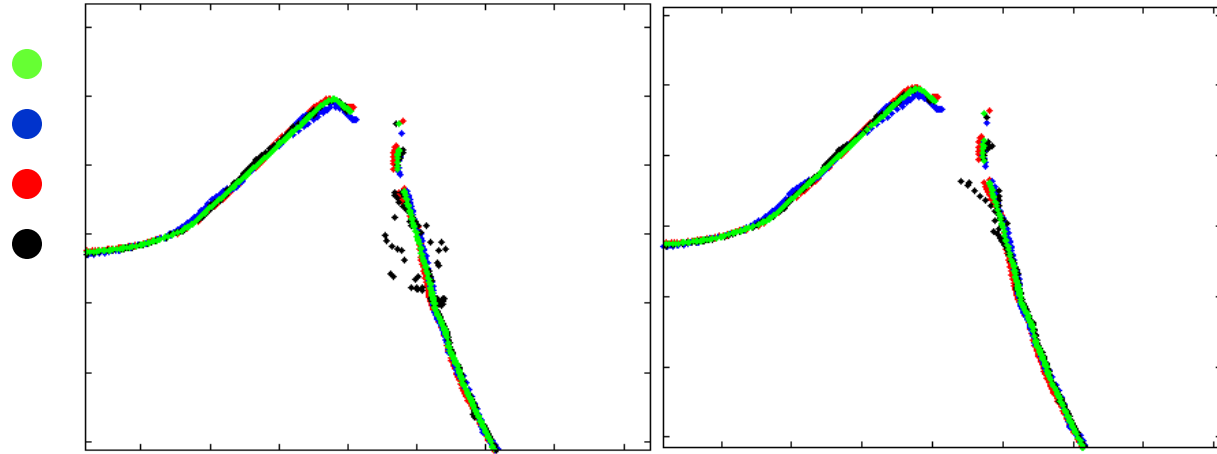
Il Leica GS20 non ha fornito un numero sufficiente di posizioni da poter essere confrontato con gli altri strumenti in modo significativo.

# Test/4

strumenti:

- Leica GPS1200
- eBonTek eGPS 597
- TomTom MkII
- ev. kit u-blox + goGPS

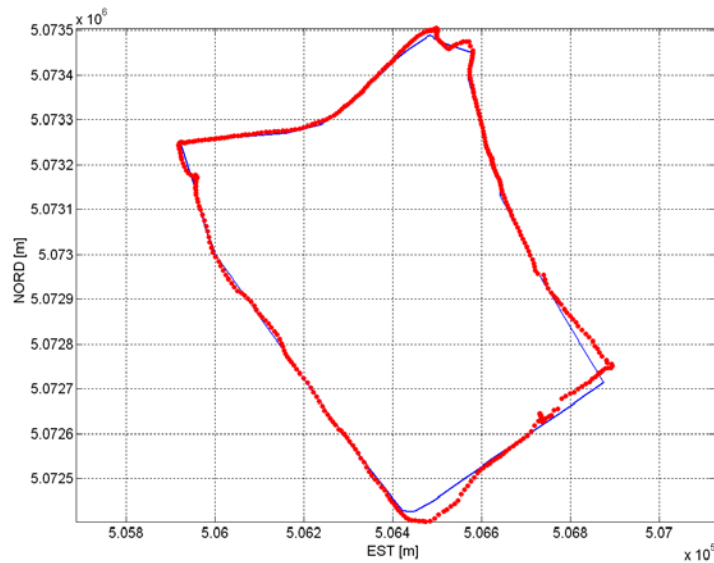
montati su un'auto  
circolante lungo un  
percorso caratterizzato  
da una visibilità del  
cielo non buona.



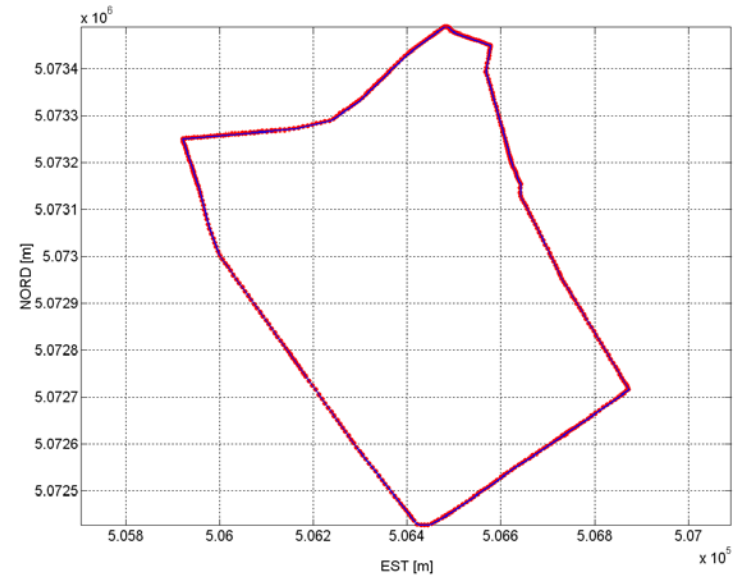
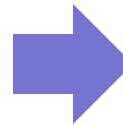
	2D			3D		
	<i>Media [m]</i>	<i>Dev. St. [m]</i>	<i>RMSE [m]</i>	<i>Media [m]</i>	<i>Dev. St. [m]</i>	<i>RMSE [m]</i>
goGPS	4.29	6.04	7.41	8.15	11.46	14.06
goGPS (C/N0>30 dBHz)	1.67	1.12	2.01	3.09	2.11	3.74
goGPS (with DTM)	1.31	0.93	1.61	2.03	0.72	2.16
eBonTek	5.13	2.46	5.69	16.23	5.01	16.99
TomTom	3.68	1.86	4.13	5.50	2.44	6.02

# Test/5

Navigazione vincolata al reticolo stradale:



traiettoria 2D ottenuta con  
goGPS non vincolato



traiettoria 2D ottenuta con goGPS  
vincolato al reticolo stradale

# Conclusioni

- goGPS permette di migliorare le prestazioni dei ricevitori a basso costo appoggiandosi ad una rete di stazioni permanenti.
- è possibile adattare i parametri del filtro di Kalman alle specifiche modalità di navigazione.
- il software consente buone prestazioni anche in condizioni di scarsa ricezione, soprattutto se supportato da informazioni esterne quali DTM o reticoli stradali.

Possibili sviluppi futuri includono:

- gestione completa dei reticoli stradali
- utilizzo informazione doppler
- integrazione con odometri/giroscopi a basso costo
- conversione del software in C/C++
- porting su dispositivi palmari/cellulari
- geoservizi a supporto della navigazione real-time
- sviluppo di goGPS come progetto open source

<http://geomatica.como.polimi.it/elab/goGPS>